

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

ABSTRACT

PARTICIPATORY LEARNING ENVIRONMENT: REQUIREMENTS AND DESIGN ANALYSIS

**by
Vikas Manohar Achhpiliya**

The primary purpose of this thesis is to develop the requirements and model preliminary database and class designs for the Participatory Learning Approach Environment (PLAE). The requirements and the designs will be presented for version 1.0 and 2.0 with version 2.0 supporting an enhanced requirements set.

Participatory Learning Approach Environment (PLAE) is an exercise management tool for the Participatory Learning Approach (PLA) – a pedagogy approach based on the theory of constructivism. The PLAE facilitates and enhances learning among students while easing the administrative burden of the instructor and cognitive overload of the students. The theory and concepts on which the PLA is based are discussed, followed by a discussion on other pedagogy approaches and a comparison of them to the PLA.

**PARTICIPATORY LEARNING ENVIRONMENT:
REQUIREMENTS AND DESIGN ANALYSIS**

by

Vikas Manohar Achhpiliya

**A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Information Systems**

Department of Information Systems

May 2004

Blank Page

APPROVAL PAGE

PARTICIPATORY LEARNING ENVIRONMENT: REQUIREMENTS AND DESIGN ANALYSIS

Vikas Manohar Achhpiliya

Dr. Michael Bieber, Thesis Advisor
Associate Professor of Information Systems, NJIT

Date

Dr. Julian M. Scher, Committee Member
Associate Professor of Information Systems, NJIT

Date

Dr. Katia Passerini, Committee Member
Assistant Professor of Management Information Systems, NJIT

Date

BIOGRAPHICAL SKETCH

Author: Vikas Manohar Achhpiliya
Degree: Master of Science
Date: May 2004

Undergraduate and Graduate Education:

- Master of Science in Information Systems,
New Jersey Institute of Technology, Newark, NJ, 2004
- Bachelor of Commerce
University of Mumbai, Mumbai, India, 1999

Major: Information Systems

“खुद ही को कर बुलुन्द इतना के हर तक्दीर से पेहे खुदा
बन्दे से खुद पुछे बता तेरी रज़ा क्या है ।”

(Strive to achieve such a level of greatness that before good luck bestows its gifts upon you, may God come and ask ‘Son, what is it that your heart desires?’)

This thesis is dedicated to:

My Dad, for always believing in me and encouraging me.

My Mom, for always supporting me every step of my life.

My Brother, because I love him.

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my thesis advisor, Dr. Michael Bieber, without whom this thesis could not have been accomplished. I am grateful to him for sharing his vast wealth of knowledge and for providing me with access to valuable resources. It has been an exhilarating experience to work with a great researcher like him.

I wish to offer special thanks to Dr. Julian Scher and Dr. Katia Passerini for actively participating in my committee and providing me with valuable advice and words of encouragement.

Many thanks are extended to Anirban Bhaumik for constantly providing me with valuable technical inputs. It was a great learning experience to work with a knowledgeable person like him.

Finally, I wish to thank all my friends through my years of education here in the United States. Special thanks are offered to Ankita Raut and Siddharth Sheopori for not only helping me finish my thesis but also for being a family away from home. I am fortunate to be among them.

TABLE OF CONTENTS

Chapter	Page
1 CONCEPTS AND THEORY	1
1.1 Introduction.....	1
1.2 Constructivism.....	1
1.3 Assessment	2
1.3.1 Self Assessment	4
1.3.2 Peer Assessment.....	5
1.3.3 Collaborative Assessment.....	6
1.4 Peer Assessment Techniques.....	6
1.4.1 Networked Innovative Assessment Procedures	6
1.4.2 Goldfinch’s Peer Assessment Technique.....	8
1.5 Tools Used for the Assessment Techniques	9
1.5.1 NetPeas	9
1.5.2 SPARK.....	9
1.6 Participatory Learning Approach: Introduction.....	10
1.7 Workflow	11
1.8 Comparing Peer Assessment Techniques with PLA	13
1.9 Prior PLA Research	14
1.10 PLAE: Participatory Exercise Management Tool	17
1.11 Explanation of PLAE Terms	17
1.12 Relationship between Assignments, Artifacts and Template Sets.....	21
1.13 PLAE Example	23

TABLE OF CONTENTS

(Continued)

Chapter		Page
2	REQUIREMENT ANALYSIS FOR PLAE VERSION 1.0.....	26
2.1	Login to PLAE.....	27
2.2	School Management	30
2.2.1	View Schools	30
2.2.2	Add New School	30
2.2.3	Add New Contact Person.....	32
2.2.4	Edit School Details	33
2.2.5	Delete School	33
2.3	Instructor Management.....	33
2.3.1	View Instructors.....	34
2.3.2	Add New Instructor.....	34
2.3.3	Edit Instructor	36
2.3.4	Delete Instructor.....	37
2.3.5	Assign Instructor Role to Student	37
2.4	Experimenter Management.....	38
2.4.1	View Experimenters.....	38
2.4.2	Add New Experimenter	38
2.4.3	Edit Experimenter	41
2.4.4	Delete Experimenter	41
2.5	Course Management	41
2.5.1	View Courses	41

TABLE OF CONTENTS

(Continued)

Chapter		Page
	2.5.2 Add New Course.....	42
	2.5.3 Edit Course.....	44
	2.5.4 Delete Course.....	44
2.6	Course Options	44
	2.6.1 Select a Course.....	45
	2.6.2 Logoff (For All Users)	46
	2.6.3 Help (For All Users)	46
	2.6.4 Mark Read.....	46
	2.6.5 Change Password (For All Users).....	47
	2.6.6 Visual Indication for Pending Tasks.....	47
	2.6.7 Administer (Only for Instructor).....	48
	2.6.8 Edit Profile (Only for Student)	49
2.7	Student Registration.....	49
	2.7.1 New Student Registration	49
	2.7.2 Instructor Creating a Student Profile	52
2.8	Discussion Management.....	53
	2.8.1 Post Message.....	53
	2.8.2 Reply	55
	2.8.3 Previous Message.....	56
	2.8.4 Next Topic	56
	2.8.5 Mark Message Read.....	57

TABLE OF CONTENTS

(Continued)

Chapter		Page
	2.8.6 Mark Thread Read	57
	2.8.7 Discussion Area Management (Only for Instructor)	57
	2.8.8 Create New Discussion Area (Only for Instructor)	58
	2.8.9 Delete Discussion Area (Only for Instructor)	58
2.9	Admission Control	59
	2.9.1 Request Admission (Only for Student)	59
	2.9.2 Admission Controller (Only for Instructor)	60
	2.9.3 Grant/Decline Admission (Only for Instructor)	60
	2.9.4 Revoke Admission (Only for Instructor)	61
2.10	Assignment Management	62
	2.10.1 View Assignments	62
	2.10.2 Add Assignment	62
	2.10.3 Edit Assignment	65
	2.10.4 Delete Assignment	66
2.11	Artifact Management	66
	2.11.1 View Artifacts (Only for Instructor)	67
	2.11.2 Add New Artifact (Only for Instructor)	67
	2.11.3 Edit Artifact (Only for Instructor)	74
	2.11.4 Delete Artifact (Only for Instructor)	74
	2.11.5 Submit Artifact	75
	2.11.6 View/Edit Submitted Artifact	79

TABLE OF CONTENTS

(Continued)

Chapter		Page
2.12	Template Management	80
2.12.1	View Template Sets	81
2.12.2	View Template Layout	81
2.13	Workflow Management	82
2.13.1	Create workflow (Only for system)	82
2.13.2	Workflow Viewer (Only for instructor)	83
2.14	Student Allocation Tool.....	83
2.14.1	Manual Generation of Student Allocations (Only for Instructor)	84
2.14.2	View Student Allocations	85
2.14.3	Automatic Generation of Student Allocations (Only for System)	85
2.15	Reminder Preferences	86
2.15.1	Set Default Reminder Preferences (Only for Instructor).....	86
2.15.2	Send Reminder Email (Only for System).....	87
2.16	View/Edit Final Grades	87
2.16.1	View Final Grades	88
2.16.2	Edit Final Grades	89
3	DATABASE AND CLASS DESIGNS FOR PLAE VERSION 1.0.....	90
3.1	Entity Relationship Diagram	90
3.2	Class Diagram.....	105

TABLE OF CONTENTS

(Continued)

Chapter		Page
4	REQUIREMENT ANALYSIS FOR PLAE VERSION 2.0.....	115
4.1	Course Management	115
4.1.1	Add New Course.....	116
4.1.2	Create a 'Cross-Listing' Course.....	117
4.1.3	Edit Course.....	119
4.2	Assignment Management	119
4.2.1	Add Assignment.....	120
4.2.2	Duplicate Assignment.....	121
4.3	Reminder Preferences	122
4.4	Manage Messages	124
4.5	Artifact Management	125
4.5.1	Add New Artifact (Only for Instructor/TA)	125
4.5.2	Submit Artifact.....	128
4.6	Automatic Evaluation Tool.....	128
4.7	Template Management	129
4.8	Group Management	134
4.8.1	Create Criteria-Based Groups (Automatic)	134
4.8.2	Create Groups (Manual)	137
4.9	Student Registration.....	138
4.10	Student Allocation Tool.....	139

TABLE OF CONTENTS

(Continued)

Chapter	Page
4.10.1 Manual Generation of Student Allocations (Only for Instructor)	140
4.10.2 Automatic Generation of Student Allocations (Only for System)	141
5 DATABASE AND CLASS DESIGNS FOR PLAE VERSION 2.0	142
5.1 Entity Relationship Diagram	142
5.2 Class Diagram.....	160
6 CONCLUSION AND FUTURE WORK	168
6.1 Limitations	168
6.2 Future Extensions	168
6.3 Conclusion	169
APPENDIX A PRE-DEFINED ELEMENT_TYPES ENTITY	171
APPENDIX B PLAE ARTIFACT CONFIGURATION EXAMPLE.....	172
APPENDIX C SCREEN AFTER LOGIN FOR INSTRUCTOR/STUDENT	174
APPENDIX D SUBMIT ARTIFACT SCREEN FOR OBJECTIVE QUESTIONS	175
REFERENCES	176

LIST OF TABLES

Table	Page
1.1 Five Workflows in an Example PLA Implementation	12
1.2 Comparisons of Peer Assessment Techniques with PLA	14
3.1 School Entity.....	92
3.2 User Entity	92
3.3 Instructor_School Entity	93
3.4 Student_School Entity	93
3.5 School_Contact Entity	94
3.6 Experimenter_School Entity	94
3.7 Course Entity	94
3.8 Admission Entity	95
3.9 Template_Sets Entity	95
3.10 Templates Entity	96
3.11 Element_Types Entity.....	97
3.12 Template_Definition Entity	97
3.13 Discussion_Messages Entity.....	98
3.14 User_Messages_Read Entity.....	98
3.15 Assignment Entity.....	99
3.16 Artifact Entity	100
3.17 Artifact_Roles_NotAllowed Entity	101
3.18 Evaluation_Artifact_Points_Breakdown Entity.....	102
3.19 Artifact_Submissions Entity	102

LIST OF TABLES

(Continued)

Table	Page
3.20 Workflow Entity	103
3.21 Reminder_Preferences Entity	104
3.22 Final_Grades Entity	105
3.23 Class User	107
3.24 Class Student.....	107
3.25 Class Instructor	108
3.26 Class Administrator	108
3.27 Class Experimenter	109
3.28 Class AdmissionController	110
3.29 Class StudentRegistrar	110
3.30 Class DiscussionArea.....	110
3.31 Class School.....	111
3.32 Class Reminder	111
3.33 Class StudentAllocation.....	112
3.34 Class Artifact	112
3.35 Class Assignment.....	112
3.36 Class Course.....	113
3.37 Class Workflow	113
3.38 Class TemplateSet.....	113
3.39 Class FinalGrade	114
5.1 User Entity	144

LIST OF TABLES

(Continued)

Table	Page
5.2 Course Entity	146
5.3 Template_Sets Entity	147
5.4 Template_Definition Entity	148
5.5 Assignment Entity.....	149
5.6 Artifact_Submissions Entity	151
5.7 Workflow Entity	152
5.8 Reminder_Preferences Entity	153
5.9 Template_Sections Entity	154
5.10 Course_Max_Eval Entity.....	155
5.11 Course_TA Entity	155
5.12 Course_Instructors Entity	156
5.13 Course_Merger Entity.....	156
5.14 Group_Members Entity	156
5.15 Groups Entity	157
5.16 Group_Sets Entity	157
5.17 Major Entity	159
5.18 User_Messages_Preferences Entity	159
5.19 Class DiscussionArea.....	163
5.20 Class Reminder	164
5.21 Class Assignment.....	165
5.22 Class Course.....	165

LIST OF TABLES
(Continued)

Table	Page
5.23 Class TemplateSet.....	166
5.24 Class GroupSet.....	166
5.25 Class GradingPreference.....	167

LIST OF FIGURES

Figure	Page
1.1 Assignment-artifact-template set relationship.	22
1.2 Variation in assignment-artifact-template set relationship.	23
3.1 ER diagram for PLAE version 1.0.	91
3.2 Class diagram for PLAE version 1.0.	106
5.1 ER diagram for PLAE version 2.0.	143
5.2 Class diagram for PLAE version 2.0.	161

CHAPTER 1

CONCEPTS AND THEORY

1.1 Introduction

Pedagogy approaches have evolved over the years with different theories of learning and teaching being conceived. Constructivism is one such theory about learning which suggests that learning is not the result of teaching but it is a result of constructing one's own knowledge. The PLA has its foundations laid on the theory of constructivism, according to which students learn by creating problems, solving the problems and evaluating the solutions themselves.

In this chapter, the theory of constructivism will be discussed with other concepts such as peer, self and collaborative assessment, all of which can be included in PLA depending upon the learning objectives of the class. Goldfinch's (1994) and Liu et al (2002) peer assessment approach will be discussed followed by a discussion on the software tools which implemented these approaches: SPARK and NetPeas, respectively.

1.2 Constructivism

According to the theory of constructivism (Piaget 1928, Vygotsky 1978), learning is not the result of teaching; rather it is the result of what students do with the information presented to them (Sewell, 2002). Learners are not passive recipients of knowledge (Sewell, 2002), they have to create their own knowledge and associate it to their existing knowledge. The emphasis of constructivism is not remembering stuff but understanding the concepts (Fox, 2001). People learn best when they actively construct their own

knowledge (Wilson and Lowry 2000, Tytler 2002), organize it, look for internal patterns and relate it to what they already know (Perkins, 1999).

There are 3 types of roles in constructivism (Phillips, 1995). Perkins (1999) calls them *active learner*, *social learner* and *creative learner*. *Active learners* take a more active role in learning and are more participative in the entire learning process instead of just listening; they discuss, debate, question, hypothesize and investigate. *Social learners* construct their own knowledge by getting involved in dialogue or a conversation. *Creative learners* create or recreate their own knowledge with guidance from their teachers.

Fox (2001) has summarized constructivism in the following six points:

- (1) Learning is an active process.
- (2) Knowledge is constructed, rather than innate, or passively absorbed.
- (3) Knowledge is invented not discovered.
- (4a) All knowledge is personal and idiosyncratic.
- (4b) All knowledge is socially constructed.
- (5) Learning is essentially a process of making sense of the world.
- (6) Effective learning requires meaningful, open-ended, challenging problems for the learner to solve.

1.3 Assessment

Boud (1990) identifies two purposes for assessment (1) Formative assessment or assessment for learning which intends to improve the quality of student learning. Students are given encouragement, response and feedback on what they do with a view

for them to become more effective in their learning. (2) Summative assessment which is concerned with accreditation of knowledge. It is primarily concerned with awarding a degree or diploma to a student by making a judgment based on various components of assessment.

Traditionally, assessment is considered to be the responsibility of an expert, and in the academic field that role of an expert is generally assumed by the instructor. The instructor would set assessment tasks for the students and then grade those against the expectations of the instructor that defined learning outcomes (Fallows and Chandramohan, 2001). However, in this approach, students can yield to techniques such as intensive memorizing and rote application of formulas (Fallows and Chandramohan 2001, Entwistle 2000, Hargreaves 1997) if they are aware of the assessment criteria the instructor is looking for and can prepare themselves for those criteria (Miller and Parlet 1974, Sewell 2002). This results in “surface” learning without the students understanding the “deep” concepts of the learning material which contravenes the whole objective of the learning exercise. Real knowledge requires “deep” understanding (Sewell, 2002). Students often do not apply the concepts and theories learnt in classrooms to their real life situations (Perkins, 1999).

Stefani (1998) proposed a more active participation of students in the assessment process to make them better reflective learners. The assessment process has a very powerful influence on learning and if designed well can reap huge benefits to teaching and learning approaches (Boud et al., 1999, McConnell 1999). The assessment process should be reflective of the teaching and learning objectives (Boud, 1990). Boud (1990) and Hargreaves (1997) believe that assessment practices should develop lifelong learning

skills such as ability to evaluate one's own work and of one's peers, which the employers seek from students when they enter the professional world.

Several studies (Davies 2000, Liu et al., 2002, Fallows and Chandramohan 2001, Kwok and Ma 1999, McConnell 1999) in the past have ventured in the area of peer assessment, self assessment and collaborative assessment with positive results. The results showed that students were satisfied with a new approach of being assessed by someone else other than their instructors. The results also highlighted that students showed a "deep" approach to learning instead of "surface" learning where "deep" learning can be defined as a thorough understanding of concepts. These different assessment techniques contribute to learning in various ways.

1.3.1 Self Assessment

In self-assessment a student does an honest critical evaluation of his or her own work, by setting his/her own goals and continually controlling his/her work against personally set criteria and evaluating the final outcome (Higgins et al., 1994). Self assessment refers to the involvement of learners in making judgments about outcomes or achievements of their learning (Liu et al., 2002).

In a study conducted by Orsmand, et al (1997), they showed that self-assessment is an important educational tool that can be used to teach students to evaluate and criticize their own work. Dochy and Segers (1999) list two guidelines for educators who wish to employ self-assessment in their style of teaching. The two guidelines are as follows: (1) Self-assessment takes time, and sometimes support for students will be necessary during the self-assessment. (2) Self-assessment can be used fairly easily for formative purposes. Students should learn to see this as a tool for learning.

1.3.2 Peer Assessment

In peer assessment, students' work is evaluated and assessed by their peers. Students evaluate and grade each others and learn to compare the quality of their own work with those of their peers. Peer assessment leads to peer learning which refers to students learning from and with each other without immediate intervention of the instructor (Boud et al., 1999). Fry (1990) also describes the process of peer assessment as very natural and something that everyone engages in since childhood.

Peer assessment can be a valuable tool, if used correctly, in formative assessment (Dochy and Segers 1999, Davies 2000, Fallows and Chandramohan 2001, Liu et al., 2001). It encourages the students to take a more active role both in learning and in the assessment process. Peer assessment process promotes metacognition (Liu et al., 2001). Metacognition has two aspects: knowledge and regulation of an individual's cognition. The knowledge aspect refers to knowing about one's strengths and weaknesses, the task demands and the learning strategies. The regulation aspect refers to control of one's cognitive process such as planning, monitoring of execution and checking of cognitive strategy. Students indicated a high level of satisfaction with the whole process of peer assessment in a study conducted by Gatfield (1999) and reported that they were motivated to learn from the peer review process (Liu et al., 2001).

Various researchers have expressed their apprehensions about using peer rating in higher education as they believe that social relationships and friendship between those who are being assessed can act as a personal source of bias (Dancer and Dancer 1992, Brindley and Scofield 1998, Rafiq and Fullerton 1996). However, various studies (Douglas

2001, Montgomery 1986) have demonstrated that relational bias of this kind was either absent or negligible and it is possible for peer assessment to be relatively free of bias.

1.3.3 Collaborative Assessment

Collaborative assessment involves a negotiation and agreement of assessment methods between students and instructors (Harris and Bell, 1994). As the students and instructors agree on the evaluation scheme, they collaboratively produce the best effort by offering help and continuous feedback to their peers (Kwok and Ma, 1999).

Using peer, self or collaborative assessment techniques encourages students to critically evaluate their own and other's contributions and to develop the ability to recognize their own strengths and weaknesses and those of their team members (Freeman and McKenzie, 2002). However for a successful self, peer or a collaborative assessment process to happen, students and instructors have to develop a sense of trust among them and understand that they all belong to a community of learners (McConnell, 1999).

1.4 Peer Assessment Techniques

Two peer assessment techniques from the current literature are discussed below:

1.4.1 Networked Innovative Assessment Procedures

Liu et al (2002) have integrated self assessment into networked peer assessment procedures and call this new combination as networked innovative assessment procedures. Networked innovative assessment procedures comprise of the following steps:

- (1) Instructor seriously demonstrates the educational objectives of peer and self assessment in the beginning and uses some real samples to prepare students for later assessment activities.
- (2) Instructor repeats the following steps for each assignment.
- (3) Instructor teaches a part of course materials covered in the semester.
- (4) Students and instructor jointly discuss assignments and the criteria for marking.
- (5) NetPeas (*a networked peer assessment tool, explained later*) randomly assigns reviewers (each reviewer marks three assignments).
- (6) Reviewers assign a score to the assignment and comment on it.
- (7) At the same time, reviewers also assign a score to their own works and write a narrative to describe the values of their own works.
- (8) The instructor marks each student's work and observes the comments written by students without revealing the results.
- (9) NetPeas notifies the students of browsing grades and comments given by peers and comparing peer and self assessment.
- (10) Based on those comments, each student must make corrections or modifications accordingly.
- (11) Steps 3 to 10 are repeated once, twice or not at all (researchers can choose the times of repetition based on their needs).

After the student has submitted his/her own work, students have to grade the work of their peers as assigned by the system. The suggestions and grades as evaluated by the system are then distributed back to the original author. The author must reflect on the

differences of self and peer assessment and then modify their original work based on the self-regulated recommendations from peer reviewers.

1.4.2 Goldfinch's Peer Assessment Technique

Goldfinch and Raeside (1990) suggested the first peer assessment technique but Goldinch (1994) advised some refinements to that technique. Li (2001) confirmed the validity of Goldfinch's technique by applying it to engineering student group projects. The peer assessment technique is made up of two parts: Part 1 seeks to determine the contribution of each student in the tasks making up the project. Part 2 is more concerned with recognizing each students' participation in the functioning of the group. In the 1994 refinement, Goldfinch included that each student also grades his/her performance to guard against the problem where over-generous students effectively penalize themselves. The technique's major emphasis is on building group-working among the students. In Part 1, students are asked to list the tasks for the project and the main contributors to each task. In Part 2, each student marks his/her peers for their contribution to the group within a range of -1 to 3 points. These scores are used to generate a 'PA score' which is converted by means of look-up table or formula to a 'PA factor' which gives the percentage of the group's mark to be awarded to the student.

1.5 Tools Used for the Assessment Techniques

1.5.1 NetPeas

Liu et al (2002) designed a Networked Peer Assessment System known as NetPeas. NetPeas was designed to facilitate peer, self and instructor assessment in undergraduate students in an operating systems course. NetPeas was designed to achieve the following objectives:

- 1) Higher anonymity levels than paper-and-pencil peer assessment.
- 2) Increased freedom of time and location for learners.
- 3) Ability of students to modify their work more timely.
- 4) Increased student-instructor and student-student interaction and feedback.
- 5) Lower transmission and delivery costs than “paper-and-pencil” peer assessment.
- 6) Fewer limitations on transmission of data than “paper-and-pencil” peer assessment.

1.5.2 SPARK

Freeman and McKenzie (2002) designed a self and peer assessment tool called SPARK, acronym for Self and Peer Assessment Resource Kit, based on Goldfinch’s (1994) well designed and evaluated paper-based system. SPARK’s objectives were to reduce the limitations of paper-based systems and enable self and peer assessment of teamwork to be used with any number of students. SPARK was designed for use in disciplines in which it was important to develop students’ teamwork ethic, teamwork is assessed and web-based learning was a major part of the learning.

1.6 Participatory Learning Approach: Introduction

Participatory Learning Approach (PLA) is a pedagogy approach (Shen et al., 2004b), based on the theory of constructivism, emphasizes on active involvement of students in the full life cycle of the homework, projects, quizzes, exams, and other kinds of problems. Full life cycle means creating, solving and evaluating the problem. So in simple words, a student (*problem creator*) will create the problem, another student (*problem solver*) will solve the problem and *problem creator* will evaluate and grade the solution. The PLA also provides an optional step wherein the problem solver can dispute over the grade assigned to him/her by the problem creator which will in turn be arbitrated by the instructor or a student.

Involvement of the student at every step serves a different purpose. While creating the problems, students need a deep understanding of the course material to create a sufficient enough difficult problem. Solving the problem requires the students to be articulate and precise in their solutions as these solutions will be graded by their peers. Evaluating other students' solutions requires the students to be fair and accurate in their judgment of their peers' work. Evaluating their peers also helps students to compare their work with the work of their peers and develop a lifelong skill of making correct evaluations. By actively participating in all steps of a problem life cycle, students not only learn from their work but also read, assess and learn from other students' work. They can compare their work to the work of their peers which provides them with new ideas and drives them to think of ways to make their work better.

The above mentioned homework cycle is in its simplest form and by no means exhaustive. The PLA is very flexible as an approach allowing the instructor to

add/modify/delete steps throughout life cycle of the homework depending upon the learning requirements of the students and the course. The PLA can be implemented with individual students and also in groups.

1.7 Workflow

Every instance of a full life cycle of homework is referred to as ‘Workflow’ in PLA. Every workflow has several ‘events’ which can be defined as something that a participant or a system does. Some examples of events are designing the problem, solving the problem, evaluating the solution, registering a dispute, etc. Every event of the workflow has a ‘role’ associated with it. A role can be defined as the participant responsible for carrying out the event. For example, for the event creating the problem, the role associated could be ‘problem creator’, for the event solving the problem, the role could be ‘problem solver’, for evaluating the problems, the role could be ‘solution evaluator’ and for registering the dispute the role could be ‘disputer’.

Table 1.1 below gives a detailed example of five workflows and demonstrates the flexibility of PLA in terms of events a workflow can have.

Table 1.1 Five Workflows in an Example PLA Implementation

Workflow Instance	Problem Creator (Creating Problems)	Problem Reviewer (Reviewing Problems)	Problem Solver (Solving Problems)	Evaluator #1 (Evaluating solution)	Evaluator #2 (Evaluating solution)	Disputer (Registering Dispute)	Dispute Arbitrator (Dispute Arbitration)
1	Alan	Carlos	Betty	Alan	Carlos	Betty	Instructor
2	Betty	Eva	Carlos	Betty	Dan	Carlos	Instructor
3	Carlos	Alan	Dan	Carlos	Eva	Dan	Instructor
4	Dan	Betty	Eva	Dan	Alan	Eva	Instructor
5	Eva	Dan	Alan	Eva	Betty	Alan	Instructor

In Table 1.1 a workflow consists of 7 events:

- (1) Creating problems – each student creates a problem.
- (2) Reviewing problems – the problems created are reviewed by another student for quality.
- (3) Solving problems – each student solves the problem created by another student.
- (4) Evaluating solution #1 – the solution is evaluated by the problem creator who assigns a grade to the solution.
- (5) Evaluating solution #2 – the solution is assigned a grade by a student who is neither the problem creator nor the problem solver (*One final grade is calculated here depending upon the grading system chosen by the instructor which can be highest of the two grades, average of the two grades, lowest of the two grades etc.*)
- (6) Registering dispute (*optional*) – the problem solver can register a dispute with the instructor if he/she is not content with the grade he/she receives.
- (7) Dispute Arbitration – the dispute registered by the student is arbitrated by the instructor.

1.8 Comparing Peer Assessment Techniques with PLA

Table 1.2 compares the characteristics of the Networked innovative assessment procedures, Goldfinch's peer assessment technique and PLA to highlight the difference in the PLA model with other peer assessment techniques.

Table 1.2 Comparisons of Peer Assessment Techniques with PLA

Characteristics	Networked innovative assessment procedures	Goldfinch's peer assessment technique	Participatory Learning Approach(PLA)
Basic Premise	Promote learning by encouraging students to compare their work with the work of their peers.	Build group-working skills in students through peer assessment of participation in team tasks.	Build deep understanding of concepts in students through active participation in each step of homework/exam lifecycle.
Includes self assessment	Yes	Yes	Instructor-dependent*
Includes peer assessment	Yes	Yes	Yes
Flexibility with steps	Rigid	Rigid	Flexible
Flexibility with iterations	Flexible	Only one iteration	Flexible
Supports individual working	Yes	No	Yes
Supports group working	No	Yes	Yes
Supports Anonymity	Yes	Yes	Yes
Marking criteria	Students and instructors collaboratively decide	Students and instructors collaboratively decide	Instructor-dependent*
Fixed grading range	No	Yes	No

*Instructor-dependent means that option can be included in the PLA cycle if the instructor desires. However it was not included in the PLA research conducted before.

From Table 1.2 it is apparent that PLA is a very loose, flexible model with a well defined premise. This flexibility allows the instructor to design the PLA cycle depending upon the course and students' learning objectives.

1.9 Prior PLA Research

The PLA was conducted at New Jersey Institute of Technology for either the midterm or the final exam in CIS677 for 5 semesters between Fall 1999 and Summer 2002 (Wu et al., 2004). CIS677, "Information Systems Principles," is a core course for Masters and Ph.D. students, featuring intensive writing and reading assignments. The exam consisted

of two essay questions. Both distance learning and traditional classroom sections participated with student working individually. The PLA was conducted using WebBoard, an asynchronous discussion tool. The results of the study suggested that majority of the students felt that they learned throughout the process and the process was an enjoyable one (Wu et al., 2004, Shen et al., 2004a). The following PLA process was adopted:

- (1) The instructor set up “conferences” or “threaded discussion areas” on the WebBoard computer conferencing system, and assigned each student two problem IDs. (“Threading” displays a posting grouped with all of its corresponding replies.)
- (2) Students designed their two examination problems and posted them on WebBoard under its assigned problem ID.
- (3) The instructor approved the problems, editing them if necessary, decided which students should solve each (ensuring that students in project teams did not solve each other’s questions), and then emailed each student with the IDs of the two problems they should solve.
- (4) Students posted solutions as a “threaded reply” to the corresponding problem on WebBoard.
- (5) Students evaluated and graded the solution to each problem they authored, following detailed evaluation guidelines requiring 2 sentences justification for each of 6 sub-criteria (such as degree of synthesis, use of class readings, etc.). They posted evaluations and grades as a “threaded reply” to the corresponding solution on WebBoard.

- (6) Students were emailed two additional problems to evaluate, providing a second opinion.
- (7) If the grades were within 10 points, the instructor used the higher of the two grades. Otherwise he/she re-graded the solution using the two student evaluations for assistance. This final grade was posted as a “threaded reply”.
- (8) Students could optionally dispute their grade, in which case they had to evaluate and grade their own solution using the same evaluation guidelines, posting their disputes as a “threaded reply”.
- (9) The instructor determined and posted a final grade for the disputed solution.

The entire PLA process was conducted using an asynchronous discussion tool. A lot of steps had to be done manually by the instructor which presented an additional overhead, such as assigning problems to students, emailing student reminders, ensuring that 2 students are not assigned the same problem, ensuring that the same student does not get incompatible roles within the same workflow instance, like problem creator and problem solver; and deciding the final grade for each student. Students further had to ensure that they post their problems/solutions as a message under the correct thread, or else the instructor had to physically move that message to correct thread.

Considering the amount of manual work involved for the instructor, it would be very helpful to have software to facilitate and control the whole process. Ideally, the software should allow the instructor to set up the whole process initially and then overview the process as it happens and make necessary changes as and when desired. This would reduce a considerable amount of load off the instructor without any compromise in the quality of teaching or learning objectives.

It should further streamline the procedures and provide guidance for the students, thus reducing cognitive overload for them as well.

1.10 PLAE: Participatory Exercise Management Tool

Absence of existing software which can handle the PLA process to its full extent provides the motivation to develop one which can not only handle the PLA process but also reduce the overhead on the instructor allowing him/her to adopt a more qualitative role than a quantitative one. With this lack of overhead the instructor can spend more time to identify students with poor performance early and provide them with the required guidance and support to overcome their problems (Davies, 2000).

The Participatory Learning Approach Environment (PLAE), a participatory exercise management tool, is designed specifically to handle the PLA process and ease the additional burden off the instructor. The main design goals for PLAE include, maximizing student learning, minimizing student and instructor overhead in conducting PLA and be easy to use. The PLAE will guide the students and instructors at each step and automate monotonous tasks for the instructor once it is set up correctly. It will provide a set of tools to the instructor to manage the PLA process and at the same time be easy to use.

1.11 Explanation of PLAE Terms

The next section will describe the PLA example as it would be implemented using PLAE, but before doing that, it is necessary to define some terms and concepts which the PLAE

uses. These terms would be used often in the subsequent chapters; hence it is important to understand them.

(1) Assignment:

An assignment can be explained as the task designed by the instructor which is to be completed by the student by the assigned due date. An assignment can be made up of one to many artifacts. An example of an assignment can be homework, midterm, final exam, etc.

(2) Artifact:

An artifact can be defined as a deliverable within an assignment. For example, a problem, a solution or an evaluation within an assignment. It is not necessary that the artifact has to be a deliverable by the student; it can be from the instructor as well. For example, a problem review by the instructor to ensure the quality of the problem after the student has submitted a problem.

(3) Discussion area:

Discussion area allows the student or the instructor to interact among themselves by the means of asynchronous discussion. Student or instructor posts messages in a discussion area which can be read by other participants. Students or instructors can reply to messages posted or upload a file to the discussion area.

(4) Task area:

Participants post their artifacts to the system under the discussion area allotted for an assignment. Such discussion areas are called task areas.

(5) Template:

A template is a pre-formatted structure which assists the participant in submitting the artifact correctly to the system.

(6) Template set:

A template set is a set of templates that will be used by the artifacts of an assignment. A template set is associated with the assignment and templates within the set are associated with artifacts of the assignment.

(7) Problem:

A problem is an artifact type representing the problem submitted by a student. Some examples of problems would be essay questions, math problems, multiple choice questions etc. It is a part of an assignment.

(8) Solution:

A solution is an artifact type representing the solution to the problem submitted by a student. It is a part of an assignment.

(9) Evaluation:

An evaluation is an artifact type representing the evaluation of another artifact. Generally an evaluation is done for a solution artifact, however it is not necessary. An evaluation can also be done for a review artifact. It is a part of an assignment.

(10) Dispute:

A dispute is an artifact type representing the dispute submitted by the student. It is a part of an assignment.

(11) Arbitration:

Arbitration is an artifact type representing the arbitration submitted by the instructor or student. It is a response to the dispute artifact. It is a part of an assignment.

(12) Manual grade:

Manual grade is an artifact type representing the manual grading done by the instructor or student if the evaluation grades are too close to calculate final grade automatically. It is a part of an assignment.

(13) Review:

Review is an artifact type representing the review done by the instructor or student of another artifact. An example of a review artifact would be review of the problems submitted by students to check for quality. It is a part of an assignment.

(14) Workflow:

Workflow is a process with specific start and end dates with one or many steps which have their own start and end dates within the process. In PLAE, workflow is created when an assignment is created. The artifacts within the assignment are the steps that make up a workflow.

(15) Workflow instance:

Workflow instance is a copy of a workflow with the same attributes like start date, end date, artifacts within the assignment, etc. In PLAE, one workflow instance is created for each student in the course. For example, if there are 30

students in the course, for every assignment created in that course, 30 workflow instances would be created for each student.

(16) Workflow step:

A workflow is made up of many steps each having a set of attributes such as start date and end date. In PLAE, artifacts represent the steps within a workflow.

(17) Subjective question:

Subjective question refers to a question that does not have a specific answer. The answer for such a question depends upon the various skills such as question interpretation, knowledge synthesis and articulate written communication. Examples of subjective questions would be essay type questions, short answers, executive summary of a project, etc.

(18) Objective question:

Unlike a subjective question, an objective question has a specific answer. The answer is either correct or incorrect. An example of an objective question would be a question with multiple answers being provided and asking a student to select the correct answer.

1.12 Relationship between Assignments, Artifacts and Template Sets

Figure 1.1 gives a visual representation of the relationship between assignment, artifacts and template sets. As shown in Figure 1.1, two assignments have having seven artifacts each have been associated to a template set. Once an assignment and template set are associated, the artifacts within the assignment are automatically associated with their matching templates within the template set.

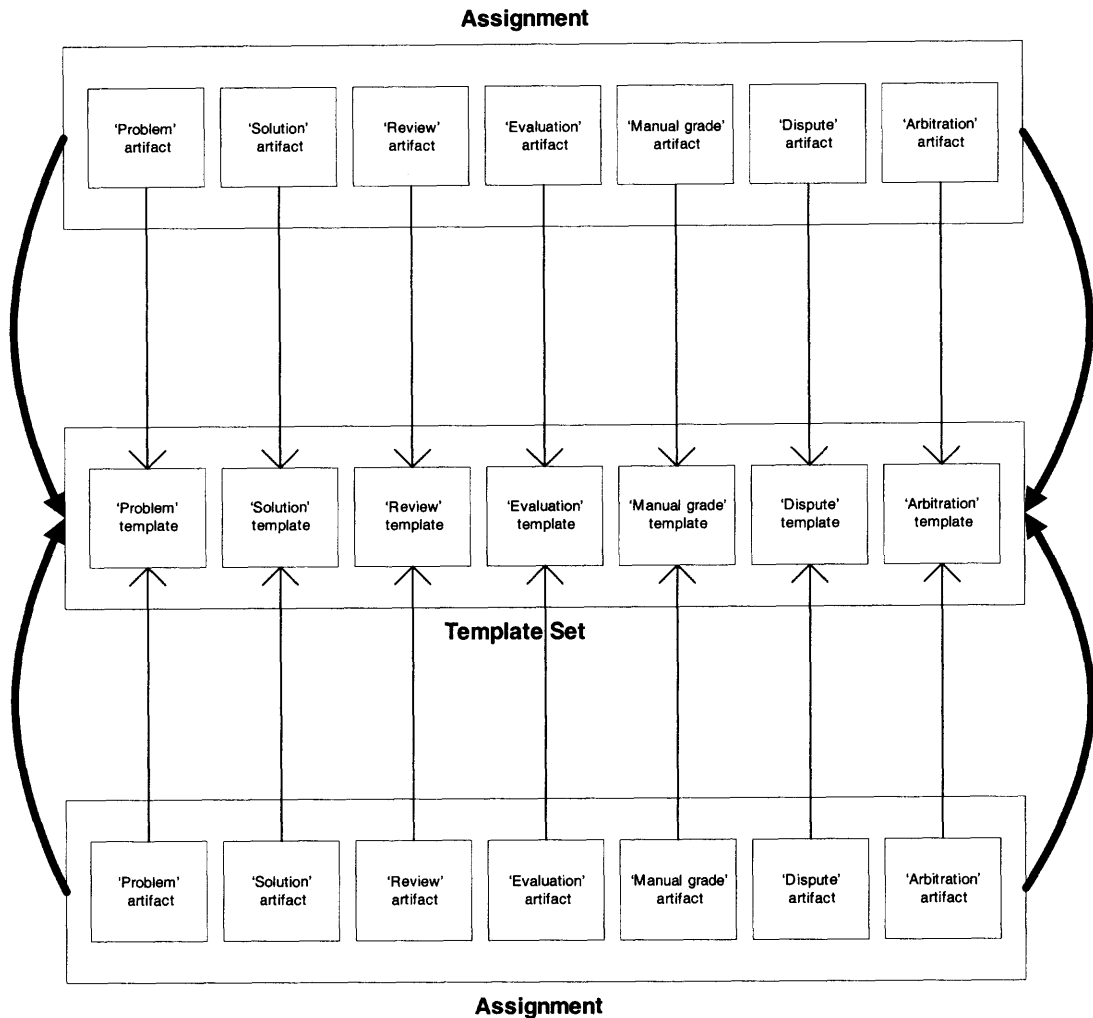


Figure 1.1 Assignment-artifact-template set relationship.

Figure 1.2 shows a variation in the relationship between assignments, artifacts and template sets. In Figure 1.2 the top assignment has seven artifacts and the bottom assignment has six artifacts, two artifacts each for a problem, a solution and an evaluation. Both assignments use the same template set. It is not necessary that an assignment should have all seven artifact types to use a template set.

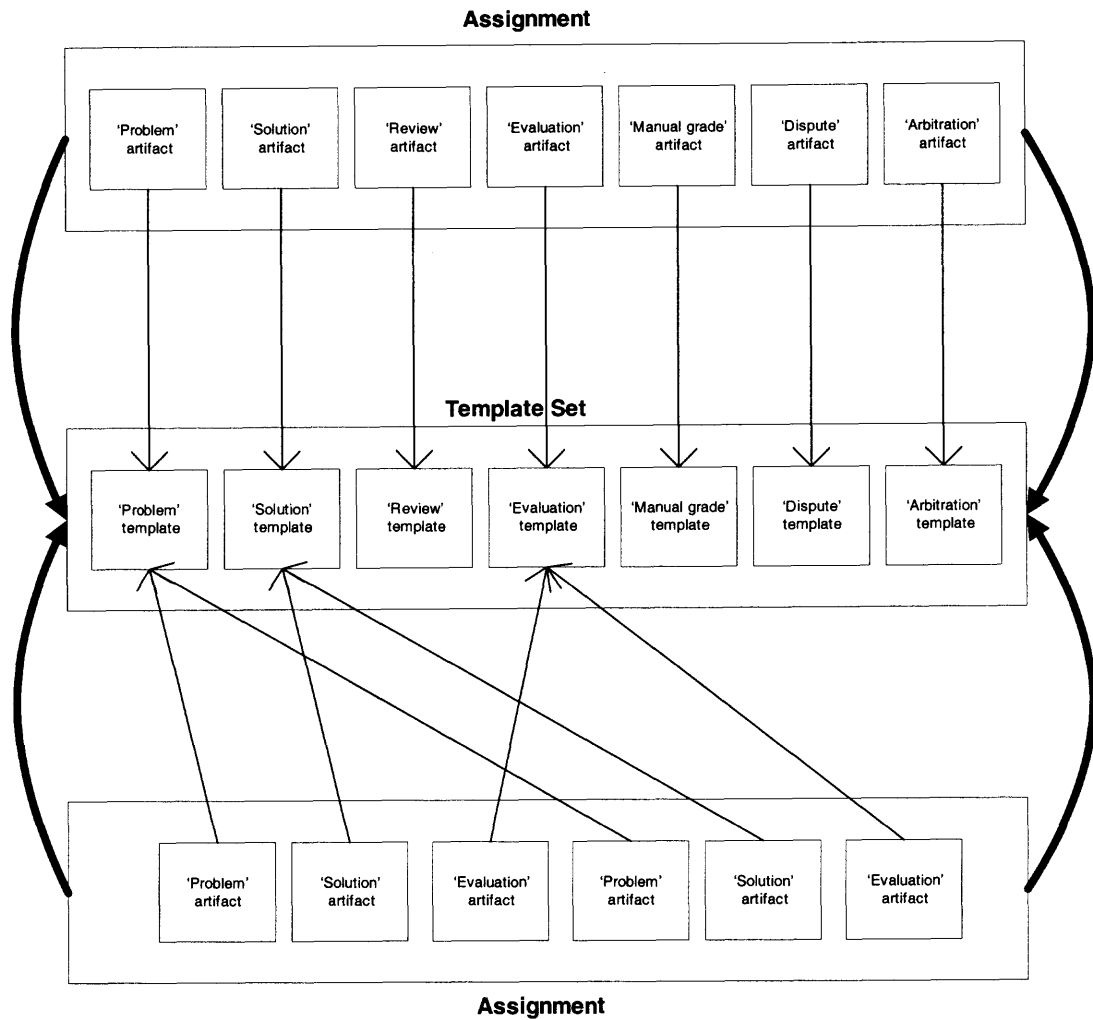


Figure 1.2 Variation in assignment-artifact-template set relationship.

1.13 PLAE Example

- (1) The instructor sets up “discussion areas” for students to interact with each other and “task areas” for each assignment.
- (2) The instructor sets up each artifact within an assignment.
- (3) The PLAE will assign each student the responsibility to create 2 problems (*automated*).

- (4) Students will submit their problems under the respective assignment in the appropriate task area for that problem using the “problem template” as a guide.
Each problem is an artifact.
- (5) The instructor approves each problem in its own task area, editing it if necessary.
Problem review is an artifact.
- (6) The PLAЕ will assign students to solve those problems, ensuring that students in project teams do not solve each others’ questions (*automated*). The solution is an artifact.
- (7) Students post their solutions using the “solution template” which places the solution under the correct corresponding problem.
- (8) The PLAЕ will assign students to evaluate the solutions, ensuring that students who solved the problems did not evaluate their own solution (*automated*).
- (9) Students will post their evaluations for a solution using the “evaluation template”.
- (10) The PLAЕ will calculate the final grade depending upon the grading system selected by the instructor and post the final grade in its problem’s task area (*automated*).
- (11) Students can register a grade dispute using the “dispute template” in its problem’s task area.
- (12) The instructor will arbitrate the dispute and provide a decision using the “arbitration template”.

All the steps carried out by the instructor in the manual process are *automated* with the use of PLAE. This would give an idea of the amount of workload that has been taken off the instructor. For students too, it is much easier to submit artifacts since PLAE provides templates and guides the students to the correct screens to submit their artifacts.

CHAPTER 2

REQUIREMENT ANALYSIS FOR PLAE VERSION 1.0

This chapter analyzes and presents the features of the PLAE version 1.0. The features selected for this version are the basic functional requirements (system capabilities required to meet the needs of end users) which will help the instructor conduct the process of PLA successfully without increasing the workload on the instructor. The selected features would also guide students throughout the PLA process with as little cognitive overload as possible. The requirements gathered below are based on the experiences of an instructor who conducted the PLA process using a normal conferencing system.

The features mentioned below describe the graphical elements on the screen visible to the user as well as processes invisible to the user. The requirements mentioned below are language-independent and technology-independent; however they are based on the assumption that the entire system would be web-based. The requirements for PLAE version 1.0 are written envisioning the use of the system by 4 users: Student, Instructor, Administrator and Experimenter. A ‘student’ user’s interaction with the system involves participating in asynchronous online discussions, artifact submission and their own profile management. An ‘instructor’ user interacts with the system to manage assignments, artifacts, templates as well as participate in asynchronous online discussions and the overall management of the course. The ‘administrator’ user is responsible for managing courses, instructors and the overall maintenance of the system. The PLAE system will be released with a default user already present in the system called ‘administrator’. An ‘experimenter’ user is interested in monitoring the overall PLA

process and collecting feedback from students and instructors. He/she can browse through the tools, read messages, artifacts and conduct surveys. The following sections of the chapter are organized by dividing the PLAE system into its main functionalities and explaining features within each functionality. Any additional explanation for a feature is provided in *italics*.

2.1 Login to PLAE

This is a core step for any user before he/she can use the PLAE system. The system will display a set of options for administrators, instructors and students relevant to their roles.

The user will be able to login to the system by entering his/her username and password on the homepage of the PLAE system. After successful authentication of the login credentials the system will display the appropriate options depending upon the type of user logging in. This functionality is used by four users: administrator, instructor, student, and experimenter.

The functionality is detailed below:

- (1) The system will capture the following information from the user:
 - Username
 - Password
- (2) The system will restrict the username to twenty (20) alphanumeric characters.
- (3) The system will restrict the password to twenty (20) alphanumeric characters and ensure the password entered by the user is in an unreadable format on the screen.
- (4) The system will display the message “Please enter your username” and move focus back to the ‘Username’ field if it is left blank.

- (5) The system will display the message “Please enter your password” and move focus back to the ‘Password’ field if it has been left blank.
- (6) The system will display the message “Invalid username/password pair, please re-enter.” and clear the password field and move focus back to the ‘Username’ field in the event of unsuccessful authentication of the username/password pair.
- (7) The system will, after successful authentication, confirm from the user the role (administrator, student, instructor) he/she would like to assume in the event that username has multiple roles assigned to him/her.

For ‘student’ role

- (8) The system will display a list of selectable list of courses he/she has admission to and a non-selectable list of courses to which he/she does not have admission.
- (9) The system will provide a visual indication to the student to differentiate the courses to which he/she does/does not have admission to.
- (10) The system will provide a visual indication next to each course for which the student has any pending task.
- (11) The system will display an option ‘Request admission’ and the instructor’s name next to courses in which the student does not have permission to participate. (Refer to 2.9.1)

For ‘instructor’ role

- (12) The system will display a selectable list of courses which the instructor is teaching.
- (13) The system will provide a visual indication next to each course for which the instructor has any pending task.

(14) The system will provide a visual indication next to each course for which the instructor has any pending admission requests

For 'administrator' role

(15) The system will display the following options:

- School Management (Refer to 2.2)
- Instructor Management (Refer to 2.3)
- Experimenter Management (Refer to 2.4)
- Course Management (Refer to 2.5)

For 'experimenter' role

(16) The system will allow an experimenter to go through discussion and task areas, read messages and artifacts.

(17) The system will allow an experimenter user to read messages and artifacts even if they are deleted and not visible to other users.

(The experimenters are allowed to see the deleted messages and artifacts too as it may be useful for them to understand the feedback of students and instructors for the PLA process.)

(18) The system will display a different visual style to show deleted messages and artifacts.

2.2 School Management

This functionality can only be used by the administrator. He/she can manage all participating schools/colleges/universities using this functionality.

2.2.1 View Schools

After the administrator selects 'School Management' he/she will view all the schools registered in the system.

- (1) The system will display a list of registered schools sorted alphabetically.
- (2) The system will display the following information for each school: school name, school address and contact person for the school.
- (3) The system will display a 'Edit' option next to each school.
- (4) The system will display a 'Delete' option next to each school.
- (5) The system will display the option 'Add new school' below the list of schools.

2.2.2 Add New School

This feature allows the administrator to add a new school to the system. The administrator will have to provide the information mentioned below to successfully add a school.

- (1) The system will display a page to capture the following options after the system administrator selects 'Add new school' option:
 - School name
 - Street address (Optional)
 - City
 - State

- Zip/pin code (Optional)
 - Country
- (2) The system will display the message “Please enter the name of the school/college/university” and move focus to the school name field if the system administrator has left the school name field blank.
 - (3) The system will ensure that school name is not more than two hundred and fifty (250) alphabetic characters.
 - (4) The system will display the message “Please enter the city where school is located” and move focus to the school city field if the system administrator has left the school city field blank.
 - (5) The system will ensure that the school city is not more than one hundred (100) alphabetic characters.
 - (6) The system will display the message “Please enter the state where the school is located” and move focus to the school state field if the system administrator has left the school state field blank.
 - (7) The system will ensure that the school state is not more than one hundred (100) alphabetic characters.
 - (8) The system will display a list of all countries, allowing the administrator to select one.
 - (9) The system will display the message “Please enter the country of the school” and move focus to the country field if the system administrator has left the country field blank.

- (10) The system will display an option 'Add contact person' to add contact details of a person associated with the school (Refer to 2.2.3).
- (11) The system will display a confirmation message after successfully adding the school to the database, after submission of the information by the administrator.

2.2.3 Add New Contact Person

This feature allows the administrator to add a new contact person associated with the school.

- (1) The system will capture the following information iteratively every time the administrator selects 'Add contact person':
 - Name
 - Email
 - Phone
- (2) The system will display the message "Please enter the name of the contact person" and move focus to the name field if the system administrator has left the name field blank.
- (3) The system will display the message "Email or phone number has to be entered" if the system administrator leaves both email and phone fields blank.

(Administrator will have to enter either the email or phone. Both fields cannot be left blank.)
- (4) The system will restrict the length of the email to one hundred (100) alphanumeric characters.
- (5) The system will restrict the length of the phone to fifteen (15) numeric characters.

2.2.4 Edit School Details

This feature allows the administrator to edit details of a school already present in the system. The system will allow the administrator to edit all the information mentioned in 2.2.2(1).

2.2.5 Delete School

This feature allows the administrator to delete a participating school already present in the system.

- (1) The system will confirm with the administrator before deleting the school from the system.
- (2) The system will not delete the entities (instructors, courses, students) related to the school but just make them invisible to other users.

(The entities related to the school will not be physically deleted from the system but it will not be visible in any of the system functionalities like student registration, course management and instructor management. They will be retained for research purposes.)

2.3 Instructor Management

This feature enables the administrator to create instructors within the system. The administrator can add a new instructor, modify and delete an existing instructor. This feature will also enable the administrator to assign the role of an instructor to an already existing student.

2.3.1 View Instructors

The administrator will see a list of instructors already present in the system after he/she selects 'Instructor Management'.

- (1) The system will display a list of registered instructors sorted alphabetically on last name basis.
- (2) The system will display 'Edit' 'option next to each instructor.
- (3) The system will display 'Delete' option next to each instructor.
- (4) The system will display the option 'Add new instructor' below the list of instructors.
- (5) The system will display option 'Assign instructor role to student' which will allow the administrator to assign a student who is already a user in the system, the role of an instructor.

2.3.2 Add New Instructor

This feature allows the administrator to add a new instructor to the system. The administrator will have to provide the information mentioned below to successfully add the instructor.

- (1) The system will display a page to capture the following options after the system administrator selects 'Add new instructor' option:
 - First name of the instructor
 - Middle initials of the instructor (optional)
 - Last name of the instructor
 - School/College/University of the instructor
 - Username for the instructor
 - Email id of instructor

- Password for instructor login
- (2) The system will display the message “Please enter the instructor’s first name” and move focus to the first name field if the system administrators has left the first name of the instructor field blank.
 - (3) The system` will ensure that the first name of the instructor is not more than fifty (50) alphabetic characters.
 - (4) The system will restrict that the middle initials of the instructor to ten (10) alphabetic characters.
 - (5) The system will display the message “Please enter instructor’s last name” and move focus to the last name field if the system administrator has left the last name of the instructor field blank.
 - (6) The system will ensure that the last name of the instructor is not more than fifty (50) alphabetic characters.
 - (7) The system will display a list of all registered school names, allowing the administrator to choose multiple schools.

(It is possible that an instructor be associated with more than one school.)
 - (8) The system will display the message “Please enter the instructor’s email id” and move focus to the email field if the system administrator has left the email field blank.
 - (9) The system will ensure that the email of the instructor is not more than hundred (100) alphanumeric characters.

- (10) The system will display the message “Please enter instructor’s username” and move focus to the last name field if the system administrator has left the username for the instructor blank.
- (11) The system will ensure that the username of the instructor is not more than twenty (20) alphanumeric characters.
- (12) The system will display the message “Please enter the password which the instructor will use to login” and move focus to the password field if the system administrator has left the password field blank.
- (13) The system will ensure that the password of the instructor is not more than twenty (20) alphanumeric characters.
- (14) The system will confirm that the username specified by the instructor is not being used by any other user in the system.
- (15) The system will add the instructor to the database upon submission by the system administrator and display a confirmation message.
- (16) The system will send an email to the instructor with the username and password which can be used to login to the system.

2.3.3 Edit Instructor

The administrator can edit the details of an instructor already present in the system using this feature. The administrator can edit all details mentioned in 2.3.2(1) including the password of the instructor. The system will allow the administrator to change the password of the instructor in the event that the instructor forgets his/her password and wants it to be reset.

2.3.4 Delete Instructor

This feature allows the administrator to delete an instructor from the system.

- (1) The system will ask for a confirmation after the system administrator selects the 'Delete' option before deleting the instructor's details from the database.
- (2) The system will display a confirmation message to the system administrator after successful deletion of the instructor from the database.

2.3.5 Assign Instructor Role to Student

A student can also assume the role of an instructor and teach a class. For example, Ph.D. students or T.A.'s (Teaching Assistants). This feature allows the administrator to assign the role of an instructor to any student already present in the system.

- (1) The system will display the last names of each alphabet on a separate page to allow easy browsing of the student names.
- (2) The system will allow the administrator to select the alphabet within which he would like to locate the name of the student.
- (3) The system will, optionally, accept the name of the student and allow the administrator to search the name of the student.
- (4) The system will allow the administrator to select the student to which he would like to assign the role of an instructor.
- (5) The system will inform the administrator after the successful assignment of the instructor role to the selected student.

2.4 Experimenter Management

This feature enables the administrator to create experimenters within the system. The administrator can add a new experimenter or modify and delete an existing experimenter. Experimenters monitor and supervise the PLA process by reading messages, artifacts and conducting surveys. They can browse through all courses and read messages and artifacts even if they are deleted and not visible to other users.

2.4.1 View Experimenters

The administrator will view the list of all experimenters registered in the system after he/she selects 'Experimenter Management'.

- (1) The system will display a list of registered experimenters sorted alphabetically on last name basis.
- (2) The system will display the following information for each experimenter: last name, first name, school affiliated with and username of experimenter.
- (3) The system will display 'Edit' option next to each experimenter.
- (4) The system will display 'Delete' option next to each experimenter.
- (5) The system will display the option 'Add new experimenter' below the list of experimenters.

2.4.2 Add New Experimenter

This feature allows the administrator to add a new experimenter to the system. The administrator will have to provide the information mentioned below to successfully add a new experimenter.

- (1) The system will display a page to capture the following options after the system administrator selects 'Add new experimenter' option:
 - First name of the experimenter
 - Middle initials of the experimenter (optional)
 - Last name of the experimenter
 - School/College/University affiliation
 - Username of the experimenter
 - Email id of the experimenter
 - Password for the experimenter login
- (2) The system will display the message "Please enter experimenter's first name" and move focus to the first name field if the system administrator has left the first name of the experimenter field blank.
- (3) The system will ensure that the first name of the experimenter is not more than fifty (50) alphabetic characters.
- (4) The system will ensure that the middle initials of the experimenter are not more than ten (10) alphabetic characters.
- (5) The system will display the message "Please enter the experimenter's last name" and move focus to the last name field if the system administrator has left the last name of the experimenter field blank.
- (6) The system will ensure that the last name of the experimenter is not more than fifty (50) alphabetic characters.
- (7) The system will display a list of all registered school names, allowing the administrator to choose one.

- (8) The system will display the message “Please enter the experimenter’s email id” and move focus to the email field if the system administrator has left the email field blank.
- (9) The system will ensure that the email of the experimenter is not more than one hundred (100) alphanumeric characters.
- (10) The system will display the message “Please enter the experimenter’s username” and move focus to the last name field if the system administrator has left the username for the experimenter blank.
- (11) The system will ensure that the username of the experimenter is not more than twenty (20) alphanumeric characters.
- (12) The system will display the message “Please enter the password which the experimenter will use to login” and move focus to the password field if the system administrator has left the password field blank.
- (13) The system will ensure that the password of the experimenter is not more than twenty (20) alphanumeric characters.
- (14) The system will confirm that the username specified by the experimenter is not being used by any other user in the system.
- (15) The system will add the experimenter to the database after submission by the system administrator and display a confirmation message.
- (16) The system will send an email to the experimenter with the username and password which can be used to login to the system.

2.4.3 Edit Experimenter

Administrator can edit the details of an experimenter already present in the system by using this feature. The administrator can edit all details mentioned in 2.4.2(1) including the password of the experimenter. The system shall allow the administrator to change the password of the experimenter in the event that the experimenter forgets his/her password and wants it to be reset.

2.4.4 Delete Experimenter

This feature allows the administrator to delete an experimenter from the system.

- (1) The system will ask for confirmation after system administrator selects 'Delete' option before deleting the experimenter's details from the database.
- (2) The system will display a confirmation message to the system administrator after successful deletion of the experimenter from the database.

2.5 Course Management

This feature enables the administrator to create courses within the system. The administrator can add a new course, modify/delete an existing course and assign an instructor to a course. This functionality can only be used by the administrator.

2.5.1 View Courses

The administrator will view all courses in the system after he/she selects 'Course Management'

- (1) The system will display a list of registered courses sorted alphabetically and grouped according to semester.

- (2) The system will display the following information: course name, number, section, school and instructor.
- (3) The system will display 'Edit' and 'Delete' options next to each course.
- (4) The system will also display the option 'Add new course' to allow the administrator to add a new course.

2.5.2 Add New Course

The administrator can add a new course in the system that wishes to use the PLA in the class. The administrator can add different sections for each course that he/she adds.

- (1) The system will display a page to capture the following options after the system administrator selects 'Add new course' option:
 - School/college/university where the course is offered
 - Course number
 - Course section
 - Course name
 - Instructor
 - Semester
- (2) The system will display a list of all registered school names, allowing the administrator to choose one.
- (3) The system will display the message "Please enter course number" and move focus to the course number field if the system administrator has left the course number field blank.
- (4) The system will ensure that the course number is not more than twenty (20) alphanumeric characters.

- (5) The system will display the message “Please enter section id” and move focus to the section id field if the system administrator has left the section ID field blank.
- (6) The system will ensure that the section ID is not more than twenty (20) alphanumeric characters.
- (7) The system will display the message “Please enter course name” and move focus to the course name field if the system administrator has left the course name field blank.
- (8) The system will ensure that the course name field is not more than one hundred (100) alphanumeric characters.
- (9) The system will display a list of all instructors registered in the system allowing the system administrator to select one instructor.
- (10) The system will display the message “Please select the instructor for the course” and move focus to the instructor field if the system administrator has not selected the instructor for the course.
- (11) The system will allow the administrator to enter a semester.
- (12) The system will display the message “Please enter the semester” and move focus to the semester field if the system administrator has left the semester field blank.
- (13) The system will ensure that the semester field is not more than fifty (50) alphanumeric characters.
- (14) The system will ensure that the same course id with same course section in the same semester is not already present in the system.
- (15) The system will add the course details to the database and display a confirmation message to the system administrator.

2.5.3 Edit Course

The administrator can edit the course details for course already present in the system.

The administrator can edit all the information mentioned in 2.5.2(1).

2.5.4 Delete Course

The administrator can delete an existing course from the system. A probable reason for deleting a course may be if the course is not going to be taught any longer.

(1) The system will ask for confirmation after system administrator selects 'Delete' option before deleting the course details from the database.

(2) The system will not delete any templates associated with the course.

(In the PLAE version 1.0 since templates are predefined, they cannot be deleted because other courses might be using them. In the PLAE version 2.0 where customized templates are created by the instructor, templates will not be deleted because the instructor might be using the same template in another course.)

(3) The system will display a confirmation message to the system administrator after successfully deleting the course details from the database.

2.6 Course Options

Different options will be displayed to a student or instructor once he/she selects a course after successful login. For the student role, this will be a course in which he/she is participating as a student and for the instructor role this would be a course which he/she is teaching.

2.6.1 Select a Course

After successful login, the system will display the list of courses the user is currently registered for. The user will select one course in which he/she would wish to participate.

- (1) The system will allow the user to select the course he/she wishes to participate in.
- (2) The system will display the following elements for a student/instructor role before he/she selects the course to enter:
 - A 'Logoff' option (Refer to 2.6.2)
 - A 'Help' option (Refer to 2.6.3)
 - A 'Change Password' option (Refer to 2.6.5)
- (3) The system will display the following elements for a student/instructor role after he/she selects the course to enter:
 - Discussion/Task areas created by the instructor
 - A 'Logoff' option (Refer to 2.6.2)
 - A 'Help' option (Refer to 2.6.3)
 - A 'Mark Read' option (Refer to 2.6.4)
 - A 'Change Password' option (Refer to 2.6.5)
 - A Visual indication for pending tasks (Refer to 2.6.6)
 - An 'Administer' option (Only for instructor) (Refer to 2.6.7)
 - New Admission Requests (Only for instructor) (Refer to 2.9.3)

2.6.2 Logoff (For All Users)

This feature allows the user to close all sessions and logout of the system.

- (1) The system will close all sessions for the user and display the login page once the user selects 'Logoff'.
- (2) The systems will logoff the user automatically after twenty (20) minutes of no activity.

2.6.3 Help (For All Users)

The system would provide an online help document which the user can refer to in case he/she needs quick answers to questions concerning the usability of the system. The system will display an online manual of the system once the user selects the 'Help' option.

2.6.4 Mark Read

This feature allows the user to mark the messages in a discussion/task area as 'read'. Marking 'read' implies visual indication for all unread messages would be removed and it would appear only when a new message is posted.

- (1) The system will display a list of task/discussion areas with the number of unread messages in each area.
- (2) The system will list all artifact types with the number of unread artifacts for each type.
- (3) The system will allow the user to choose the area or the artifact type which he/she wants to mark as 'read'.

- (4) The system will update the status of the messages or artifacts as 'read' in the database.

2.6.5 Change Password (For All Users)

This feature allows the user to change his/her current password. The user will have to enter his/her current password and the new password which he/she desires twice as a confirmation measure.

- (1) The system will capture the following information from the user:
 - Current password
 - New password
 - Repeat new password
- (2) The system will display a message "New passwords entered do not match" if the new passwords entered by the user are not same.
- (3) The system will display a message "Incorrect old password" if the old password entered is not same with the password stored in the database.
- (4) The system will update the old password in the database with the new password and display a confirmation message to the user.

2.6.6 Visual Indication for Pending Tasks

The system will display a constant visual indication to indicate if the user has any pending tasks in the current course or in any other course for which he/she is registered. Both visual indications will be visually different allowing the user to know if he has any pending tasks by just looking at them. For example, orange color may be used indicate

pending tasks in the current course and green color to indicate pending tasks in other courses.

- (1) The system will display a constant visual indication (course pending tasks) if there are pending tasks or unlocked but submitted artifacts for the user in the current course.
- (2) The system will display a different constant visual indication (all pending tasks) if there are pending tasks for the user in the other courses for which the user is registered.
- (3) The system will allow the user to select the 'course pending tasks' visual indication allowing him/her to view all the tasks that are remaining.
- (4) The system will allow the user to select any one of the tasks from the task list to take him/her directly to that task.

2.6.7 Administer (Only for Instructor)

This feature provides the instructor with various options and tools to manage the current course. The options are listed and explained below.

The system will display the following options once the instructor selects the 'Administer' option:

- Discussion Area Management (Refer to 2.1.8.7)
- Admission Controller (Refer to 2.1.9.2)
- Template Management (Refer to 2.1.12)
- Student Allocation Tool (Refer to 2.1.14.1 and 2.1.14.2)
- Artifact Management (Refer to 2.1.11)
- Assignment Management (Refer to 2.1.10)

- Workflow Viewer (Refer to 2.1.13.2)
- View Final Grades (Refer to 2.1.16)
- Reminder Preferences (Refer to 2.1.15)

2.6.8 Edit Profile (Only for Student)

This feature allows the student to edit his/her personal information described in 2.1.7.1(1). The student can edit all the information except his/her username.

2.7 Student Registration

Students are allowed to register themselves in the PLAE system using this feature. After registering themselves the students can select the course to which they desire admission and request permission from that course instructor. This feature can also be used by instructors to create a student account for themselves while retaining their instructor profile. The student can create a new profile for himself/herself by using the option 'New Student Registration' from the home page of the PLAE system.

An instructor registered in the system can also create a student profile for himself/herself if he/she wishes to participate in a course as a student.

2.7.1 New Student Registration

This feature will allow a user to create a student account with the system which will allow him/her to request admission for a course. The user will have to enter the information mentioned below for successful registration.

- (1) The system will display an electronic form to capture the following options after the user selects 'New Student Registration' option:

- First name of the student
 - Middle initials of the student (optional)
 - Last name of the student
 - School/college/university of student
 - Desired username (optional)
 - Primary email id of the student
 - Secondary email id of the student (optional)
 - Password for the student login
 - Residence phone number (optional)
 - Mobile phone number (optional)
 - Office phone number (optional)
- (2) The system will display the message “Please enter student’s first name” and move focus to the first name field if the user has left the first name of the student field blank.
- (3) The system will ensure that the first name of the student is not more than fifty (50) alphabetic characters.
- (4) The system will ensure that the middle initials of the student to ten (10) alphabetic characters.
- (5) The system will display the message “Please enter student’s last name” and move focus to the last name field if the user has left the last name of the student field blank.
- (6) The system will ensure that the last name of the student is not more than fifty (50) alphabetic characters.

- (7) The system will display a list of all registered school names, allowing the student to choose multiple schools.
- (8) (It is possible that one student can be registered at multiple schools)
- (9) The system will display the message “Please enter student’s primary email id” and move focus to the email field if the user has left the email field blank.
- (10) The system will ensure that the email of the student is not more than one hundred (100) alphanumeric characters.
- (11) The system will display the message “Please enter the password which the student will use to login” and move focus to the password field if the user has left the password field blank.
- (12) The system will ensure that the password of the student is not more than twenty (20) alphanumeric characters.
- (13) The system will ensure that the student is not already registered in the system by checking for duplicate e-mail ids.
- (14) The system will ensure that the entered ‘desired username’ by the user is not already being used by another user.
- (15) The system will display an error message “Username not available. Please suggest another username” and move the focus back to the ‘Desired username’ field.
- (16) The system will ensure that username of the student is not more than twenty (20) alphanumeric characters.
- (17) The system will generate a username automatically if the user has not specified a username.

- (18) The system will add the student to the database after submission by the user and display a confirmation message.
- (19) The system will send an email to the student informing him/her about their username and password to login to the system.

2.7.2 Instructor Creating a Student Profile

A user who is already registered in the system as an instructor can create a student profile for himself/herself. For example, if an instructor creates a student profile for himself/herself, he/she can participate as a student in a course.

- (1) The system will capture the following information from the user:
 - Instructor username
 - Instructor password
- (2) The system will authenticate the instructor username and password within the system.
- (3) The system will, upon successful authentication, inquire the user 'Do you want to create a student profile in addition to your instructor profile?' allowing the user to select 'Yes' or 'No'.
- (4) The system will create a student profile for the instructor after an affirmative response from the system.
- (5) The system will display a confirmation message after successful creation of the instructor's student profile.
- (6) The system will return to the homepage upon a negative response from the user.

2.8 Discussion Management

Students and instructors can have an online asynchronous discussion using this feature. Instructors are also allowed to create discussion areas within which the students can post and reply to messages. This feature allows the students to interact with each other and the instructor and share their knowledge, questions, answers or comments. All the communication is asynchronous.

Students and instructors can start participating in a discussion immediately after successful login. They can select the discussion area within which they would like to participate by posting a message, replying to a message or reading a message. Instructors can create new discussion areas by selecting ‘Discussion Area Management’ under the ‘Administer’ option.

2.8.1 Post Message

The user can post a new message in the discussion area which can be read by other participants of the course.

- (1) The system will display all the current discussion areas and the number of unread messages in that discussion area after successful login by the user.
- (2) The system will allow the user to select the discussion area in which the user wishes to post a message.

Once Within a Discussion Area

- (3) The system will display an option to post a message which when selected would allow the user to enter the message to be posted.
- (4) The system will capture the following information from the user:
 - Topic of the message

- The actual message
 - Make anonymous.
- (5) The system will display the message “Please enter the topic of the message” and move the focus to Topic field if the user has left the Topic field blank.
- (6) The system will display the message “Please enter a message” and move the focus to the Message field if the user has left the Message field blank.
- (7) The system will restrict the topic of the message to hundred (100) alphanumeric characters.
- (8) The system will allow the user to make the message anonymous by selecting the ‘Make anonymous’ option.

(A message when submitted with the anonymous option does not show the username. Even the instructor or experimenter would not be able to see the username of the message.)

- (9) The system will display the message after submission under the selected discussion area as a new thread with the following attributes:
- Username who posted the message or anonymous
 - The actual name of the user or anonymous
 - The date and time the message was posted
 - The topic of the message
 - The actual message
- (10) The system will also display the following options with the message:
- ‘Reply’ option (Refer to 2.8.2)
 - ‘Previous Topic’ option (Refer to 2.8.3)

- ‘Next Topic’ option (Refer to 2.8.4)
- ‘Mark Message Read’ option (Refer to 2.8.5)
- ‘Mark Thread Read’ option (Refer to 2.8.6)

2.8.2 Reply

The user can reply to any message posted by another user.

- (1) The system will allow the user to select the message to which he wishes to reply.
- (2) The system will copy over the original message to the reply with the name of the user who posted the message with the date and time of the message.
- (3) The system will capture the following information from the user:
 - Topic of the message
 - The actual message
 - Make anonymous (Refer to 2.8.1(8))
- (4) The system will display the message “Please enter the topic of the message” and move the focus to topic field if the user has left the topic field blank.
- (5) The system will display the message “Please enter a message” and move the focus to the message field if the user has left the message field blank.
- (6) The system will restrict the message topic to hundred (100) alphanumeric characters.
- (7) The system will display the message after submission, below the message to which it was a reply to with the following attributes:
 - Username who posted the reply or anonymous
 - The actual name of the user or anonymous
 - The date and time the reply was posted

- The topic of the reply
 - The reply message
- (8) The system will display the following options with the message:
- ‘Reply’ option (Refer to 2.8.2)
 - ‘Previous Topic’ option (Refer to 2.8.3)
 - ‘Next Topic’ option (Refer to 2.8.4)
 - ‘Mark Message Read’ option (Refer to 2.8.5)
 - ‘Mark Thread Read’ option (Refer to 2.8.6)

2.8.3 Previous Message

This feature will allow the user to view the previous message in the current thread. ‘Previous’ message implies a message before the current message in the same thread. However, if there are no more previous messages in the current thread then this feature would be disabled.

- (1) The system will display the previous message of the current message in the discussion area.
- (2) The format of the message display would be as mentioned in 2.8.1(9).

2.8.4 Next Topic

This feature will allow the user to view the next message in the current thread. ‘Next’ message implies the message after the current message in the same thread. However, if there are no more messages in the current thread then this feature would be disabled.

- (1) The system will display the next message of the current message in the discussion area.

- (2) The format of the message display would be as mentioned in 2.8.1(9).

2.8.5 Mark Message Read

This feature allows the user to mark the current message as 'read'.

- (1) The system will mark the message as 'read' for that particular user.
- (2) The system will not display the visual indication to show the message as 'unread' after the user refreshes his/her screen.

2.8.6 Mark Thread Read

This feature allows the user to mark all the messages in the current thread as 'read'.

- (1) The system will mark all the messages in the current thread as 'read' for that particular user.
- (2) The system will not display the visual indication to show the message as 'unread' for all messages in the current thread after the user refreshes his/her screen

2.8.7 Discussion Area Management (Only for Instructor)

An instructor can manage the discussion areas for his/her course by various options provided in this feature.

The system will display the following options once the instructor selects 'Discussion Area Management' under the 'Administer' option:

- Create a new discussion area (Refer to 2.8.8)
- Delete a discussion area (Refer to 2.8.9)

2.8.8 Create New Discussion Area (Only for Instructor)

This feature allows the instructor to create a new discussion area where the students/instructor can post messages.

- (1) The system will capture the name of the of the new discussion area from the instructor.
- (2) The system will restrict the length of the name of the discussion area to hundred (100) alphanumeric characters.
- (3) The system will, upon successful submission, create a new discussion area with zero (0) messages in it.

2.8.9 Delete Discussion Area (Only for Instructor)

This feature allows the instructor to delete a discussion area so that it is no longer visible to the students.

- (1) The system will display a list of all current discussion areas for that course.
- (2) The system will allow the instructor to select the discussion area which he/she wishes to delete.
- (3) The system will, after submission, hide the discussion area and all the messages within it and display a confirmation message to the instructor.
- (4) The system will not physically delete the discussion area.

(They will be stored for research purposes. They can be viewed by the experimenter.)

2.9 Admission Control

A student requires admission to participate in a course from the instructor responsible for that course. The students can request admission for a course in which he/she desires to participate using this feature. On the other hand, the instructor can grant or decline admission to a student. The instructor can, optionally, also revoke admission of a student who is already participating in the course.

A student can request admission for a course after he/she successfully logs in. The student will be provided with an option 'Request Admission' for each course for which he/she desires admission. Also, an instructor will be informed of all admission requests once he/she logs into his/her course.

2.9.1 Request Admission (Only for Student)

Once the user creates a student profile for himself/herself, he/she could request admission for a course in which he/she wishes to participate.

- (1) The system will display all courses registered in the system with the school offering them sorted using the school name, allowing the student to select a course in which he/she seeks admission.
- (2) The system will capture an optional note from the student once he/she selects the course for which he/she desires admission for.
- (3) The system will restrict the note to two hundred and fifty (250) characters.
- (4) The system will save the request to the database once the user submits the note.

2.9.2 Admission Controller (Only for Instructor)

This feature allows the instructor to manage all the admissions to his/her course. The instructor can grant or revoke admission to a student using this feature.

The system will display the following options after the instructor selects admission controller:

- Grant/Decline admission (Refer to 2.9.3)
- Revoke admission (Refer to 2.9.4)

2.9.3 Grant/Decline Admission (Only for Instructor)

This feature allows the instructor to grant or decline any new admission requests that have been made by students. Granting or declining admission would allow or not allow the student to participate in a course.

- (1) The system will display all new admission requests for that course in the following format:
 - Name of the student requesting admission
 - Note written by the student.
 - Date of admission request
- (2) The system will display a 'Grant' option and a 'Decline' option next to each admission request.
- (3) The system will, after the selection of instructor's decision, capture an optional note from the instructor to explain the reason for his/her decision.
- (4) The system will restrict the note to two hundred and fifty (250) characters.
- (5) The system will send the student an email informing him/her about the instructor's decision included with the note.

2.9.4 Revoke Admission (Only for Instructor)

This feature allows the instructor to revoke admission for a student who has already been granted admission. Revoking admission would inhibit the student to participate any more in the course. This feature can be used when, for example, a student drops out of a course.

- (1) The system will display a list of students currently granted admission to participate in the course.
- (2) The system will display an option 'Revoke' next to each student.
- (3) The system will ask for confirmation from the instructor once he/she selects the 'Revoke' option.
- (4) The system will capture an optional note from the instructor to explain the reason for his/her decision.
- (5) The system will restrict the note to two hundred and fifty (250) characters.
- (6) The system will delete the rights of the student to participate in the course after the instructor submits the note.
- (7) The system will email the student informing him/her about the instructor's decision along with the note as a part of the message.
- (8) The system will replace the student with the instructor in all active workflows.

(The system will assign the instructor with the responsibilities of the student whose admission has been revoked. So, now the instructor will be responsible for all artifacts for which the student was originally responsible.)

2.10 Assignment Management

The instructor can create assignments and set their start date and end date using this feature. The instructor can also specify whether the artifacts within the assignment are initiated on submission of previous artifact (event-based) or on the artifact start date and time (time-based) (The first artifact of the assignment is always time-based).

2.10.1 View Assignments

The instructor can view a list of all assignments he/she has created in the course after he/she selects 'Assignment Management' from the 'Administer' option.

- (1) The system will display, all the assignments created for the course with the following information: assignment name, assignment description, start date and time, end date and time, initiation type and grading algorithm.
- (2) The system will display 'Edit' and 'Delete' options next to each assignment.
- (3) The system will also display an option 'Add new assignment' to create a new assignment

2.10.2 Add Assignment

The instructor can add a new assignment using this feature. An example of an assignment can be homework, midterm exam or final exam. The instructor will have to provide the information below to successfully add an assignment.

- (1) The system will capture the following information from the instructor once he/she selects the option of "Add New Assignment":
 - Assignment name
 - Assignment description (optional)

- Start date
 - Start time
 - End date
 - End time
 - Initiation type
 - Grading algorithm
 - Template set
 - Manual intervention necessary
 - Threshold value for manual intervention
- (2) The system will restrict the assignment name to fifty (50) alphanumeric characters.
 - (3) The system will display the message “Please enter the assignment name” and move focus to the assignment name field if the instructor has left the assignment name blank.
 - (4) The system will check for an assignment with the same name in the current course, and display the message “Assignment with this name already exists, please enter another name” and move focus back to the assignment name field, if an assignment with the same name is found.
 - (5) The system will restrict the combination of start date and start time to be later than the current system time.
 - (6) The system will display the message “Start date and time should be later than the current date and time” and move the focus to the ‘Start Date’ field if the start date is in an invalid range.

- (7) The system will restrict the combination of end date and end time to be later than the start date and time.
- (8) The system will display the message that “End date and time should be later than the start date and time of assignment” and move the focus to the ‘End Date’ field if the end date is in an invalid range.
- (9) The system will allow the instructor to select whether the initiation type of the assignment is ‘time-based’ or ‘event-based’, with ‘time-based’ being the default selection.

(‘Time-based’ initiation type implies that all the artifacts will start on the start date and time specified. However, ‘event-based’ implies submitting one artifact will trigger (start) off the next artifact. The initiation type applies to all artifacts in the assignment.

When two artifacts are in the same time period (having the same start and end dates) then in an ‘event-based’ artifact both artifacts would have to be submitted to trigger off the next artifact.)

- (10) The system will display a list of pre-defined algorithms to the instructor, allowing him/her to select a grading algorithm to apply to this assignment with the default algorithm being selected automatically.
- (11) The system will allow the instructor to enable or disable ‘manual intervention necessary’.

(‘Manual intervention necessary’ implies that the instructor/student responsible for the ‘manual grade’ artifact in that workflow will manually grade the artifact.

If 'manual intervention necessary' is disabled then the instructor would not be able to add 'manual grade' artifact to the assignment.)

- (12) The system will allow the instructor to specify the threshold value after which manual intervention is needed if 'manual intervention necessary' is enabled.

(If the difference in evaluation points is the same as or more than the threshold value then the system would not automatically calculate the final grade but instead notify the instructor to calculate the final grade manually. For e.g. if the threshold value is set to 2 and evaluator #1 gives 5 points to a solution and evaluator #2 gives 9 points then the difference between the points is too big to be graded automatically.)

- (13) The system will restrict the threshold value to be two (2) numeric characters.

- (14) The system will display a list of available template sets allowing the instructor to select one template set which has to be applied to the assignment (Refer to 2.12).

(Templates guide the user to input the data in a specific format. They provide a specific layout on the screen which helps the user while submitting any artifact)

- (15) The system will add the new assignment to the database upon submission.

- (16) The system will create a workflow to represent the assignment for each student (Refer to 2.13.1).

- (17) The system will create a new task area for every new assignment that is added.

2.10.3 Edit Assignment

The instructor can edit the details of an assignment using this feature. The system will allow the instructor to edit the information mentioned in 2.10.2(1) if the assignment has not started yet.

2.10.4 Delete Assignment

This feature allows the instructor to delete an existing assignment if it has not already been started.

- (1) The system will ask for a confirmation after the instructor selects the 'Delete Assignment' option.
- (2) The system will delete the assignment from the database upon confirmation.
- (3) The system will display a confirmation after successful deletion.

2.11 Artifact Management

The instructor can create artifacts, allocate them to assignments, and set other attributes for each artifact. The students can submit the artifacts assigned to them using this functionality.

The instructor can view the option 'Artifact Management' under the 'Administer' option. The system will create a workflow step for each artifact within the assignment workflow (Refer to 2.13.1).

The user will select a pending artifact to be submitted from his/her pending tasks (Refer to 2.6.6). Once they select the artifact, an appropriate template for the artifact will be displayed which will allow the students to submit their artifacts (Refer to 2.11.5).

2.11.1 View Artifacts (Only for Instructor)

The instructor can view all artifacts that he/she has created in the course after he selects 'Artifact Management' under the 'Administer' option.

- (1) The system will display all the artifacts in the course grouped by the assignment to which they belong to with the following attributes: name, description, type, start date and time, end date and time and role player associated.
- (2) The system will display 'Edit' and 'Delete' options next to each artifact. (Refer to 2.11.3 and 2.11.4).
- (3) The system will also display an option 'Add new artifact' to create a new artifact (Refer to 2.11.2).

2.11.2 Add New Artifact (Only for Instructor)

The instructor can create a new artifact by providing the information mentioned below. While creating the artifact the instructor can also specify the assignment to which the artifact belongs and the role associated with the artifact. An example of an artifact would be questions created by the students, solutions by students, solution evaluations by students etc.

- (1) The system will capture the following information once the instructor selects 'Add Artifact':
 - Type of artifact
 - Belongs to assignment
- (2) The system will display a list of distinct current artifact types allowing the instructor to choose one.

(The system will have seven predefined artifact types which the instructor can use:

Problem – The problem or question submitted by the student or instructor.

Solution – The solution or answer submitted by the student to the problem artifact.

Evaluation – The evaluation of an artifact to be assigned points to it. Evaluation artifact does not necessarily mean evaluation of the solution, it just means evaluation of another artifact.

Manual grade – The manual grade assigned by the instructor in case of the evaluations meeting the threshold value criteria set by the instructor in 2.10.2(11).

This artifact can be used only if the instructor has specified a threshold value.

Dispute – The dispute submitted by the student if he/she is not satisfied with the final grade.

Arbitration – The result of the dispute arbitration generally held by the instructor but it can also be held by another student.

Review – The review done by the instructor or student to ensure standard quality of an artifact. Review artifact can be used to review any other artifact but generally it is used to review a problem artifact, evaluation artifact or arbitration artifact.

Refer to Appendix B for artifact configuration example.)

- (3) The system will display ‘Manual grade’ artifact in the above list only if the ‘manual intervention necessary’ is enabled for the assignment (Refer to 2.10.2(11)).
- (4) The system will display a list of assignments in the course to allow the instructor to choose from for the “Belongs to Assignment” option.

(The artifact is generally one of the steps in an assignment. So an artifact needs “to belong” to an assignment.)

- (5) The system will capture the following information from the instructor once he/she selects the artifact type and the assignment to which the artifact belongs to:
- Artifact name
 - Artifact description (Optional)
 - Role associated
 - Assign role to
 - Role player should not be same as (Optional)
 - Role player should be same as (Optional)
 - Start date
 - Start time
 - End date
 - End time
 - Required artifact
- (6) The system will allow the instructor to enter the artifact name with the default artifact name being the same as the artifact type.
- (7) The system will restrict the artifact name to fifty (50) alphanumeric characters.
- (8) The system will display the message “This artifact name already exists, please enter another name” if the artifact name already exists in that assignment.
- (9) The system will display the message “Please enter the artifact name” and move focus to the artifact name field if the instructor has left the artifact name blank.
- (10) The system will display a list of distinct roles already created by the instructor, allowing the instructor to select the role associated with the artifact.

(Role is a name given to the participant assigned with the responsibility of completing the artifact. Examples of roles can be problem creator, problem solver, evaluator #1, evaluator #2, disputer, etc.)

- (11) The system will also allow the instructor to enter a new role in the event that the instructor would like to add a new role.
- (12) The system will restrict the role to be fifty (50) alphanumeric characters.
- (13) The system will display the message “Please enter the role” if the instructor has not selected an existing role or entered a new role.
- (14) The system will display the following options to allow the instructor to associate the role to either himself/herself or the students with the default selection being the students:
 - Instructor
 - Students

(The instructor can either assign the responsibility of submitting the artifact to himself or to a student)

- (15) The system will display a list of distinct roles in the current assignment, allowing the instructor to specify that the role player associated with the current artifact should not be same as the role player(s) of other artifact(s).

(The instructor can specify whether the “role player” of artifact should be different than “role player” of another artifact. For example, the student playing the role of “problem solver” should be different than the student playing the role of “problem creator”.)

- (16) The system will display a list of distinct roles in the current assignment allowing the instructor to specify that the role player associated with the current artifact should be same as the role player of other artifact.
- (The instructor can specify whether the “role player” of artifact should be same as the “role player” of another artifact. For example, the student playing the role of “evaluator #1” should be same as the student playing the role of “problem creator”. In other words, a student playing both these roles should be the same student.)*
- (17) The system will allow the instructor to specify the start date and start time of the artifact.
- (18) The system will restrict the combination of artifact start date and start time to be later than the assignment start date and time and before than assignment end date and end time.
- (19) The system will allow the instructor to specify the end date and end time of the artifact.
- (20) The system will restrict the combination of end date and end time to be later than artifact start date and time.
- (21) The system will allow the instructor to specify ‘Yes/No’ to whether the artifact is a ‘Required Artifact’ with the default selection being ‘Yes’.
- (22) The system will allow the instructor to specify ‘Reminder preferences’ for each artifact.

- (23) The system will override the default reminder preferences and change it to the customized preferences set by the instructor for the artifact (Refer to 2.15.1 for default reminder preferences).
- (24) The system will capture additional information from the instructor depending upon the artifact type selected:

For 'Solution' Artifact

- (25) The system will display a list of 'problem' artifacts in the assignment, allowing the instructor to select the one which is the 'corresponding problem artifact'.
- (An assignment can have many problems or sub-problems, hence, to know which 'problem' artifact this is a solution to, the instructor needs to select the corresponding problem artifact.)*

For 'Review' Artifact

- (26) The system will display a list of all artifacts in the assignment, allowing the instructor to select one which needs to be reviewed.
- (In the PLA, technically any artifact can be reviewed. Hence, the instructor determines which artifact needs to be reviewed.)*

For 'Evaluation' Artifact

- (27) The system will display all artifacts for the assignment whose value for maximum evaluations has not been reached, allowing the instructor to select 'The artifact to evaluate'.
- (If the artifact is an 'evaluation' artifact then the instructor can assign the artifact which needs to be evaluated. In the PLA, any artifact can be evaluated and awarded a grade.)*

- (28) The system will restrict the number of evaluations per artifact to two evaluations.

(The PLAE version 1.0 supports maximum two evaluations per artifact. Once the instructor assigns the same artifact in 'artifact to evaluate', PLAE will not display that artifact in 'artifact to evaluate' option to be selected.)

- (29) The system will, depending upon the number of questions supported by the assignment's template set, allow the instructor to specify the point value for each question.

(For example, if the template set supports two essay type questions then the instructor would have to enter the number of points each question is worth.)

- (30) The system will ensure that value of points in both evaluation artifacts is the same.

For 'Manual Grade' Artifact

- (31) The system will display all artifacts for the assignment, allowing the instructor to select 'The artifact to be graded manually'.

(If the artifact is a 'manual grade' artifact then the instructor will have to assign an artifact for which a manual grade needs to be decided.)

For 'Arbitration' Artifact

- (32) The system will display a list of 'dispute' artifacts in the assignment, allowing the instructor to select the one which is the 'corresponding dispute artifact'.

(An assignment can have many disputes hence to know which 'dispute' artifact this is arbitration for, the instructor needs to select the corresponding 'dispute' artifact.)

- (33) The system will create a workflow step to represent the artifact in all assignment workflows (Refer to 2.13.1).

(For each artifact that is created, a workflow step would be created in all workflows. So, for example, if there are 30 students in a class, 30 workflows would be created and a workflow step would be created in each workflow to represent the artifact.)

- (34) The system will add the artifact to the database and display a confirmation message upon successful addition.
- (35) The system will, if the artifact's end date or end time is later than the assignment's end date or end time, automatically update the assignment's end date or time to the artifact's end date or time.

(This will help the instructor the match the assignment's end date or time with the last artifact's end date and time.)

2.11.3 Edit Artifact (Only for Instructor)

The instructor can modify the artifact details using this feature. The system will allow the instructor to edit the details mentioned in 2.11.2(5) only if the artifact has not started.

2.11.4 Delete Artifact (Only for Instructor)

The instructor can delete an artifact if it has not already been started.

- (1) The system will display a message "This artifact period has already been started so it cannot be deleted" if the artifact's start date and time is before the current date and time.
- (2) The system will confirm that the instructor wants to delete the artifact.
- (3) The system will delete the artifact whose start date and time is later than current date and time.

- (4) The system will pre-pone the start dates of the artifacts after the deleted artifact keeping the same duration for the artifact.

(If an artifact placed in the middle of an assignment is deleted, the system will adjust the start dates of the artifacts after the deleted artifact to fill in the gap for the deleted artifact while retaining the same duration to complete the artifacts.)

- (5) The system will display a confirmation after successful deletion.

2.11.5 Submit Artifact

This feature allows the user to submit an artifact using the template assigned to it.

- (1) The system will display the list of pending artifacts for the user.
- (2) The system will allow the user to select the artifact which he/she would like to submit.

For ‘Problem’ artifact

- (3) The system will display the appropriate template associated with the ‘problem’ artifact.
- (4) The system will allow the user to enter questions into the template.
- (5) The system will allow the user to enter answer options per question if the template supports objective questions.

(The PLAE version 1.0 supports essay type questions and objective questions. For objective questions, the user would have to enter the question along with the answer options for the questions. The number of answer options depends on the number supported by the template set.)

For 'Solution' artifact

- (6) The system will display the appropriate template associated with the 'solution' artifact.
- (7) The system will display the questions submitted in the 'problem' artifact along with the answer options in a selectable format if the questions are objective.
- (8) The system will allow the user to select one (1) answer option as the answer for objective questions.

(PLAE version 1.0 supports only mutually exclusive answers.)

- (9) The system will allow the user to enter the answer for subjective questions.

For 'Review' artifact

- (10) The system will display the appropriate template associated with the 'review' artifact.
- (11) The system will display the content of the artifact being reviewed.
- (12) The system will display input elements to edit the original content of the artifact.

(Although any artifact can be reviewed in PLAE, ideally 'review' artifact is used to review and change the questions for the 'problem' artifact.)

For 'Evaluation' artifact

- (13) The system will display the appropriate template associated with the 'evaluation' artifact.
 - (14) The system will display the content of the artifact being evaluated.
- (Any artifact in PLAE can be evaluated.)*
- (15) The system will allow the user to enter 'points' value for each element in the artifact.

- (16) The system will allow the user to enter optional comments for the evaluation.
- (17) The system will calculate the final score for the 'evaluation' artifact based on the algorithm selected by the instructor for that assignment if all evaluations have been submitted.

(If all the evaluations are submitted (and locked in case of 'event-based' assignments) in a workflow then the system will calculate the final score using the algorithm selected by the instructor.)

For 'Manual Grade' artifact

- (18) The system will display the appropriate template associated with the 'manual grade' artifact.

- (19) The system will display the content of the artifact being evaluated.

(Any artifact in PLAE can be evaluated.)

- (20) The system will allow the user to enter 'points' value for each element in the artifact.

- (21) The system will allow the user to enter optional comments for the manual grading.

For 'Dispute' artifact

- (22) The system will display the appropriate template associated with the 'dispute' artifact.

- (23) The system will allow the user (student) to enter his/her dispute.

For 'Arbitration' artifact

- (24) The system will display the appropriate template associated with the 'arbitration' artifact.

- (25) The system will display the content of the dispute artifact.

- (26) The system will allow the user to enter his/her arbitration decision.
- (27) The system will not accept any blank submissions from the user.
- (28) The systems will, for an artifact which belongs to an event-based assignment, display an option 'Lock Artifact' to allow the user to specify that this is the final version of the artifact.

(In an 'event-based' assignment, submission of one artifact will trigger off the next artifact. The option 'Lock artifact' allows the user to specify that this is final version of the artifact and that the next artifact can be started. If this option is not selected then the artifact is submitted but it remains unlocked till the user 'locks' it or artifact's end date and time has reached, whichever happens earlier.)

- (29) The system will allow the user to submit an artifact without locking it.
- (30) The system will display the unlocked but submitted artifact as a pending task (Refer to 2.6.6).
- (31) The system will create an entry under the task area for the artifact.
- (32) The system will not show the username who submitted the artifact.

(Submitting the artifact is totally anonymous among students. They will not be able to see the name of the student who posted the artifact. The instructor, however, will be able to see all usernames for artifact postings in the task area.)

- (33) The system will automatically lock all unlocked artifacts at the artifact's end date and time.

2.11.6 View/Edit Submitted Artifact

This feature allows the user to browse through the task area and view the artifacts which have already been submitted.

- (1) The system will display a list of task areas and artifacts under them.
- (2) The system will allow the user to select an artifact which he/she wishes to view.
- (3) The system will display the artifact in a non-editable format for the user if the user is not the 'artifact owner' (user who has submitted the artifact).
- (4) The system will display the artifact in an editable format within its template for the 'artifact owner' if the artifact is still not locked and artifact end date and end time is later than current date and time.
- (5) The system will display the message "Save your changes or they will be lost" if the user changes an unlocked artifact and does not submit it.
- (6) The system will display the artifact in non-editable format within its template for the 'artifact owner' if the artifact has been locked or artifact end date and end time is later than current date and time.
- (7) The system will display any or all of the following options with each artifact:
 - Previous artifact
 - Next artifact
 - Mark read
 - Save artifact (only for artifact owner)
 - Lock artifact (only for artifact owner and in case of 'event-based' assignments)

('Previous artifact' and 'Next artifact' means the previous and next artifact in the same workflow, respectively.)

- (8) The system will display the 'Previous Artifact' option if there is an artifact before the current artifact.
- (9) The system will display the 'Next Artifact' option if there is an artifact after the current artifact.
- (10) The system will display the 'Mark Read' option if artifact is not marked read yet.
- (11) The system will display the previous artifact in the same workflow when user selects the 'Previous artifact' option.
- (12) The system will display the next artifact in the same workflow when user selects the 'Next artifact' option.
- (13) The system will mark the current artifact as read when user selects 'Mark read' option.
- (14) The system will allow the artifact owner to edit the artifact and submit it (Refer to 2.11.5).

2.12 Template Management

The PLAE version 1.0 would have predefined template sets which can be assigned to an assignment. Each template set would contain seven templates, one each for an artifact type.

The instructor can view the list of predefined template sets available to him/her to assign to assignments. He/she will also be able to view the layout of each template within a template set.

2.12.1 View Template Sets

The instructor can view the predefined template sets using this feature. Viewing the template set would provide the instructor the option to view template layout (Refer to 2.12.2) for each artifact and also view a brief description for each layout.

- (1) The system will, after the instructor selects 'Template Management', display a list of available artifacts with the following information:
 - Name
 - Description
 - Used in assignments
- (2) The system will display the template set name for each template set in the 'Name' option.
- (3) The system will display the description of the template set in the 'Description' option.
- (4) The system will display the assignments using the template set in the 'Used in assignments' option.
- (5) The system will allow the instructor to select a template for which he/she wishes to see the template layouts for each artifact type (Refer to 2.12.2).

2.12.2 View Template Layout

This feature allows the instructor to view template layout for an artifact. Viewing the template will give the instructor a general idea of the layout of the template before he/she decides to assign the template set to the assignment.

- (1) The system will, after the instructor selects the template set he/she wishes to view, display the following information:

- Artifact type
 - Description
- (2) The system will display all the artifact types supported by the template set in the 'artifact type' option.
 - (3) The system will display the description of the template for artifact type in the 'description' option.
 - (4) The system will display an option 'view template layout' for each artifact to allow the instructor to view the artifact template layout.

2.13 Workflow Management

For every assignment that the instructor creates, the system will create a workflow for each student in the course. Each artifact created by the instructor will be represented by a workflow step in each workflow. The workflow will be updated every time a user submits an artifact.

2.13.1 Create workflow (Only for system)

Every assignment in a course is represented by a workflow. One workflow is created for each student. So, if there are 30 students in a course, 30 workflows, one for each student would be created.

- (1) The system will create a workflow for each assignment on the start date and time of the assignment.
- (2) The system will create a workflow step for each artifact in the assignment.

(For example, if there were 5 artifacts in an assignment and 30 students in the class doing the assignment, there would be $30 \times 5 = 150$ workflow steps in all.)

- (3) The system will assign the responsibility of each step (artifact) to a student depending upon allocations generated by the student allocation tool either manually or automatically (Refer to 2.14.1 and 2.14.3).

2.13.2 Workflow Viewer (Only for instructor)

The instructor can see the status of all workflow instances for an assignment that has started.

- (1) The system will display all the assignments in the system with their status (started/not started).
- (2) The system will display the option to 'view status' for all assignments that have started.
- (3) The system will display all workflow instances for that assignment highlighting their current status.
- (4) The system will display the name of the participant responsible for each workflow step, allowing the instructor to select it which would display the artifact submitted for that step.

2.14 Student Allocation Tool

Students to be allocated to the role for an artifact are chosen randomly. The system will ensure that rules created by the instructor for allocating roles are observed (Refer to 2.11.2(15) and 2.11.2(16)).

2.14.1 Manual Generation of Student Allocations (Only for Instructor)

This feature allows the instructor to generate student allocations for artifacts. Student allocation means allocating which student would be responsible for a particular artifact within a particular workflow. The system will generate these student allocations ensuring that the rules created while creating artifacts for assigning students to artifacts are followed. The instructor will use this feature if he/she wants to view and change the student allocations before they are saved.

- (1) The system will display the list of assignments with option 'Generate student allocations' for each assignment whose start date and time are later than current start date and time or whose student allocations have not yet been generated and saved.
- (2) The system will randomly choose students for the roles associated with the artifacts within the assignment.
- (3) The system will ensure that the rules created by the instructor for each artifact are followed (Refer to 2.11.2(15) and 2.11.2(16)).
- (4) The system will display all the student allocations for the entire assignment to the instructor.
- (5) The system will display an option 'Regenerate student allocations' to allow the instructor to regenerate the student allocations if he/she is not satisfied with the allocations.
- (6) The system will display an option 'Save student allocations' to allow the instructor save the student allocations in the system.
- (7) The system will create workflows for the assignment once the student allocations are generated and saved (Refer to 2.13.1).

2.14.2 View Student Allocations

The instructor can view the current student allocations for an assignment using this feature if the assignment has already started or the student allocations have already been allocated and saved.

- (1) The system will display the list of assignments with option 'View student allocations' for each assignment whose start date and time are earlier than current start date and time or whose student allocations have been generated and saved.
- (2) The system will display the current student allocations for artifacts within that assignment after the instructor selects 'View student allocations'.

2.14.3 Automatic Generation of Student Allocations (Only for System)

This feature will generate the student allocations for artifacts within an assignment automatically on the start date and time of the assignment if manual allocations have not been already done.

- (1) The system will randomly choose students for the roles associated with the artifacts within the assignment automatically on the start date and start time of assignment if the student allocations for that artifact have not yet been generated.
- (2) The system will ensure that the rules created by the instructor for each artifact are followed for student allocations (Refer to 2.11.2(15) and 2.11.2(16)).
- (3) The system will save the student allocations to the database.
- (4) The system will create workflows for the assignment once the student allocations are generated. (Refer to 2.13.1).

2.15 Reminder Preferences

The instructor can set his default reminder settings which would be applicable to all artifacts. However, the reminder settings for an individual artifact would override the default settings.

2.15.1 Set Default Reminder Preferences (Only for Instructor)

This feature allows the instructor to set default reminder preferences for artifacts. Students/instructors will be reminded of their artifact responsibilities according to the preferences set by the instructor using this feature. However any reminder preferences set for a particular artifact will override the default reminder preferences.

- (1) The system will retrieve the current reminder preferences and display them in the appropriate options mentioned below.
- (2) The system will display the following options under ‘Reminder Preferences’:
 - Send Reminders to
 - Reminder triggers
- (3) The system will display the following options under ‘Send Reminders to:’
 - Instructor
 - Students
- (4) The system will allow any one (1) or both the options to be selected under ‘Send Reminders to’.
- (5) The system will display the following options under ‘Reminder Triggers:’
 - X days before end date
 - X days after end date (for late posters)
 - Maximum number of reminders after end date (for late posters)

- As soon as the artifact period starts
 - As soon as the artifact period ends
 - As soon as the artifact is submitted
- (6) The system will allow any one (1) or all options to be selected under “Reminder Triggers”
- (7) The system will display the message “Please select reminder trigger(s) for your reminder preferences” if the instructor has selected an option under ‘Send Reminders To’ but no option under ‘Reminder Triggers’
- (8) The system will save the preferences selected by the instructor to the database overwriting any current preferences.
- (9) The system will display a confirmation message to the instructor.

2.15.2 Send Reminder Email (Only for System)

The system will send a reminder email to instructor or students or both depending upon the preferences set by the instructor in 2.15.1.

2.16 View/Edit Final Grades

The instructor can view the final grades for students in each assignment using this feature.

2.16.1 View Final Grades

This feature allows the instructor to view all the final grades assigned to students for an evaluated artifact in an assignment at one glance.

- (1) The system will display all assignments in the course with the grading algorithm associated with each one of them and their status (started/not started).
- (2) The system will display an option 'View/Edit' next to each assignment, allowing the instructor to view the final grades for an assignment.
- (3) The system will, after the instructor selects the assignment for which he wishes to view the final grades, display all grades grouped according to the artifact name for which the grades have been assigned.

(It is possible that there are two solutions in an assignment say HW1_Solution1 and HW1_Solution2 and both solutions are graded using two evaluations each. So a student should get a final grade for each solution artifact. Hence the grades for all students will be displayed in two groups 1) HW1_Solution1, 2) HW1_Solution2)

- (4) The system will display all the student names sorted alphabetically according to last name.
- (5) The system will display the final grade next to each student name.
- (6) The system will display all the evaluation grades assigned to that student.
- (7) The system will allow the instructor to enter a new grade for any final grade that he/she wishes to change (Refer to 2.16.2).

2.16.2 Edit Final Grades

This feature allows the instructor to change the final grade of a student.

- (1) The system will allow the instructor to enter a new grade for all students for whom he/she wishes to change the previous grade.
- (2) The system will confirm from the instructor after he submits the new grade.
- (3) The system will, on positive confirmation save the new final grades.

This chapter presented a detailed functional requirements analysis for PLAE version 1.0 to implement the process. The basic features which were covered in this chapter were assignment management, artifact management, workflow management, automatic evaluation. The features in this chapter supported peer assessment on the individual level. The PLAE version 2.0 will provide support group assessment, template creation and more advanced features to control the PLA process.

CHAPTER 3

DATABASE AND CLASS DESIGNS FOR PLAE VERSION 1.0

This chapter presents an entity-relationship diagram and a class diagram for PLAE version 1.0 based on the functional requirements described in Chapter 2. An Entity Relationship (ER) diagram shows the relationship between entities within a system. An entity can be defined as a thing of significance, either real or conceptual, about which the system being modeled needs to hold information. The following entities have been identified in the version 1.0: school, user (administrator, instructor, student, and experimenter), course, assignment, artifact, template sets. An ER diagram is critical to the database design.

Class diagram is an object-oriented technique that provides a static view of the system by showing the classes within the system and the relationships between them. It is a part of the Unified Modeling Language suite (www.uml.org). Microsoft® Visio® 2002 was used to model both diagrams.

3.1 Entity Relationship Diagram

Figure 3.1 is the ER diagram for PLAE version 1.0 based on the requirements described in Chapter 2.

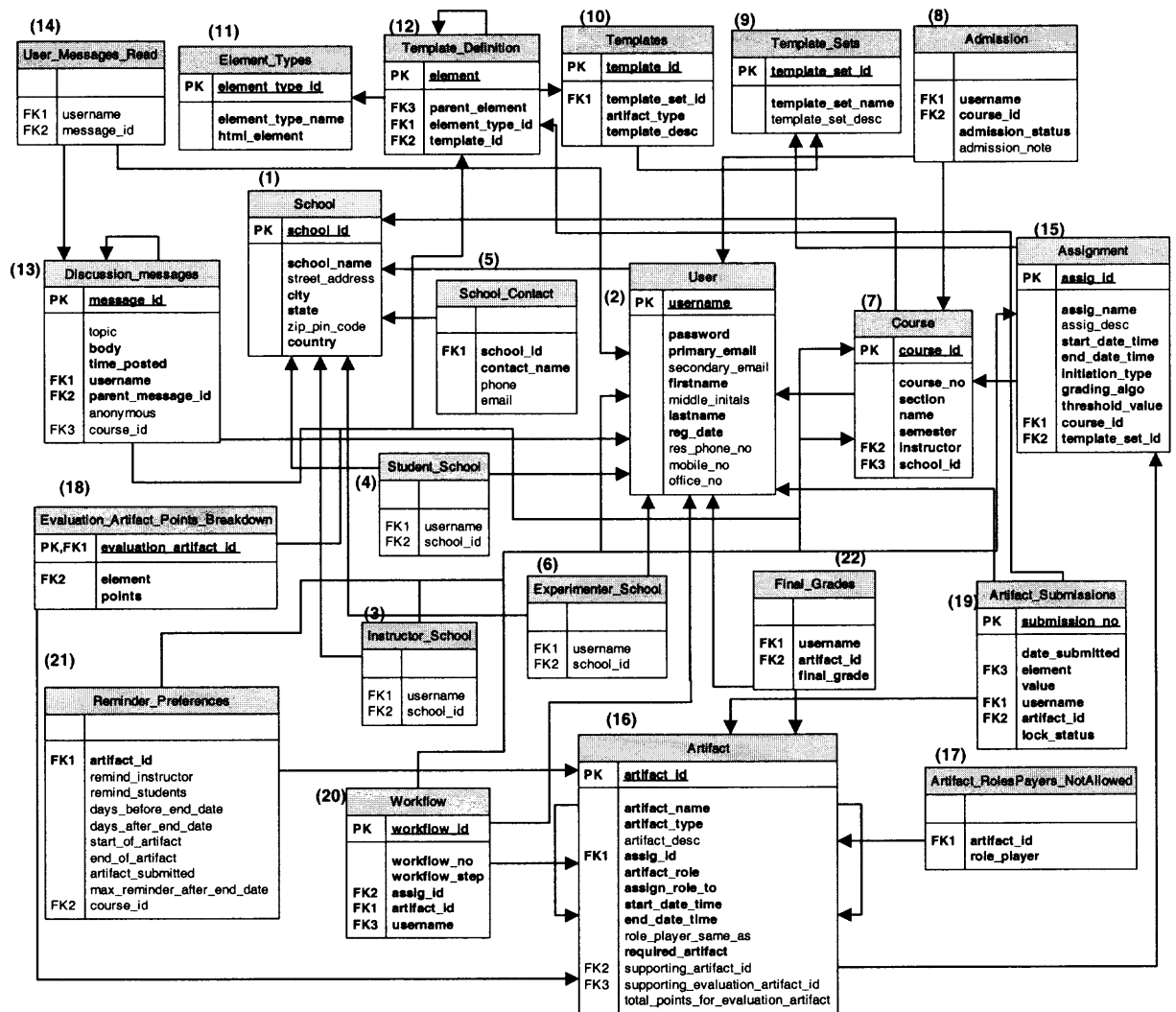


Figure 3.1 ER diagram for PLAE version 1.0.

Entity attributes in **bold** in the diagram indicate that they are required attributes for that entity. The acronym PK indicates the primary key of the entity which uniquely identifies that entity and FK indicates foreign key which identifies the relationship between two entities.

Following is a description of all the entities and their attributes in the diagram.

(1) School:

This entity represents the schools registered with the PLAE.

Table 3.1 School Entity

Attribute name	Data type	Required	Description	Value range
school_id (PK)	Integer	Yes	A unique identification number which identifies the school.	N/A
school_name	Varchar(250)	Yes	The school name.	N/A
street_address	Varchar(200)	No	The street address of the school.	N/A
city	Varchar(100)	Yes	The city where school is located.	N/A
state	Varchar(100)	Yes	The state where the school is located.	N/A
zip_pin_code	Varchar(20)	No	Zip/pin code of the school address.	N/A
country	Varchar(100)	Yes	The country of the school.	N/A

(2) User:

This entity defines the attributes of the users associated with PLAE.

Table 3.2 User Entity

Attribute name	Data type	Required	Description	Value range
username (PK)	Varchar(20)	Yes	Uniquely identifies a user. Either automatically generated by the system or entered by the user. The system will contain one default user – “administrator”	N/A
password	Varchar(20)	Yes	Password chosen by the user to access the system.	N/A
primary_email	Varchar(100)	Yes	Primary email address of the user.	N/A
secondary_email	Varchar(100)	No	Secondary email address of the user.	N/A
firstname	Varchar(50)	Yes	First name of the user.	N/A
middle_intials	Varchar(10)	Yes	The middle initials of the user.	N/A
lastname	Varchar(50)	Yes	Last name of the user.	N/A
reg_date	Datetime	Yes	The date the user registered with the system.	N/A

Table 3.2 User Entity (Continued)

Attribute name	Data type	Required	Description	Value range
res_phone_no	Varchar(15)	No	Residence phone number of the user. This is only stored for the 'student' user.	N/A
mobile_no	Varchar(15)	No	Mobile phone number of the user. This is only stored for the 'student' user.	N/A
office_no	Varchar(15)	No	Office phone number of the user. This is only stored for the 'student' user.	N/A

(3) Instructor_school:

This entity represents the instructor-school associations.

Table 3.3 Instructor_School Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar(20)	Yes	Username of the instructor.	N/A
school_id (FK2)	Integer	Yes	The school to which the instructor is affiliated.	N/A

(4) Student_school:

This entity represents the student-school associations.

Table 3.4 Student_School Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar(20)	Yes	Username of the student.	N/A
school_id (FK2)	Integer	Yes	The school to which the student is affiliated.	N/A

(5) School_Contact:

This entity represents contact details of a school representative.

Table 3.5 School_Contact Entity

Attribute name	Data type	Required	Description	Value range
school_id (FK1)	Integer	Yes	The school to which the representative is affiliated.	N/A
contact_name	Varchar(200)	Yes	The name of the representative.	N/A
phone	Varchar(15)	No	The phone number of the representative.	N/A
email	Varchar(100)	No	The email id of the representative.	N/A

(6) Experimenter_School:

This entity represents the experimenter-school associations.

Table 3.6 Experimenter_School Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar(20)	Yes	Username of the experimenter.	N/A
school_id (FK2)	Integer	Yes	The school to which the experimenter is affiliated.	N/A

(7) Course:

This entity represents all the courses registered with the system.

Table 3.7 Course Entity

Attribute name	Data type	Required	Description	Value range
course_id (PK)	Integer	Yes	Unique number which identifies the course registered with the system.	N/A
course_no	Varchar(20)	Yes	A number used by the school to identify the course.	N/A
section	Varchar(20)	Yes	The section number of the course.	N/A

Table 3.7 Course Entity (Continued)

Attribute name	Data type	Required	Description	Value range
name	Varchar(100)	Yes	The name of the course.	N/A
semester	Varchar(50)	Yes	The semester when the course is offered.	N/A
instructor (FK2)	Varchar (20)	Yes	A registered instructor teaching the course.	N/A
school_id (FK3)	Integer	Yes	The school id of the school offering the course.	N/A

(8) Admission:

This entity represents the admission status of students in a course.

Table 3.8 Admission Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar (20)	Yes	The username of the student.	N/A
course_id (FK2)	Integer	Yes	The course id from the Course entity.	N/A
admission_status	Char(1)	Yes	The status of the admission request.	Three permissible values: 'A', 'D' or 'U' indicating admitted, declined or undecided, respectively.
admission_note	Varchar(250)	No	The note written by the student while requesting admission.	N/A

(9) Template_Sets:

This entity represents all the templates saved in the system.

Table 3.9 Template_Sets Entity

Attribute name	Data type	Required	Description	Value range
template_set_id (PK)	Integer	Yes	Unique number to identify a template set.	N/A
template_set_name	Varchar(100)	Yes	A name for the template set.	N/A

Table 3.9 Template_Sets Entity (Continued)

Attribute name	Data type	Required	Description	Value range
template_set_desc	Varchar(250)	No	A template description which describes the template set.	N/A

(10) Templates:

This entity represents all the artifact templates within a template set.

Table 3.10 Templates Entity

Attribute name	Data type	Required	Description	Value range
template_id (PK)	Integer	Yes	The unique id which identifies a template.	N/A
template_set_id (FK1)	Integer	Yes	The template set to which the template belongs.	N/A
artifact_type	Integer	Yes	A number to identify the artifact type.	Seven possible values: (1,2,3,4,5,6,7) for problem, review, solution, evaluation, manual grade, dispute, arbitration, respectively
template_desc	Longtext*	Yes	The template description.	N/A

* Longtext is a data type supported by Microsoft Access

(11) Element_Types:

This entity represents the master element_types. This entity comes preloaded with data. Refer to Appendix A to see the data.

Table 3.11 Element_Types Entity

Attribute name	Data type	Required	Description	Value range
element_type_id(PK)	Integer	Yes	The unique id for an element type.	Refer Appendix A
element_type_name	Varchar(20)	Yes	The name of the element type.	Refer Appendix A
html_element	Integer	Yes	HTML element (control) used to capture the data.	Refer Appendix A

(12) Template_Definition:

This entity represents all the definition of each template within a template set.

Table 3.12 Template_Definition Entity

Attribute name	Data type	Required	Description	Value range
element (PK)	Integer	Yes	The unique id for a template element.	N/A
parent_element (FK3)	Integer	Yes	The element id with which it needs to be grouped together. For example, the four multiple choice options of a question would have the question element as the parent element. The default value is -1 which means there is no parent element.	N/A
element_type_id (FK1)	Integer	Yes	The element type.	N/A
template_id (FK2)	Integer	Yes	The template to which the element belongs.	N/A

(13) Discussion_Messages:

This entity represents the messages posted by the users in the discussion area.

Table 3.13 Discussion_Messages Entity

Attribute name	Data type	Required	Description	Value range
message_id (PK)	Integer	Yes	A unique identification number for each message.	N/A
topic	Varchar(100)	No	The topic of the message entered by the user.	N/A
body	Longtext*	Yes	The body of the message.	N/A
time_posted	Datetime	Yes	The time the message is posted. The system enters the current system time by default.	N/A
username (FK1)	Varchar(20)	Yes	The user who posts the message.	N/A
parent_message_id (FK2)	Integer	Yes	The message id to which the message is a reply to. If the message is a new thread then the parent_message_id will be -1.	N/A
anonymous	Char(1)	No	The option to display the message as anonymous.	The two values possible for this attribute are: 'Y' or 'N' for yes or no, respectively.
course_id (FK3)	Integer	Yes	The course to which the message is associated with.	N/A

* Longtext is a data type supported by Microsoft Access

(14) User_Messages_Read:

This entity stores all the messages marked read by a user.

Table 3.14 User_Messages_Read Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar (20)	Yes	The user who has read the message.	N/A
message_id (FK2)	Integer	Yes	The message id of the read message.	N/A

(15) Assignment:

All the assignments created in each course are represented by this entity.

Table 3.15 Assignment Entity

Attribute name	Data type	Required	Description	Value range
assig_id (PK)	Integer	Yes	A unique identification number for each assignment.	N/A
assig_name	Varchar (50)	Yes	The name of the assignment.	N/A
assig_desc	Longtext*	No	The description of the assignment.	N/A
start_date_time	Datetime	Yes	The date and time when the assignment should start.	N/A
end_date_time	Datetime	Yes	The date and time when the assignment should end.	N/A
initiation_type	Char(1)	Yes	The initiation type of the assignment.	This attribute can have only 2 possible values: 'E' or 'T' for 'event-based' or 'time-based', respectively.
grading_algo	Varchar(10)	Yes	The grading algorithm to be used to determine the final grade.	This attribute has three possible values: 'Highest', 'Lowest' or 'Average' to indicate highest grade, lowest grade or average of grades algorithms, respectively.
threshold_value	Integer	Yes	The threshold value on which manual intervention is necessary. The default value is -1 which indicates manual intervention is not required.	N/A
course_id (FK1)	Integer	Yes	The course to which the assignment belongs to.	N/A
template_set_id (FK2)	Integer	Yes	The template set to be used with the assignment.	N/A

*Longtext is a data type supported by Microsoft Access

(16) Artifact:

This entity represents the artifacts created within an assignment.

Table 3.16 Artifact Entity

Attribute name	Data type	Required	Description	Value range
artifact_id (PK)	Integer	Yes	A unique identification number for each artifact.	N/A
artifact_name	Varchar(50)	Yes	The name of the artifact.	N/A
artifact_type	Integer	Yes	A number to identify the artifact type.	Seven possible values: (1,2,3,4,5,6,7) for problem, review, solution, evaluation, manual grade, dispute, arbitration, respectively
artifact_desc	Longtext*	Yes	A description of the artifact.	N/A
assig_id (FK1)	Integer	Yes	The id of the assignment to which the artifact belongs.	N/A
artifact_role	Varchar(50)	Yes	The role associated with the artifact. Some examples of roles are: problem creator, problem solver, evaluator #1	N/A
assign_role_to	Varchar(20)	Yes	The participant responsible for the artifact.	Only two values are possible for this artifact: 'students' and 'instructors'.
start_date_time	Datetime	Yes	The date and time the artifact should start.	N/A
end_date_time	Datetime	Yes	The date and time the artifact should end.	N/A
role_player_same_as	Varchar(50)	Yes	The role player of the artifact can be same as the role player of another artifact in the same assignment.	N/A
required_artifact	Char (1)	Yes	This attribute indicates if it is necessary to submit the artifact.	N/A

Table 3.16 Artifact Entity (Continued)

Attribute name	Data type	Required	Description	Value range
supporting_artifact_id(FK2)	Integer	No	The artifact id of the supporting artifact. For example, in a solution artifact if the problem is to be displayed too then artifact id of the problem would be the value of this attribute for the solution artifact. In case of evaluation artifact, the value of this attribute will be the artifact id of the artifact to be evaluated.	The two values possible for this attribute are: 'Y' or 'N' for yes or no, respectively.
supporting_evaluation_artifact_id (FK3)	Integer	No	The artifact id of the evaluation artifact whose grades have to be considered to calculate the final grade. This attribute will only be used in case of 'evaluation' artifact.	N/A
total_points_for_evaluation_artifact	Integer	No	Incase of an 'evaluation' artifact, this attribute indicates the total points out of which the evaluation has to be done.	N/A

* Longtext is a data type supported by Microsoft Access

(17) Artifact_Roles_NotAllowed:

This entity represents the role players that should not be associated with the artifact.

Table 3.17 Artifact_Roles_NotAllowed Entity

Attribute name	Data type	Required	Description	Value range
artifact_id (FK1)	Integer	Yes	A unique identification number for each artifact.	N/A
Role_player	Varchar(50)	Yes	The role player that should not be associated with the artifact. For example, a problem creator should not be responsible for solving the problem.	N/A

(18) Evaluation_Artifact_Points_Breakdown:

This entity represents breakdown of points of an artifact being evaluated during an evaluation.

Table 3.18 Evaluation_Artifact_Points_Breakdown Entity

Attribute name	Data type	Required	Description	Value range
evaluation_Artifact_id (PK, FK1)	Integer	Yes	The artifact id of the evaluation artifact.	N/A
element (FK2)	Integer	Yes	The element representing the points element in the evaluation artifact template.	N/A
points	Integer	Yes	The points value for each element.	N/A

(19) Artifact_Submissions:

This entity represents the artifacts submitted by the users.

Table 3.19 Artifact_Submissions Entity

Attribute name	Data type	Required	Description	Value range
submission_no (PK)	Integer	Yes	The unique identification number of the submission.	N/A
date_submitted	Datetime	Yes	The date the artifact is submitted.	N/A
element (FK3)	Integer	Yes	The input element used to submit the artifact.	N/A
value	Longtext*	Yes	The value of the element submitted by the user.	N/A
username (FK1)	Varchar(20)	Yes	The user who submitted the artifact.	N/A
artifact_id (FK2)	Integer	Yes	The id of the artifact being submitted.	N/A
lock_status	Char(1)	Yes	To indicate whether the submitted artifact is locked.	There are only 2 possible values: 'L' or 'U' for 'Locked' or 'Unlocked', respectively. The default value is 'L'.

* Longtext is a data type supported by Microsoft Access

(20) Workflow:

This entity represents the participant allocations to artifacts and the state of the workflows in a course.

Table 3.20 Workflow Entity

Attribute name	Data type	Required	Description	Value range
workflow_id (PK)	Integer	Yes	A unique identification number for each workflow.	N/A
workflow_no	Integer	Yes	The workflow number in a particular course for each student. For every assignment, there will be same number of workflows as the number of students. For example, if there are 20 students there will be 20 workflows for an assignment.	N/A
workflow_step	Integer	Yes	The step number within each workflow. Each artifact is represented by a step in a workflow. For example, if there are 5 artifacts in an assignment then there are $20 \times 5 = 100$ workflow steps in all.	N/A
assig_id (FK2)	Integer	Yes	The assignment to which the workflow belongs.	N/A
artifact_id (FK1)	Integer	Yes	The id of the artifact which is being represented as workflow step.	N/A
username(FK3)	Varchar(20)	Yes	The user responsible for the workflow step.	N/A

(21)Reminder_Preferences:

This entity represents the reminder preferences set for each artifact and in general.

Table 3.21 Reminder_Preferences Entity

Attribute name	Data type	Required	Description	Value range
artifact_id (FK1)	Integer	Yes	Unique identification number of the artifact. Default value is -1 which means default reminder settings.	N/A
remind_instructor	Char(1)	No	A flag values to indicate if the instructor should be reminded about the artifact submission.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively. The default value is 'N'.
remind_students	Char(1)	No	A flag value to indicate if the students should be reminded about the artifact submission.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively. The default value is 'N'.
days_before_end_date	Integer	No	A numeric value to indicate the number of days before the artifact due date, the reminder should be sent. The default value is 0.	N/A
days_after_end_date	Integer	No	A numeric value to indicate the number of days after the artifact due date, the reminder should be sent. The default value is 0.	N/A
end_of_artifact	Integer	No	A flag to indicate if a reminder should be sent when the artifact ends. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
start_of_artifact	Integer	No	A flag to indicate if a reminder should be sent when the artifact starts. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
artifact_submitted	Integer	No	A flag to indicate if a reminder should be sent when the artifact is submitted. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.

Table 3.21 Reminder_Preferences Entity (Continued)

Attribute name	Data type	Required	Description	Value range
max_reminder_after_end_date	Integer	No	A numeric value to indicate the maximum number of reminders to be sent after the end date.	N/A
course_id	Integer	Yes	A numeric value which indicates which course the default reminder preferences apply to.	N/A

(22) Final_Grades:

This entity represents final grades students receive after manual or automatic evaluation.

Table 3.22 Final_Grades Entity

Attribute name	Data type	Required	Description	Value range
Username (FK1)	Varchar(20)	Yes	The username of the student.	N/A
artifact_id (FK2)	Integer	Yes	Unique number of the artifact for which a final grade is awarded.	N/A
Final_grade	Integer	Yes	The final grade calculated.	N/A

3.2 Class Diagram

Figure 3.2 shows the static view of PLAE version 1.0 by using a class diagram. The diagram shows the major entities as classes, the relationships between them and the cardinality of the relationships. Generally, a class diagram includes the functions and attributes within a class. However, this diagram only lists the major functions within a class and design of attributes and any other functions necessary for proper functioning of the class are left open to be decided during the actual detailed design or implementation of the class.

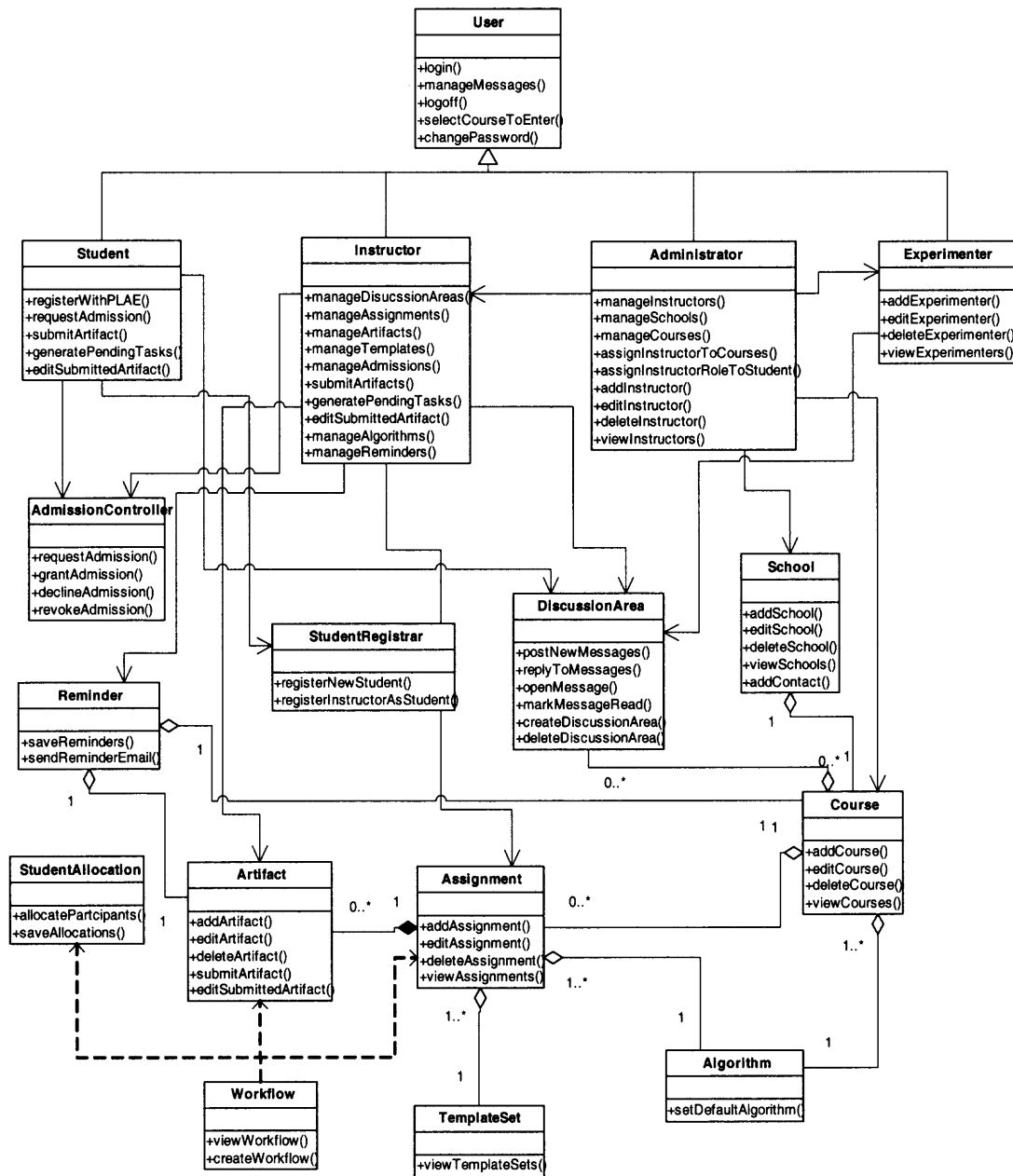


Figure 3.2 Class diagram for PLAE version 1.0.

The following section presents the classes involved and provides a brief description of the functions within each class.

(1) User:

This is a general class for all users. It includes functions common to all users.

Table 3.23 Class User

Function name	Purpose	Requirement reference
login()	To allow the user to login to the system.	2.1.1
manageMessages()	To allow the user to manage his/her messages i.e. read a message, post a new message, mark message read etc.	2.8.1-2.8.6, 2.6.4
logout()	To allow the user to logout from the system.	2.6.2
selectCourseToEnter()	To allow the user to select the course he/she wishes to enter.	2.6.1
changePassword()	To allow the user to change the password.	2.6.5

(2) Student:

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the student user.

Table 3.24 Class Student

Function name	Purpose	Requirement reference
registerWithPLAE()	To allow the student to register with PLAE system.	2.7.1
requestAdmission ()	To allow the student to request admission in a course.	2.9.1
manageArtifacts()	To allow students to manage artifacts by submitting artifacts or editing submitted artifacts.	2.11.5, 2.11.6
getPendingTasks()	To retrieve the pending tasks for a student.	2.6.6

(3) Instructor:

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the instructor user.

Table 3.25 Class Instructor

Function name	Purpose	Requirement reference
manageDiscussionAreas()	To allow the instructor to create and delete discussion areas and also to post messages, read messages, mark a message as 'read'.	2.8.1-2.8.9
manageAssignments()	To allow the instructor to create, edit and delete assignments.	2.10.1-2.10.4
manageArtifacts()	To allow the instructor to create, edit and delete artifacts.	2.11.1-2.11.6
viewTemplateSets()	To allow the instructor to view the template sets available.	2.12.1
manageAdmissions()	To allow the instructor to manage student admissions in a course.	2.9.3, 2.9.4
getPendingTasks()	To retrieve the pending tasks for an instructor.	2.6.6
manageReminders()	To allow the instructor to set reminder preferences.	2.15.1
addInstructor()	To allow the administrator to add a new instructor to the PLAE system.	2.3.2
editInstructor()	To allow the administrator to edit an instructor.	2.3.3
deleteInstructor()	To allow the administrator to delete an instructor.	2.3.4
viewInstructors()	To allow the administrator to view all instructors registered in the system.	2.3.1

(4) Administrator:

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the administrator user.

Table 3.26 Class Administrator

Function name	Purpose	Requirement reference
manageInstructors()	To allow the administrator to add, edit and delete an instructor.	2.3.1-2.3.4
manageSchools()	To allow the administrator to add, edit and delete a school from the system.	2.2.1-2.2.5

Table 3.26 Class Administrator (Continued)

Function name	Purpose	Requirement reference
manageCourses()	To allow the administrator to add, edit and delete a course from the system.	2.5.1-2.5.4
manageExperimenters()	To allow the administrator to add, edit and delete a experimenter from the system.	2.4.1-2.4.4
assignInstructorRoleToStudent()	To allow the administrator to assign the instructor role to student.	2.3.5

(5) Experimenter:

This is a child class of ‘User’. It inherits all the functions described in the ‘User’ class and some additional functions specific to the experimenter user.

Table 3.27 Class Experimenter

Function name	Purpose	Requirement reference
addExperimenter()	To allow the administrator to add a new experimenter to the PLAE system.	2.4.2
editExperimenter()	To allow the administrator to edit an experimenter details.	2.4.3
deleteExperimenter()	To allow the administrator to delete an experimenter.	2.4.4
viewExperimenter()	To allow the administrator to view all experimenter registered in the system.	2.4.1

(6) AdmissionController:

This class controls the admission of a student in a course.

Table 3.28 Class AdmissionController

Function name	Purpose	Requirement reference
requestAdmission()	To allow a student to request admission for a course.	2.9.1
grantAdmission()	To allow an instructor to grant admission to a student.	2.9.3
declineAdmission()	To allow an instructor to decline admission to a student.	2.9.3
revokeAdmission()	To allow an instructor to revoke admission from a student.	2.9.4

(7) StudentRegistrar:

This class controls the registration of a student in the PLAE.

Table 3.29 Class StudentRegistrar

Function name	Purpose	Requirement reference
registerNewStudent()	To allow a student to register with the PLAE system.	2.7.1
registerInstructorAsStudent()	To allow an instructor to create a student profile for himself/herself.	2.7.2

(8) DiscussionArea:

This class manages the discussion areas in a course.

Table 3.30 Class DiscussionArea

Function name	Purpose	Requirement reference
postNewMessage()	To allow a user to post a new message.	2.8.1
browseMessages()	To allow the user to browse through messages.	2.8.3, 2.8.4
replyToMessage()	To allow a user to reply to a message.	2.8.2
openMessage()	To allow the user to open a message to read.	2.8.2(1)

Table 3.30 Class DiscussionArea (Continued)

Function name	Purpose	Requirement reference
markMessageRead()	To allow the user to mark a message as 'read'	2.8.5, 2.8.6
createDiscussionArea()	To allow the instructor to create a new discussion area.	2.8.8
deleteDiscussionArea()	To allow the instructor to delete a discussion area.	2.8.9

(9) School:

This class manages the schools in the PLAE system.

Table 3.31 Class School

Function name	Purpose	Requirement reference
addSchool()	To allow the administrator to add a new school to the PLAE system.	2.2.2
editSchool()	To allow the administrator to edit school details.	2.2.4
deleteSchool()	To allow the administrator to delete a school.	2.2.5
viewSchools()	To allow the administrator to view all schools registered in the system.	2.2.1
addSchoolContact()	To allow the administrator to add a contact details of a person at the school.	2.2.3

(10) Reminder:

This class manages the reminder settings for a course and artifacts.

Table 3.32 Class Reminder

Function name	Purpose	Requirement reference
saveReminders()	To save the reminder preferences set by the instructor.	2.15.1
sendReminderEmail()	To send a reminder email according to the reminder preferences configured by the instructor.	2.15.1
generatePendingTasks()	To retrieve the pending tasks for a student/instructor.	2.6.6

(11) StudentAllocation:

This class manages the student allocations to artifacts.

Table 3.33 Class StudentAllocation

Function name	Purpose	Requirement reference
allocateParticipants()	To allocate the responsibilities of an artifact to a student.	2.14.1, 2.14.3
saveAllocations()	To save the allocations generated.	2.14.1(6)

(12) Artifact:

This class manages the artifacts within an assignment.

Table 3.34 Class Artifact

Function name	Purpose	Requirement reference
addArtifact()	To allow the instructor to add a new artifact to a assignment.	2.11.2
editArtifact()	To allow the instructor to edit an artifact in a assignment.	2.11.3
deleteArtifact()	To allow the instructor to delete an artifact from a assignment.	2.11.4
submitArtifact()	To allow the students/instructor to submit an assignment.	2.11.5, 2.11.6
openArtifact()	To view a submitted artifact.	2.11.6

(13) Assignment:

This class manages the assignments within a course.

Table 3.35 Class Assignment

Function name	Purpose	Requirement reference
addAssignment()	To allow the instructor to create a new assignment.	2.10.2
editAssignment()	To allow the instructor to edit an assignment.	2.10.3
deleteAssignment()	To allow the instructor to delete an assignment.	2.10.4
viewAssignment()	To allow the instructor to view all assignments.	2.10.1

(14) Course:

This class manages the courses in the PLAE system.

Table 3.36 Class Course

Function name	Purpose	Requirement reference
addCourse()	To allow the administrator to create a new course.	2.5.2
editCourse()	To allow the administrator to edit course details.	2.5.3
deleteCourse()	To allow the administrator to delete a course.	2.5.4
viewCourses()	To allow the administrator to view all courses.	2.5.1

(15) Workflow:

This class manages the workflows within a course. Workflow class is dependent on StudentAllocation, Artifact and Assignment classes.

Table 3.37 Class Workflow

Function name	Purpose	Requirement reference
viewWorkflow()	To allow the instructor to view workflows for an assignment.	2.13.2
createWorkflow()	To create workflows within an assignment	2.13.1

(16) TemplateSet:

This class allows viewing of template sets in PLAE version 1.0.

Table 3.38 Class TemplateSet

Function name	Purpose	Requirement reference
viewTemplateSets()	To allow the instructor to view all template sets available in the system.	2.12.1
viewTemplateLayout()	To allow the instructor to view the layout of a template within a template set.	2.12.2

(17) FinalGrade:

This class calculates the final grades of a student.

Table 3.39 Class FinalGrade

Function name	Purpose	Requirement reference
calculateFinalGrade()	To calculate a final grade based on the scores submitted in the 'evaluation' artifact.	2.11.5(17)
viewFinalGrades()	To allow the instructor to view final grades for an artifact.	2.16.1
editFinalGrade()	To allow the instructor to edit the final grade of a student.	2.16.2

The diagrams presented above should not be considered as rigid, they only provide a helpful starting point for the actual implementation of the system. They should be modified and re-modeled as required for the best approach while implementing the system.

CHAPTER 4

REQUIREMENT ANALYSIS FOR PLAE VERSION 2.0

This chapter analyzes and presents the features for PLAE version 2.0. The features selected for this version are designed to enhance the flexibility from the administrator's and the instructor's point of view to manage the PLA process across many courses. This chapter introduces a new user -Teaching Assistant (TA) for a course. The instructor and a teaching assistant of a course have the same administrative control over a course, so anything that an instructor can do, can also be done by the TA. This applies to all the PLAE version 1.0 features as well.

This chapter is organized in a similar manner to Chapter 2 with only the new features or changes to the old features being explained. All the features described in Chapter 2 stay valid unless stated otherwise in the chapter.

4.1 Course Management

Version 2.0, in addition to functionalities from 1.0, supports assigning multiple instructors or teaching assistants to one course (Refer to 4.1.1), creating a cross-listed course (Refer to 4.1.2) and editing an existing course to assign multiple instructors (Refer to 4.1.3).

4.1.1 Add New Course

In version 2.0, while adding a new course it is not mandatory to specify instructor(s) or the semester for a course. The reason for this change is if the instructor wishes to cross-list that course with another course then he/she could specify the instructor and the semester for the new cross-listed course.

Administrator can also specify teaching assistants (TA) for a course who can assist the instructor in managing the course. The TAs will have the same control over a course as an instructor.

The administrator would be able to assign multiple instructors to a course. The reason for this feature is that one course can have an instructor and an add-on instructor who needs to monitor the PLA process. All instructors added to the course would have the same control over a course.

- (1) The system will display a page to capture the following options after the system administrator selects 'Add new course' option:
 - School/college/university where the course is offered
 - Course number
 - Course section
 - Course name
 - Instructor (optional)
 - Teaching assistants (optional)
 - Semester (optional)
- (2) The system will display a list of all instructors registered for the school allowing the system administrator to select multiple instructors for the course.

- (3) The system will display a list of all instructors registered for the school allowing the system administrator to associate multiple teaching assistants to the course.
(Teaching assistants are separated from instructors because artifact reminder preferences can be set so that reminders are sent to teaching assistants and not instructors.)
- (4) The system will display a warning message “You did not select an instructor for the course. Are you sure? Yes/No” if the administrator submits the course information without selecting instructor(s) for the course.
- (5) The system will, after positive confirmation, add the course details to the database and display a confirmation message to the system administrator.

4.1.2 Create a ‘Cross-Listing’ Course

Very often many courses are merged to create one single course. For example, the electrical engineering department and computer engineering department can offer a single course which shares the same fundamentals.

This feature allows the administrator to create such courses by selecting multiple existing courses to be merged. A ‘cross-listed’ course is just like a normal course only it is created by merging two or more courses from different departments or different schools.

- (1) The system will capture the following information from the administrator:
 - Course alias
 - Description (optional)
 - Courses to be merged
 - School/college/university where the cross-list course is offered

- Instructor
 - Teaching assistants
 - Semester
- (2) The system will display the message “Please enter course alias” and move focus to the course alias field if the system administrator has left the course alias field blank.
- (An alias is a name given to the resultant (cross-listed) course after merging of 2 or more courses.)*
- (3) The system will ensure that the course alias is not more than twenty (20) alphanumeric characters.
- (4) The system will ensure that the course alias is unique i.e. not already assigned to some other cross-list course.
- (5) The system will display all courses registered in the system allowing the administrator to select courses which need to be merged.
- (A “cross-listed” course can be made up by merging 2 or more courses.)*
- (6) The system will display a list of all registered school names, allowing the administrator to choose one.
- (7) The system will display a list of all instructors registered in the system allowing the system administrator to select multiple instructors.
- (8) The system will display the message “Please select the instructor for the course” and move focus to the instructor field if the system administrator has not selected an instructor for the course.
- (9) The system will display a list of all instructors registered for the school allowing the system administrator to associate multiple teaching assistants to the course.

- (10) The system will allow the administrator to enter the semester.
- (11) The system will display the message “Please enter a semester” and move focus to the semester field if the system administrator has left the semester field blank.
- (12) The system will ensure that the semester field is not more than fifty (50) alphanumeric characters.
- (13) The system will add the course details to the database and display a confirmation message to the administrator.

4.1.3 Edit Course

The administrator can edit all the information mentioned in 4.1.1(1) with the difference from version 1.0 being that the administrator can assign multiple instructors or teaching assistants to a course.

4.2 Assignment Management

In version 2.0, new features have been added to ‘Assignment Management’ to help the instructor manage the assignments in the course better. Assignments, in version 2.0, can be associated with group sets to support group activity (Refer to 4.2.1) and can be configured to hide prior corresponding ‘evaluation’ artifacts from evaluators.

A new feature ‘Duplicate assignment’ (Refer to 4.2.2) is provided to the instructor/TA if he/she wishes to create an assignment with similar configurations as one created previously.

4.2.1 Add Assignment

In addition to version 1.0, in version 2.0 a group set can be assigned to an assignment to support group activity. A group can be created by including 2 or more students in it and assigning it a responsibility of an artifact.

The instructor/TA can also select an additional option to hide prior corresponding evaluations in the assignment. If this option is selected the evaluator will not be able to view corresponding prior evaluations for the assignment till he/she submits his/her own evaluation. An instructor/TA may wish to do so if he/she thinks that evaluations from other participants would influence the evaluations.

(1) The system will capture the following information from the user in addition to the information described in 2.10.2(1):

- Associate group set (optional)

(2) The system will display a list of all group sets created in the course, allowing the instructor to select one group set to be associated with the assignment.

(A group set has many groups with each group having several students as its members. However 1 student can be a member of only 1 group within a group set.

This ensures that all students in the course are members of a group.

It is possible to associate only one group set with an assignment. If the responsibility of an artifact is assigned to a group then a group will be chosen from the assigned group set.)

(3) The system will display an option 'Hide prior corresponding evaluations' which the instructor/TA can select.

(If this option is selected then all evaluations submitted prior in the same workflow for the same artifact would be hidden till the evaluator submits his/her 'evaluation' artifact.)

- (4) The system will perform all steps described from 2.10.2(1) to 2.10.2(17).

4.2.2 Duplicate Assignment

The instructor can select an existing assignment and duplicate it to create a similar assignment without having to go through all the steps of creating a new assignment. All the artifacts assigned to the assignment would also be duplicated. The instructor would have to specify the start date of the new assignment and a new (duplicate) assignment would be created with similar artifacts as in the original with similar time increments. It should be noted that only the schema of the assignment will be duplicated but the student allocations to the workflow instances of the assignment would not be duplicated. They will be generated by the system afresh as it would have done for a new assignment.

For example, an instructor conducts three similar online quizzes during the semester. The instructor can just create an assignment for the first quiz, create and assign artifacts to it and then duplicate the assignment to create two new assignments for quizzes 2 and 3 without having to go through the steps of creating a new assignment and artifacts twice.

- (1) The system will display a page to capture the following information from the instructor:
- New assignment name
 - Description
 - Start date

- Start time
- (2) The system will restrict the assignment name to fifty (50) alphanumeric characters.
 - (3) The system will display the message “Please enter the assignment name” and move focus to the assignment name field if the instructor has left the assignment name blank.
 - (4) The system will check for an assignment with the same name in the current course, and display the message “Assignment with this name already exists, please enter another name” if assignment name already exists and move focus back to the assignment name field.
 - (5) The system will restrict the combination of start date and start time to be later than the current system time.
 - (6) The system will display the message “Start date and time should be later than the current date and time” and move the focus to the ‘Start Date’ field if the start date in an invalid range.
 - (7) The system will automatically create new artifacts to duplicate the assignment with similar time increments and display a confirmation message to the instructor.

4.3 Reminder Preferences

In version 1.0, students and instructors both had common preferences. In version 2.0, different set of reminder preferences can be set for instructor, students and TA’s.

The instructor/TA can set a different set of reminders for the instructor, TA and student user types. The instructor/TA would be able to set the same options mentioned in (2) below with different values for all three user types. It should be noted that reminder

settings for ‘TA’ user type would be applicable to all TAs for the course. The same principle applies to ‘student’ and ‘instructor’ user types as well.

The reason for such flexibility is that the instructor wouldn’t want to be reminded every time an artifact begins or ends. However, he/she might want the TA and the students to be aware of that. He/she just wants to know when there are late posters. In such a scenario, he/she could set different preferences for students, TA and himself/herself.

The functionality for setting preferences is explained below:

- (1) The system will retrieve the current reminder preferences and display them in the appropriate options mentioned below.
- (2) The system will display three groups of ‘Reminder Triggers’ options mentioned below, one each for the students, instructors and TAs:
 - X days before end date
 - X days after end date (for late posters)
 - Maximum number of reminders after end date (for late posters)
 - As soon as the artifact period starts
 - As soon as the artifact period ends
 - As soon as the artifact is submitted
- (3) The system will allow any one (1) or all options to be selected under ‘Reminder Triggers’
- (4) The system will save the preferences selected by the user to the database overwriting the current preferences.
- (5) The system will display a confirmation message to the user.

4.4 Manage Messages

This functionality allows all users to manage their unread messages for a particular course. The users can select any of the options described in (1) below to mark their messages 'read' automatically.

This option would be accessible to the student once he/she logs in to the course however for the instructor/TA it will be accessible under the 'Administer' option (Refer to 2.6.7).

The functionality for configuring message options is explained below:

(1) The system will display the following options once the user selects 'Manage messages':

- Mark message read automatically

(To mark an individual message 'read' as and when it is read.)

- Mark thread read automatically

(To mark the entire message thread 'read' as and when a message in the thread is read.)

- Mark discussion area read automatically

(To mark the entire discussion area 'read' as and when a message in the discussion area is read.)

(2) The system will allow the user to select any or all of the above option(s).

(3) The system will save the options selected for the course and display a confirmation message to the user.

4.5 Artifact Management

Version 2.0 includes the feature to assign the responsibility of submitting an artifact to the TA and a group in addition to individual students and instructors.

The monotonous step of entering points value for questions for each ‘evaluation’ artifact is automated. (Refer to 2.11.2(29))

4.5.1 Add New Artifact (Only for Instructor/TA)

In version 1.0, the instructor/TA would have to specify points for all questions for all ‘evaluation’ artifacts. However, in version 2.0 the instructor/TA would enter the points value for the first ‘evaluation’ artifact and the system would fill up the values automatically for the remaining ‘evaluation’ artifacts once they are assigned the same ‘artifact to evaluate’.

This would save the instructor/TA the monotonous step for adding point values for each artifact. For example, if the instructor/TA has to create 3 ‘evaluation’ artifacts then he/she would have to just enter points value for questions only for the first ‘evaluation’ artifact and the system will fill up the values for the next 2 artifacts automatically once they are assigned the same ‘artifact to evaluate’ as the first one.

The PLAE version 2.0 supports students working in groups. Hence, the responsibility for an artifact can be assigned to a group. When a group is assigned with a responsibility, all group members would be responsible for the artifact. On the contrary if a group is not supposed to be associated with an artifact then all group members would be inhibited from being associated with that artifact.

The responsibility for an artifact can also be assigned to a TA.

- (1) The system will display the following options to allow the instructor/TA to select the participant to whom the role has to be associated with, (the default selection being the students):

- Instructor

(The responsibility of an artifact can be assigned to instructor(s) of the course.)

- TA

(The responsibility of an artifact can be assigned to TA(s) of the course.)

- Students

(The responsibility of an artifact can be assigned to student(s) of the course.)

- (2) The system will display an additional option 'Group' if the assignment to which the artifact belongs supports group activity (Refer to 4.2.1(1)).

(The responsibility of an artifact can be assigned to a group from the group set assigned to the assignment.)

For 'Evaluation' Artifact

- (3) The system will display all artifacts for the assignment whose value for maximum evaluations has not reached, allowing the instructor to select 'The artifact to evaluate'.

(If the artifact is an 'evaluation' artifact then the instructor can assign the artifact which needs to be evaluated. Any artifact can be evaluated and awarded a grade to.)

- (4) The system will restrict the number of evaluations per artifact to the value set in 'grading preferences' in 4.6(3).

- (5) The system will display an option 'Hide prior evaluations' which the instructor/TA can select.

(If this option is selected then all evaluations submitted prior in the same workflow for the same artifact would be hidden till the evaluator submits his/her 'evaluation' artifact.)

- (6) The system will, depending upon the number of questions supported by the assignment's template set, allow the instructor to specify the point value for each question.

(For example, if the template set supports two essay type questions then the instructor would have to enter the number of points each question is worth.)

- (7) The system will fill up the values for a new 'evaluation' artifact if it is supposed to evaluate an artifact which already has been assigned an evaluation artifact.

(This would save the instructor/TA the monotonous step for adding point values for each artifact. For example, if the instructor/TA has to create 3 'evaluation' artifacts then he/she would have to just enter points value for questions only for the first 'evaluation' artifact and the system will fill up the values for the next 2 artifacts automatically once they are assigned the same 'artifact to evaluate' as the first one.)

- (8) The system will save the artifact upon submission and display a confirmation message to the instructor/TA.

4.5.2 Submit Artifact

In addition to the options described in 2.11.5, the submit artifact feature would also support the feature to submit a file (Refer to 4.7(7)) as artifact if it is defined so by the template of the artifact being submitted.

4.6 Automatic Evaluation Tool

In version 1.0, maximum two evaluations were supported to evaluate an artifact. However, version 2.0 would allow the user to set the maximum number of evaluations to evaluate an artifact to any number he/she wishes.

The instructor/TA can set the number of maximum evaluations per artifact which would restrict the maximum number of evaluations to the specified value. For example, if the instructor/TA sets a value of 4 for the maximum evaluations then an artifact can have no more than 4 evaluations.

The functionality for setting the maximum number of evaluations is explained below:

- (1) The system will display an option 'Maximum evaluations allowed'.
- (2) The system will allow the user to enter a value for the option 'Maximum evaluations allowed' with the default value being two (2).
- (3) The system will restrict the values entered for 'Maximum evaluations allowed' to be two (2) numeric characters.
- (4) The system will save the preferences to the database and display a confirmation message.

4.7 Template Management

In addition to functionalities ‘View template set’ and ‘View template layout’ from version 1.0, version 2.0 would allow the user to create their own template sets to be used in assignments. This functionality is only available to the instructor/TA.

The tool to create template sets in PLAE is called a ‘template wizard’. The template wizard guides the instructor/TA through the process of creating a new template set by asking a series of questions. The wizard would work through a series of screens where the questions asked on the next screen would depend on the answers submitted on the current screen.

The functionality of the template wizard is explained below:

- (1) The system will display “Template set name” allowing the user to enter a name for the template set.
- (2) The system will display the message “Please enter the template set name” and move focus to the template set name field if the user has left it blank.
- (3) The system will restrict the template set name to one hundred (100) alphanumeric characters.
- (4) The system will display “Template set description” allowing the user to enter a description for the template set.
- (5) The system will display the question “How many sections do you want in the template?”
- (6) The system will, depending upon the number of sections entered, display “What type of section is section n ?” where n refers to the section number.
- (7) The system will display the following options for each section:

- Subjective (Essay type)
- Objective (Multiple choice)
- File submission

(The user can make a template with various sections with each section being different for subjective (essay-type), objective (multiple choices) questions or file submission. When file submission is selected, the student would be allowed to submit a file as an artifact.)

- (8) The system will, for each section, display the following questions depending upon the section type.

For objective sections

- (9) The system will display the question “How many questions do you want in this section?” allowing the user to enter a value.

(The user can set the number of questions to be displayed per section.)

- (10) The system will restrict the entered value to two (2) numeric characters.

- (11) The system will display the question “How many options per question?” allowing the user to enter a value.

(An option is a possible answer choice. The user can set any number of options per question.)

- (12) The system will restrict the entered value to two (2) numeric characters.

- (13) The system will display the question “Can multiple options be selected per question?”

- (14) The system will display the following two exclusive options as answers to the question mentioned in (13):

- Yes
- No

(If the user answers 'yes' to this question then it means more than one answer can be selected for the question.)

For subjective sections

- (15) The system will display the question “How many questions do you want in this section?” allowing the user to enter a value.
- (16) The system will restrict the value entered to two (2) numeric characters.
- (17) The system will display the question “What is the approximate length of the answers?”
- (18) The system will display the following exclusive options as answers to the question mentioned in (17):
- Short
 - Medium
 - Long

(This is important as it will help determine the size of the visual element to be displayed to the student to submit the answer.)

For file submission sections

- (19) The system will implicitly define an instructions area in the template and an option to submit a file.
- (For 'file submission' sections, the student will be allowed to enter comment/instructions and upload a file while submitting the 'problem' or 'solution' artifact)*

Final step

(20) The system will display “Enter a description for the ‘problem’ artifact template” allowing the user to enter a description for the ‘problem’ artifact template.

(21) The system will display “Enter a description for the ‘solution’ artifact template” allowing the user to enter a description for the ‘solution’ artifact template.

(Descriptions entered in (20) and (21) would be a general description of the ‘problem’ and ‘solution’ artifact template, respectively which will be displayed within the template set so that the instructor/TA could read the description to get a better understanding of the template layout.)

(22) The system will display “For the ‘review’ artifact, do you want a common comment area or one comment area per element?”

(23) The system will display the following exclusive options as answers to the question mentioned in (22) with ‘common comment area’ selected by default:

- Common comment area
- One comment area per element

(This option allows the instructor to decide if a different element should be displayed per question to record the comments or a common element should be displayed to record all comments for all elements together.)

(24) The system will display “Enter a description for the ‘review’ artifact template” allowing the user to enter a description for the ‘review’ artifact template.

(25) The system will display “For the ‘evaluation’ artifact, do you want a common comment area or one comment area per element?”

(26) The system will display the following exclusive options as answers to the question mentioned in (25) with ‘common comment area’ selected by default:

- Common comment area
- One comment area per element

(This option allows the instructor to decide if a different element should be displayed per question to record the comments or a common element should be displayed to record all comments for all elements together.)

(27) The system will display “Enter a description for the ‘evaluation’ artifact template” allowing the user to enter a description for the ‘evaluation’ artifact template.

(28) The system will, for the ‘manual grade’ artifact, display “Show ‘evaluation’ artifact grades while manual grading?”

(29) The system will display the following exclusive options as answers to the question mentioned in (28) with ‘Yes’ selected by default:

- Yes
- No

(30) The system will display “Enter a description for the ‘manual grade’ artifact template” allowing the user to enter a description for the ‘manual grade’ artifact template.

(31) The system will display “Enter a description for the ‘dispute’ artifact template” allowing the user to enter a description for the ‘dispute’ artifact template.

(32) The system will display “Enter a description for the ‘arbitration’ artifact template” allowing the user to enter a description for the ‘arbitration artifact template.

- (33) The system will save the template set details and display a confirmation message to the user, on a successful submission.

4.8 Group Management

The instructor/TA can create a group set which is a set of groups from which a group can be chosen to be assigned with the responsibility of an artifact. These group sets can be created either manually or automatically based on certain criteria specified.

After selecting 'Group Management' under the 'Administer' option, the instructor/TA will see a list of all groups created based on the criteria specified by the instructor/TA. The following information will be displayed for each group: group name and group members. The system will display 'Create criteria-based groups' and 'Create groups manually'.

4.8.1 Create Criteria-Based Groups (Automatic)

This feature allows the user to create groups in a course based on the following criteria: gender, major, degree, nationality, course (for 'cross-listed' courses) and section (for 'cross-listed' courses).

- (1) The system will capture the following information from the user:

- Desired group size
- Group set name
- Groups include students with

(The instructor/TA can select the criteria for selecting students to form a group.)

- (2) The system will restrict the 'desired group size' to two (2) numeric characters.

- (3) The system will display the message “Please enter the desired group size” if the user has not entered any value for ‘desired group size’.
- (4) The system will ensure that the value for ‘desired group size’ is more than two (2).
- (5) The system will restrict the ‘group set name’ to twenty (20) alphanumeric characters.

(Group set name is an alias to the set which contains all groups created based on a common criteria. This group set name will be used to automatically name the groups within the group set. The format of a group name is ‘GroupSetName_No’ where ‘GroupSetName’ is the name of the set and ‘_No’ is a sequentially generated number.)

- (6) The system will display the message “Please enter the desired group set name” if the user has not entered any value for ‘group set name’.

For ‘Groups include students with’ option

- (7) The system will display a ‘Gender’ option for all courses with the following mutually exclusive options:

- Same gender

(Students of same gender will be included in the group.)

- Different gender

(Students of different gender will be included in the group.)

- (8) The system will display a ‘Major’ option for all courses with the following mutually exclusive options:

- Same major

(Students with same major will be included in the group.)

- Different major

(Students with different majors will be included in the group.)

- (9) The system will display a 'Degree' option for all courses with the following mutually exclusive options:

- Same degree

(Students on same level will be included in the group. For example all Masters students will be in one group)

- Different degree

(Students on different levels will be included in the group.)

- (10) The system will display a 'Nationality' option for all courses with the following mutually exclusive options:

- Same nationality

(Students of same nationality will be included in the group.)

- Different nationality

(Students of different nationalities will be included in the group. This option can be used to ensure national diversity in a group.)

- (11) The system will display the additional option of 'Course' for 'cross-listed' courses for the 'Groups include students with' option:

- Same course

(Students of the same course will be included in the group.)

- Different course

(Students from different courses will be included in the group.)

(12) The system will display the additional option of ‘Section’ for ‘cross-listed’ courses for the ‘Groups include students with’ option:

- Same section

(Students of the same section will be included in the group.)

- Different section

(Students from different sections will be included in the group.)

(13) The system will create groups once the user submits the information ensuring that the criteria specified above are adhered to the maximum possible extent.

(Sometimes it will not be possible to adhere to all criteria because of lack of diversity in students. In such an event, the system will adhere to the specified criteria as much as possible while creating groups.)

(14) The system will display the groups created with its members after successful creation of groups.

4.8.2 Create Groups (Manual)

This feature allows the user to create groups manually. The user can select the members to be included in each group. Manual group creation is a multiple step process. The user will be asked to create groups till all students in the course are assigned to a group.

(1) The system will capture the following information from the user:

- Group set name

(Group set name is an alias to the set which will contain all manually created groups.)

- Minimum number of students in a group

(This is the minimum number of students required in each group. Based on this value, the system will decide the maximum number of iterations necessary for creating manual groups. The system will iterate with manual group creation process till all students in the course are assigned to a group within a group set.)

- (2) The system will iterate to capture the following information from the user till all students in the course are assigned to a group within a group set:

- Group name

(The name of the group within the group set.)

- Select members

(The students selected to be the members of the group.)

- (3) The system will ensure that the user selects at least the minimum number of students specified in (1) for each group.
- (4) The system will save the groups to the database and display a confirmation message to the user.

4.9 Student Registration

In version 2.0, while registering students are asked to enter some optional information which is used while creating groups by the instructor/TA. This feature is similar to 2.7.1 except for the additional information which a student needs to enter.

The functionality for registering with PLAE is explained below:

- (1) The system will capture the following information from the user in addition to information mentioned in 2.7.1(1).
 - Gender

- Major
 - Degree
 - Nationality (Optional)
- (2) The system will display the following mutually exclusive options for the 'Gender' option:
- Male
 - Female
- (3) The system will display a list of majors allowing the user to select one.
- (4) The system will display the following mutually exclusive options for the 'Degree' option:
- Bachelors
 - Masters
 - Ph.D
- (5) The system will add the student to the database after submission of all information by the user and display a confirmation message.
- (6) The system will send an email to the student with the username and password which can be used to login to the system.

4.10 Student Allocation Tool

This functionality randomly allocates the responsibility of an artifact to students, groups, instructor or TA as chosen by the instructor/TA while creating an artifact (Refer to 4.5.1(1)). The system will ensure that rules created by the instructor or TA for allocating roles are observed (Refer to 2.11.2(15) and 2.11.2(16))

4.10.1 Manual Generation of Student Allocations (Only for Instructor)

The system will perform the same steps for allocating students as in version 1.0. However, for version 2.0 the system can also allocate groups, multiple instructors and TAs to artifact responsibilities in addition to students.

If the allocations are to be done to instructor or TA and there are multiple instructors or TAs then these allocations will be done proportionately, which means all instructors or TAs will be assigned with equal number of responsibilities.

- (1) The system will display the list of assignments with option 'Generate student allocations' for each assignment whose start date and time are later than current start date and time or whose student allocations have not yet been generated and saved.
- (2) The system will randomly choose students for the roles associated with the artifacts within the assignment.
- (3) The system will ensure that the rules created by the instructor for each artifact are followed.
- (4) The system will divide the artifacts proportionately to the number of TAs or instructors in case there are multiple TAs or instructors.

(For example, if there are 2 TAs then each TA would receive half the number of artifacts. Similar rationale applies in case multiple instructors as well.)

- (5) The system will select groups to assign responsibilities of artifacts from the group set associated with the assignment.
 - (6) The system will not associate individual students with conflicting group artifacts.
- (For example, 'problem' artifact is to be done by Group 1 which has A, B, C as group members and 'solution' artifact is supposed to be done by an individual who*

was not involved in the problem creation. System will ensure that A, B or C are not selected to do the 'solution' artifact.)

- (7) The system will display all allocations for the entire assignment to the instructor.
- (8) The system will display an option 'Regenerate student allocations' to allow the instructor to regenerate the student allocations if he/she is not satisfied with the allocations.
- (9) The system will display an option 'Save student allocations' to allow the instructor save the student allocations in the system.
- (10) The system will create workflows for the assignment once the student allocations are generated and saved

4.10.2 Automatic Generation of Student Allocations (Only for System)

This feature will generate the allocations for artifacts within an assignment automatically on the start date and time of the assignment if manual allocations have not been done already.

- (1) The system will perform steps 4.10.1(2) to 4.10.1(6) to allocate artifact responsibilities.
- (2) The system will save the allocations to the database.
- (3) The system will create workflows for the assignment once the allocations are generated.

CHAPTER 5

DATABASE AND CLASS DESIGNS FOR PLAE VERSION 2.0

5.1 Entity Relationship Diagram

Figure 5.1 is the ER diagram for PLAE version 2.0 based on the requirements described in Chapter 4. The new and changed entities and relationships are indicated by ellipses.

Entity attributes in **bold** in the diagram indicate that they are required attributes for that entity. The acronym PK indicates the primary key of the entity which identifies that entity and FK indicates foreign key which identifies the relationship between two entities. The new entities are highlighted with dashed (---) ellipses and modified entities are highlighted by dotted (...) ellipses.

Following is a description of all the entities and their attributes in the diagram.

(1) School:

This entity represents the schools registered with the system. The entity schema is same as Table 3.1.

(2) User:

This entity defines the attributes of the users associated with PLAE.

Table 5.1 User Entity

Attribute name	Data type	Required	Description	Value range
username (PK)	Varchar(20)	Yes	Uniquely identifies a user. Either automatically generated by the system or entered by the user. The system will contain one default user – “administrator”	N/A
password	Varchar(20)	Yes	Password chosen by the user to access the system.	N/A
primary_email	Varchar(100)	Yes	Primary email address of the user.	N/A
secondary_email	Varchar(100)	No	Secondary email address of the user.	N/A
firstname	Varchar(50)	Yes	First name of the user.	N/A
middle_intials	Varchar(10)	Yes	The middle initials of the user	N/A
lastname	Varchar(50)	Yes	Last name of the user.	N/A
reg_date	Datetime	Yes	The date the user registered with the system.	N/A

Table 5.1 User Entity (Continued)

Attribute name	Data type	Required	Description	Value range
res_phone_no	Varchar(15)	No	Residence phone number of the user. This is only stored for the 'student' user.	N/A
mobile_no	Varchar(15)	No	Mobile phone number of the user. This is only stored for the 'student' user.	N/A
office_no	Varchar(15)	No	Office phone number of the user. This is only stored for the 'student' user.	N/A
gender	Char(1)	No	The gender of the user. This attribute will be used only for a 'student' user.	Only two possible values: 'M' or 'F' for male or female, respectively.
major_id (FK1)	Integer	No	The major of the user. This attribute will be used only for a 'student' user.	N/A
degree	Char(1)	No	The degree pursued by the user. This attribute will be used only for a 'student' user.	Three possible values: 'B', 'M', or 'P' for bachelors, masters or Ph.D's, respectively.
nationality	Varchar(100)	No	The nationality of the user. This attribute will be used only for a 'student' user.	N/A

(3) Instructor_school:

This entity represents the instructor-school associations. The entity schema is same as Table 3.3.

(4) Student_school:

This entity represents the student-school associations. The entity schema is same as Table 3.4.

(5) School_contact:

This entity represents contact details of a school representative. The entity schema is same as Table 3.5.

(6) Course:

This entity represents all the courses registered with the system.

Table 5.2 Course Entity

Attribute name	Data type	Required	Description	Value range
course_id (PK)	Integer	Yes	Unique number which identifies the course registered with the system.	N/A
course_no	Varchar(20)	Yes	A number used by the school to identify the course.	N/A
section	Varchar(20)	Yes	The section number of the course.	N/A
description	LongText*	No	The description of a course. This attribute will be used only in case of 'cross-listed' course.	N/A
name	Varchar(100)	Yes	The name of the course.	N/A
semester	Varchar(50)	Yes	The semester when the course is offered.	N/A
school_id (FK3)	Integer	Yes	The school id of the school offering the course.	N/A
merged	Char(1)	Yes	A flag value to indicate if the course is a 'cross-listed' course.	Two possible values: 'Y' or 'N' for Yes or No, respectively.

*LongText is a data type supported by Microsoft Access

(7) Admission:

This entity represents the admission status of students in a course. The entity schema is same as Table 3.8.

(8) Template_Sets:

This entity represents all the templates saved in the system.

Table 5.3 Template_Sets Entity

Attribute name	Data type	Required	Description	Value range
template_set_id (PK)	Integer	Yes	Unique number to identify a template set.	N/A
template_set_name	Varchar(100)	Yes	A name for the template set.	N/A
template_set_desc	Longtext*	No	A template description which describes the template set.	N/A
course_id(FK1)	Integer	Yes	The course to which the template set belongs.	N/A

* Longtext is a datatype supported by Microsoft Access

(9) Templates:

This entity represents all the artifact templates within a template set. The entity schema is same as Table 3.10.

(10) Element_Types:

This entity represents the master element_types. The entity schema is same as Table 3.11. This entity comes preloaded with data. Refer to Appendix A to see the data.

(11) Template_Definition:

This entity represents all the definition of each template within a template set.

Table 5.4 Template_Definition Entity

Attribute name	Data type	Required	Description	Value range
element (PK)	Integer	Yes	The unique id for a template element.	N/A
parent_element (FK2)	Integer	Yes	The element id with which it needs to be grouped together. For example, the four multiple choice options of a question would have the question element as the parent element. The default value is -1 which means there is no parent element.	N/A
element_type_id (FK1)	Integer	Yes	The element type.	N/A
template_id (FK3)	Integer	Yes	The template to which the element belongs.	N/A
section_no (FK3)	Integer	Yes	The template section to which the element belongs.	N/A

(12) Discussion_Messages:

This entity represents the messages posted by the users in the discussion area. The entity schema is same as Table 3.13.

(13) User_Messages_Read:

This entity stores all the messages marked read by a user. The entity schema is same as Table 3.14.

(14) Assignment:

All the assignments created in each course are represented by this entity.

Table 5.5 Assignment Entity

Attribute name	Data type	Required	Description	Value range
assig_id (PK)	Integer	Yes	A unique identification number for each assignment.	N/A
assig_name	Varchar (50)	Yes	The name of the assignment.	N/A
assig_desc	Longtext*	No	The description of the assignment.	N/A
start_date_time	Datetime	Yes	The date and time when the assignment should start.	N/A
end_date_time	Datetime	Yes	The date and time when the assignment should end.	N/A
initiation_type	Char(1)	Yes	The initiation type of the assignment.	This attribute can have only 2 possible values: 'E' or 'T' for 'event-based' or 'time-based', respectively.
grading_algo	Varchar(10)	Yes	The grading algorithm to be used to determine the final grade.	This attribute has three possible values: 'Highest', 'Lowest' or 'Average' to indicate highest grade, lowest grade or average of grades algorithms, respectively.
threshold_value	Integer	Yes	The threshold value on which manual intervention is necessary. The default value is -1 which indicates manual intervention is not required.	N/A
course_id (FK1)	Integer	Yes	The course to which the assignment belongs to.	N/A
template_set_id (FK2)	Integer	Yes	The template set to be used with the assignment.	N/A

Table 5.5 Assignment Entity (Continued)

Attribute name	Data type	Required	Description	Value range
hide_prior_eval	Char(1)	No	The flag to indicate if the prior evaluations submitted should be hidden from the user for this assignment.	Two possible values: 'Y' or 'N' for yes or no, respectively.

*Longtext is a datatype supported by Microsoft Access

(15) Artifact:

This entity represents the artifacts created within an assignment. The entity schema is same as Table 3.16.

(16) Artifact_Roles_NotAllowed:

This entity represents the role players that should not be associated with the artifact. The entity schema is same as Table 3.17.

(17) Evaluation_Artifact_Points_Breakdown:

This entity represents breakdown of points of an artifact being evaluated during an evaluation. The entity schema is same as Table 3.18.

(18) Artifact_Submissions:

This entity represents the artifacts submitted by the users or groups.

Table 5.6 Artifact_Submissions Entity

Attribute name	Data type	Required	Description	Value range
submission_no (PK)	Integer	Yes	The unique identification number of the submission.	N/A
date_submitted	Datetime	Yes	The date the artifact is submitted.	N/A
element (FK4)	Integer	Yes	The input element used to submit the artifact.	N/A
value	Longtext*	Yes	The value of the element submitted by the user.	N/A
username (FK1)	Varchar(20)	No	The user who submitted the artifact.	N/A
artifact_id (FK2)	Integer	Yes	The id of the artifact being submitted.	N/A
group_id(FK4)	Integer	No	The id of the group that submitted the artifact.	N/A
lock_status	Char(1)	Yes	To indicate whether the submitted artifact is locked.	There are only 2 possible values: 'L' or 'U' for 'Locked' or 'Unlocked', respectively. The default value is 'L'.

*Longtext is a datatype supported by Microsoft Access

(19) Workflow:

This entity represents the participant allocations to artifacts and the state of the workflows in a course.

Table 5.7 Workflow Entity

Attribute name	Data type	Required	Description	Value range
workflow_id (PK)	Integer	Yes	A unique identification number for each workflow.	N/A
workflow_no	Integer	Yes	The workflow number in a particular course for each student. For every assignment, there will be same number of workflows as the number of students. For example, if there are 20 students there will be 20 workflows for an assignment.	N/A
workflow_step	Integer	Yes	The step number within each workflow. Each artifact is represented by a step in a workflow. For example, if there are 5 artifacts in an assignment then there are $20 \times 5 = 100$ workflow steps in all.	N/A
assig_id (FK2)	Integer	Yes	The assignment to which the workflow belongs.	N/A
artifact_id (FK1)	Integer	Yes	The id of the artifact which is being represented as workflow step.	N/A
username(FK3)	Varchar(20)	No	The user responsible for the workflow step.	N/A
group_id (FK4)	Integer	No	The group responsible for the workflow step.	N/A

(20) Reminder_Preferences:

This entity represents the reminder preferences set for each artifact and in general.

Table 5.8 Reminder_Preferences Entity

Attribute name	Data type	Required	Description	Value range
artifact_id (FK1)	Integer	Yes	Unique identification number of the artifact. Default value is -1 which means default reminder settings.	N/A
Course_id	Integer	Yes	A numeric value which indicates which course the default reminder preferences apply to.	N/A
user_type	Integer	Yes	A value to indicate to whom the reminder preferences belong.	Three possible values: 'S', 'I' or 'T' to indicate students, instructors or teaching assistants, respectively.
days_before_end_date	Integer	No	A numeric value to indicate the number of days before the artifact due date the reminder should be sent. The default value is 0.	N/A
days_after_end_date	Integer	No	A numeric value to indicate the number of days after the artifact due date the reminder should be sent. The default value is 0.	N/A
end_of_artifact	Integer	No	A flag to indicate if the reminder should be sent when the artifact ends. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
start_of_artifact	Integer	No	A flag to indicate if the reminder should be sent when the artifact starts. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
artifact_submitted	Integer	No	A flag to indicate if the reminder should be sent when the artifact is submitted. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.

Table 5.8 Reminder_Preferences Entity (Continued)

Attribute name	Data type	Required	Description	Value range
max_reminder_after_end_date	Integer	No	A numeric value to indicate the maximum number of reminders to be sent after the end date.	N/A

(21) Final_Grades

This entity represents final grades students receive after manual or automatic evaluation. The entity schema is same as Table 3.22.

(22) Template_Sections

This entity represents the sections within a template.

Table 5.9 Template_Sections Entity

Attribute name	Data type	Required	Description	Value range
template_id (PK, FK1)	Integer	Yes	The unique number identifying the template	N/A
section_no (PK)	Integer	Yes	The section number within the template. The section number for a template always starts with 1. Template_id and section_no form a composite primary key.	N/A
section_type	Char(1)	Yes	The type of section. There are three types of sections: subjective, objective, file submission.	Three permissible values: 'O', 'S' or 'F' to indicate objective, subjective or file submission sections, respectively.
no_of_questions	Integer	No	The number of questions in a section in case of subjective and objective sections. For file submission sections, this value will always be 1.	N/A
no_of_options_per_question	Integer	No	The number of options per question in case of objective sections.	N/A

Table 5.9 Template_Sections Entity (Continued)

Attribute name	Data type	Required	Description	Value range
answer_length	Integer	No	The answer length in case of subjective questions.	Three permissible values: 'S', 'M' or 'L' to indicate short, medium or long answers, respectively.
multiple_options_selectable	Char(1)	No	A flag value to indicate if multiple options in an objective section can be selected.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.

(23) Course_Max_Eval

This entity represents the maximum number of evaluations allowed in a course.

Table 5.10 Course_Max_Eval Entity

Attribute name	Data type	Required	Description	Value range
course_id (PK, FK1)	Integer	Yes	The unique number identifying the course.	N/A
max_no_eval	Integer	Yes	The maximum number of evaluations in a particular course. The default value is 2.	N/A

(24) Course_TA

This entity represents the teaching assistants for a course.

Table 5.11 Course_TA Entity

Attribute name	Data type	Required	Description	Value range
course_id (FK1)	Integer	Yes	A number identifying a course.	N/A
username (FK2)	Varchar (20)	Yes	The username of the teaching assistant.	N/A

(25) Course_Instructors

This entity represents the instructors for a course.

Table 5.12 Course_Instructors Entity

Attribute name	Data type	Required	Description	Value range
course_id (FK1)	Integer	Yes	A number identifying a course.	N/A
username (FK2)	Varchar (20)	Yes	The username of the instructor.	N/A

(26) Course_Merger

This entity represents the courses merged to create a 'cross-listed' course.

Table 5.13 Course_Merger Entity

Attribute name	Data type	Required	Description	Value range
course_id (FK1)	Integer	Yes	A number identifying the 'cross-listed' course.	N/A
course_merged	Integer	Yes	A number identifying the course merged.	N/A

(27) Group_Members

This entity represents the members that make up a group.

Table 5.14 Group_Members Entity

Attribute name	Data type	Required	Description	Value range
group_id (FK1)	Integer	Yes	A number identifying the group.	N/A
username (FK2)	Varchar (20)	Yes	The username of the instructor.	N/A

(28) Groups

This entity represents the student groups created in a group set.

Table 5.15 Groups Entity

Attribute name	Data type	Required	Description	Value range
group_id (PK)	Integer	Yes	A unique number identifying a group.	N/A
group_set_id (FK1)	Integer	Yes	The group set id to which the group belongs to.	N/A
group_name	Varchar (20)	Yes	The name of the group.	N/A

(29) Groups_Sets

This entity represents the group sets created in a course.

Table 5.16 Group_Sets Entity

Attribute name	Data type	Required	Description	Value range
group_set_id (PK)	Integer	Yes	A unique number identifying a group set.	N/A
course_id (FK1)	Integer	Yes	The course to which group set belongs.	N/A
group_size	Integer	Yes	The minimum number of students in each group.	N/A
group_type	Char(1)	Yes	A value to indicate if the group set was created automatically or manually.	Two possible values: 'A' or 'M' to indicate automatic or manual group sets, respectively.
gender	Integer	No	A flag value to indicate if 'gender' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same gender' criterion used and 'different gender' criterion used, respectively.

Table 5.9 Group_Sets Entity (Continued)

Attribute name	Data type	Required	Description	Value range
major	Integer	No	A flag value to indicate if 'major' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same major' criterion used and 'different major' criterion used, respectively.
degree	Integer	No	A flag value to indicate if 'degree' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same degree' criterion used and 'different degree' criterion used, respectively.
nationality	Integer	No	A flag value to indicate if 'nationality' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same nationality' criterion used and 'different nationality' criterion used, respectively.
course	Integer	No	A flag value to indicate if 'course' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same course' criterion used and 'different course' criterion used, respectively.
section	Integer	No	A flag value to indicate if 'section' criterion was used to create groups.	Three possible values: (-1, 0, 1) to indicate criterion not used, 'same section' criterion used and 'different section' criterion used, respectively.

(30) Major

This entity represents the majors saved in the system.

Table 5.17 Major Entity

Attribute name	Data type	Required	Description	Value range
major_id (PK)	Integer	Yes	A unique number identifying a major.	N/A
major_name	Varchar(100)	Yes	The major name.	N/A

(31) User_Message_Preferences

This entity represents the personal preferences for managing messages for each user for a course.

Table 5.18 User_Messages_Preferences Entity

Attribute name	Data type	Required	Description	Value range
username (FK1)	Varchar(20)	Yes	The username of the user	N/A
course_id	Integer	Yes	The number representing the course.	N/A
message_read	Char(1)	Yes	A flag value to indicate if a message should be marked 'read' automatically. The default value is 'N.'	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
thread_read	Char(1)	Yes	A flag value to indicate if a thread should be marked 'read' automatically. The default value is 'N.'	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.
discussion_read	Char(1)	Yes	A flag value to indicate if a discussion should be marked 'read' automatically. The default value is 'N'.	There are only 2 possible values: 'Y' or 'N' for 'Yes' or 'No', respectively.

(32) Experimenter_School:

This entity represents the experimenter-school associations. The entity schema is same as Table 3.6.

5.2 Class Diagram

Figure 5.2 is the class diagram for PLAE version 2.0 based on the requirements described in Chapter 4 and ER diagram above. The new classes are indicated by dashed (---) ellipses and modified classes are indicated by dotted (...) ellipses. The class diagram schema has not changed a lot to incorporate the requirements of version 2.0 because the functionality implemented within the classes will change. This helps in maintaining loose coupling between systems, if any system is integrated with PLAE, that system will not be required to change.

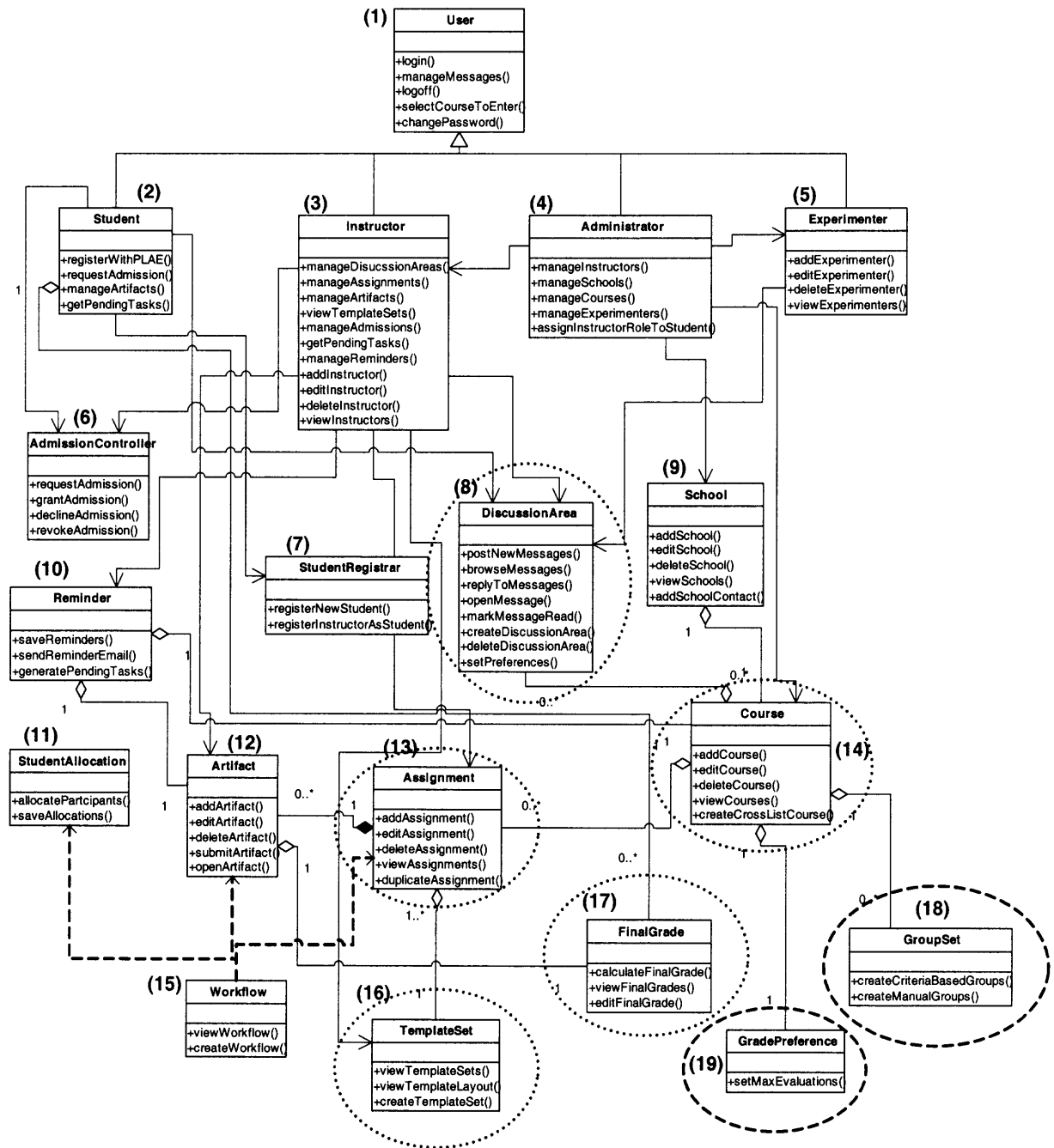


Figure 5.2 Class diagram for PLAE version 2.0.

The following section presents the classes involved and provides a brief description of the functions within each class.

(1) User:

This is a general class for all users. It includes functions common to all users. The class schema is same as Table 3.23.

(2) Student

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the student user. The class schema is same as Table 3.24.

(3) Instructor

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the instructor user. The class schema is same as Table 3.25.

(4) Administrator

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the administrator user. The class schema is same as Table 3.26.

(5) Experimenter

This is a child class of 'User'. It inherits all the functions described in the 'User' class and some additional functions specific to the experimenter user. The class schema is same as Table 3.27.

(6) AdmissionController

This class controls the admission of a student in a course. The class schema is same as Table 3.28.

(7) StudentRegistrar

This class controls the registration of a student in PLAE. The class schema is same as Table 3.29.

(8) DiscussionArea

This class manages the discussion areas in a course. The class has the following functions in addition to the functions described Table 3.30.

Table 5.19 Class DiscussionArea

Function name	Purpose	Requirement reference
postNewMessage()	To allow a user to a post a new message.	2.8.1
browseMessages()	To allow the user to browse through messages.	2.8.3, 2.8.4
replyToMessage()	To allow a user to reply a message.	2.8.2
openMessage()	To allow the user to open a message to read.	2.8.2(1)
markMessageRead()	To allow the user to mark a message read	2.8.5, 2.8.6
createDiscussionArea()	To allow the instructor to create a new discussion area.	2.8.8
deleteDiscussionArea()	To allow the instructor to delete a discussion area.	2.8.9
setPreferences()	To allow the user to set his/her preferences for 'read' messages.	4.4

(9) School

This class manages the schools in the PLAE system. The class schema is same as Table 3.31.

(10) Reminder

This class manages the reminder settings for a course and artifacts.

Table 5.20 Class Reminder

Function name	Purpose	Requirement reference
saveReminders()	To save the reminder preferences set by the instructor.	4.3
sendReminderEmail()	To send a reminder email according to the reminder settings by the instructor.	2.15.1
generatePendingTasks()	To retrieve the pending tasks for a student/instructor.	2.6.6

(11) StudentAllocation

This class manages the student allocations to artifacts. The class schema is same as Table 3.33.

(12) Artifact

This class manages the artifacts within an assignment. The class schema is same as Table 3.34.

(13) Assignment

This class manages the assignments within a course.

Table 5.21 Class Assignment

Function name	Purpose	Requirement reference
addAssignment()	To allow the instructor to create a new assignment.	4.2.1
editAssignment()	To allow the instructor to edit an assignment.	2.10.3
deleteAssignment()	To allow the instructor to delete an assignment.	2.10.4
viewAssignment()	To allow the instructor to view all assignments.	2.10.1
duplicateAssignment()	To allow the instructor to duplicate an assignment.	4.2.2

(14) Course

This class manages the courses in the PLAE system.

Table 5.22 Class Course

Function name	Purpose	Requirement reference
addCourse()	To allow the administrator to create a new course.	4.1.1
editCourse()	To allow the administrator to edit course details.	4.1.3
deleteCourse()	To allow the administrator to delete a course.	2.5.4
viewCourses()	To allow the administrator to view all courses.	2.5.1
createCrossListCourse()	To allow the administrator to create a 'cross-listed' course.	4.1.2

(15) Workflow

This class manages the workflows within a course. Workflow class is dependant on StudentAllocation, Artifact and Assignment classes. The class schema is same as Table 3.37.

(16) TemplateSet

This class manages template sets in PLAE version 2.0.

Table 5.23 Class TemplateSet

Function name	Purpose	Requirement reference
viewTemplateSets()	To allow the instructor to view all template sets available in the system.	2.12.1
viewTemplateLayout()	To allow the instructor to view the layout of a template within a template set.	2.12.2
createTemplateSet()	To allow the instructor to create a new template set.	4.7

(17) FinalGrade

This class calculates the final grades of a student. The class schema is same as Table 3.39.

(18) GroupSet:

This class creates manual groups or groups based on instructor-specified criteria.

Table 5.24 Class GroupSet

Function name	Purpose	Requirement reference
createCriteriaBasedGroups()	To allow the instructor to create a group set based on criteria.	4.8.1
createManualGroups()	To allow the instructor to create a manual group set.	4.8.2

(19) GradingPreference:

This class sets the grading preferences for a course.

Table 5.25 Class GradingPreference

Function name	Purpose	Requirement reference
setMaxEvaluations()	To allow the instructor to set the maximum number of evaluations supported for a course.	4.6

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Limitations

The requirements described in Chapters 2 and 4 do not describe non-functional requirements of the system. Non-functional requirements can be described as quality attributes of a system. Some example of such requirements would be performance, reliability, system response time, load balancing, to name a few. These requirements have been intentionally left out of this thesis due to lack of required statistics such as volume of users, number of concurrent users and hardware and software to be used for the system.

The designs presented in the previous chapters should not be considered as final, rather as a starting point for the implementation of PLAE since the designs have not been evaluated.

6.2 Future Extensions

Here are some extensions that should be considered for subsequent versions of PLAE that would enhance the overall functionality of the system:

- (1) The ability to allow students to submit an evaluation for any artifact in addition to their assigned 'evaluation' artifacts, perhaps for extra credit.
- (2) The instructor should be able to reject an artifact submission, compelling the student to work on the artifact again.

- (3) The instructor should be able to create his/her own algorithms to calculate the final grade.
- (4) There should be a workflow library from where instructor should be able to choose an existing workflow without having to create one. Any workflows created by the instructors should be added to the workflow library making it available for use by other instructors.
- (5) Group tools should be added to PLAE to enhance group collaboration.
- (6) Students and instructors should be able to come up with an evaluation scheme collaboratively.
- (7) The system should be integrated with an anti-plagiarism service like www.turnitin.com to check for the authenticity and originality of artifacts.
- (8) Introduction of rights management in a course to allocate different rights to instructors and TAs over a course.

6.3 Conclusion

There is no doubt that it is imperative to have good system design as a foundation for the successful implementation of PLAE. However, it is absolutely necessary that students and instructors are comfortable with the Internet and the approach to learning through online delivery. Online delivery of education does not only compliment face-to-face delivery of education but it can be used as an effective and independent channel of education. Barker (2002) points out that learning online requires a different set of skills to adapt to the new medium. It is important to address the instructors' and the student skills of online delivery (Hughes and Daykin, 2002).

Finally, to implement the PLAE versions described in this thesis, the system could either be built from scratch or several public domain systems could be integrated to meet the requirements. For designing future versions of PLAE, the usability of the system, the value of the current functionalities and feedback of the instructors and students should be evaluated and considered.

APPENDIX A

PRE-DEFINED ELEMENT_TYPES ENTITY

Element_Types entity

Element_type_id	Element_name	Html_element
1	question	textarea
2	question_options	textbox
3	answer	textarea
4	answer_options	radio
5	comments	textarea
6	points	textbox
7	multiple_answer_options	check

APPENDIX B

PLAE ARTIFACT CONFIGURATION EXAMPLE

The PLA process in example is described as follows:

1. Student (problem creator) creates a problem which includes creating two questions based on a Peter Keen article.
2. The questions are reviewed by a student (problem reviewer) who is different from a problem creator. The problems are not graded.
3. A student (problem solver) answers the questions. The problem solver is different from the problem creator and problem reviewer.
4. The solution is evaluated by two students (evaluator #1 and evaluator #2). Evaluator #1 should be the same as the problem creator and evaluator #2 should be a student different from the problem solver and evaluator #1. Both the evaluations are grouped together so that system can calculate the final grade based on the two evaluations and assign it to the problem solver.
5. The instructor decides the final grade for the problem solver if the two evaluations meet the threshold criteria set while configuring the assignment.
6. The problem solver can register a dispute if he/she is not content with his/her final grade.
7. The dispute will be arbitrated by the instructor and he/she will review the final grade of the problem solver.

Note: 'N/A' means that option is not available for that artifact.

'-' means that the option is available but nothing is selected or entered.

'+' means the instructor manually groups the evaluation artifacts.

'' means system automatically groups the evaluation after manual grouping of artifacts*

[illegible]

APPENDIX C

SCREEN AFTER LOGIN FOR INSTRUCTOR/STUDENT

The screenshot shows a web application interface for an instructor or student after login. The interface is displayed within a Microsoft Internet Explorer browser window.

Navigation Menu (Left):

- Discussion Areas**
 - Chapter 1-2
 - Chapter 3-4
- Task Areas**
 - Homework1
 - Workflow #1
 - Workflow #2*
 - Workflow #3
 - OnlineQuiz1
 - Final Exam

Main Content Area:

Hello Glenn McMillan, you have the following pending tasks:

Task Name	Due Date
Homework1	4/8/2004 12:00am
OnlineQuiz1	4/12/2004 12:00am

Assignment Workflow (Bottom):

Assignment: Homework1(Workflow2)

Workflow Steps:

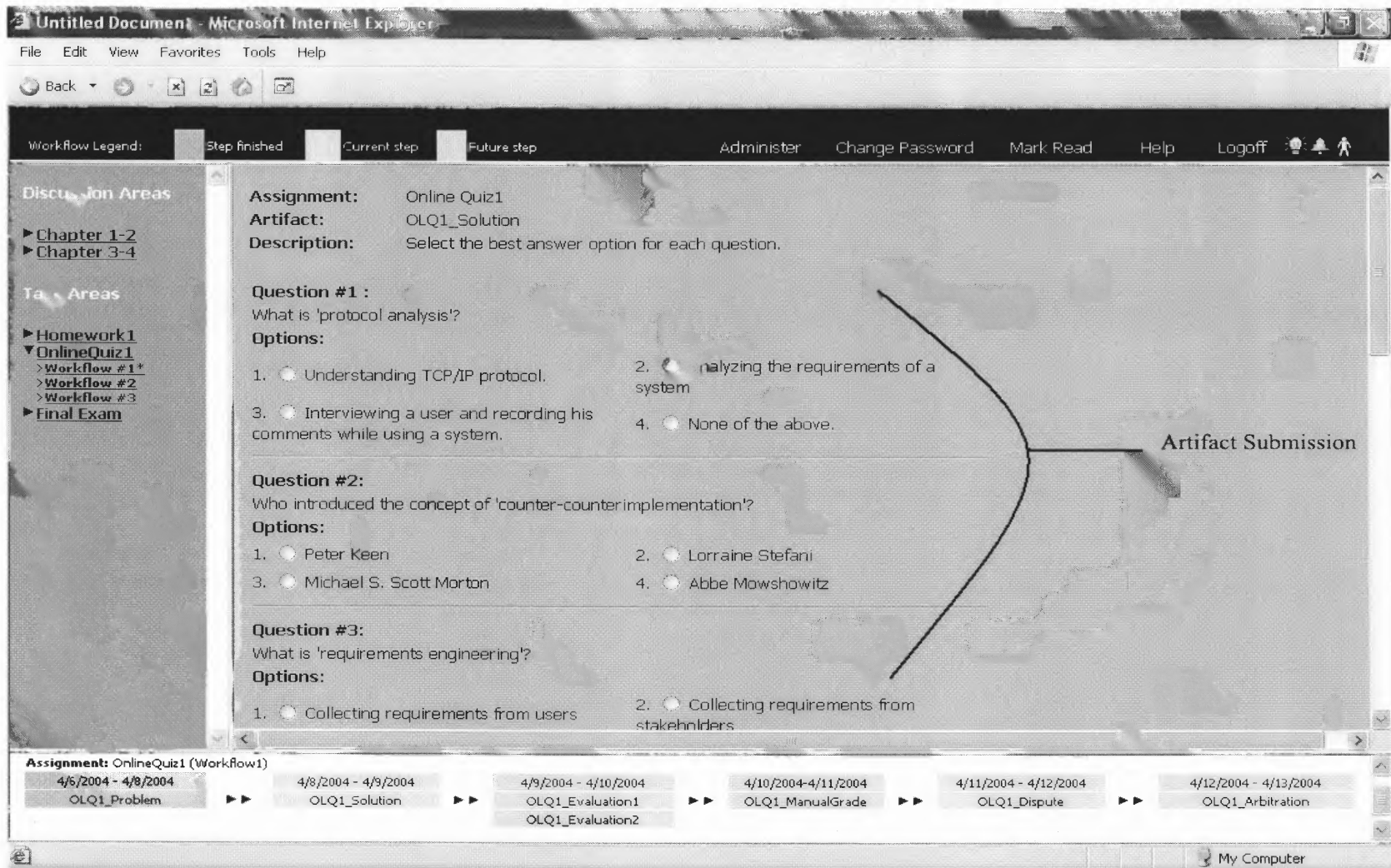
- 4/1/2004 - 4/3/2004: HW1_Problem
- 4/3/2004 - 4/4/2004: HW1_Review
- 4/4/2004 - 4/7/2004: HW1_Solution (Current step)
- 4/7/2004 - 4/8/2004: HW1_Evaluation1, HW1_Evaluation2
- 4/8/2004 - 4/9/2004: HW1_ManualGrade
- 4/9/2004 - 4/11/2004: HW1_Dispute
- 4/11/2004 - 4/13/2004: HW1_Arbitration

Annotations:

- Discussion Areas:** Points to the Discussion Areas section in the navigation menu.
- Pending Tasks:** Points to the pending tasks list.
- Course options:** Points to the navigation menu.
- Artifact:** Points to the HW1_Solution step in the workflow.
- Assignment Workflow:** Points to the entire workflow section.

APPENDIX D

SUBMIT ARTIFACT SCREEN FOR OBJECTIVE QUESTIONS



REFERENCES

- Barker, P. G. (2002). On being an online tutor. *Innovations in Education and Teaching International*, 39(1), 3-13.
- Boud, D. (1990). Assessment and the promotion of academic values. *Studies in higher education*, 15(1).
- Boud, D., Cohen, R., and Sampson, J. (1999). Peer learning and assessment. *Assessment & evaluation in higher education*, 24(4).
- Brindley, C., and Scoffield, S. (1998). Peer assessment in undergraduate programmes. *Teaching in higher education*, 13(1).
- Dancer, W.T., and Dancer, J. (1992). Peer rating in higher education. *Journal of Education for Business*, 67(5).
- Davies, P. (2000). Computerized peer assessment. *Innovations in education and training international*.
- Dochy, F., and Segers, M. (1999). The Use of Self-, Peer and Co-assessment in Higher Education: a review. *Studies in Higher Education*, Oct 99, 24(3)
- Douglas, M. (2001). Reciprocity as a source of bias in multiple peer assessment of group work. *Studies in higher education*, 26(1).
- Entwistle, N. (2000). Promoting Deep Learning through Teaching and Assessment, in Assessment to Promote Deep Learning: Insights from Aahe's 2000 and 1999 Assessment Conferences, Suskie, L. (Ed.), Washington, DC: *American Association for Higher Education*. 9-20.
- Fallows, S., and Chandramohan, B. (2001). Multiple approaches to assessment: reflections on use of tutor, peer and self-assessment. *Teaching in higher education*, 6(2).
- Fox, R. (2001). Constructivism examined. *Oxford review of education*, 27(1).
- Freeman, M., and McKenzie, J. (2002). SPARK, a confidential web-based template for self and peer assessment of student teamwork: benefits of evaluating across different subjects. *British journal of educational technology*, 33(5).
- Fry, S. A. (1990). Implementation and evaluation of peer marking in higher education. *Assessment and evaluation in higher education*, 15(3), 177-189.

- Gatfield, T. (1999). Examining student satisfaction with group projects and peer assessment. *Assessment and evaluation in higher education*, 24(4).
- Goldfinch, J. (1994). Further developments in peer assessment of group projects. *Assessment and evaluation in higher education*, 19(1).
- Goldfinch, J. M., and Raeside, R. (1990). Development of a peer assessment technique for obtaining individual marks on a group project. *Assessment & Evaluation in Higher Education*, 15(3), 210-225.
- Hargreaves, D. J. (1997). Student learning and assessment are inextricably linked. *European Journal of Engineering Education*, 22(4), 401-410.
- Harris, D. and Bell, C. (1994). *Evaluating and Assessing for Learning*. (New York, Kogan Page).
- Higgins, K. M., Harris, N. A., and Kuehn, L. L. (1994). Placing assessment into the hands of young children: a study of student-generated criteria and self-assessment. *Education Assessment*, 2(4), 309-324.
- Hughes, M. and Daykin, N. (2002). Towards Constructivism: Investigating Students' Perceptions and Learning as a Result of Using an Online Environment. *Innovations in Education and Teaching International*, 217-224.
- Kwok, R. and Ma, J. (1999). Use of a group support system for collaborative assessment. *Computers and Education*, 109-125.
- Li, L. (2001). Some Refinements on Peer Assessment of Group Projects. *Assessment & Evaluation in Higher Education*, 26(1).
- Liu, E., Lin, S., and Yuan, S. (2002). Alternatives to Instructor Assessment: A Case Study of Comparing Self and Peer Assessment with Instructor Assessment Under a Networked Innovative Assessment Procedures. *International Journal of Instructional Media*, 29(4).
- Liu, E., Lin, S., Chiu, Chi., and Yuan, S. (2001). Web-Based Peer Review: The Learner as both Adapter and Reviewer. *IEEE Transactions on Education*, 44(3).
- McConnell, D. (1999). Examining a collaborative assessment process in networked lifelong learning. *Journal of Computer Assisted Learning*, 232-243.
- Miller, C. M. L., and Parlett, M. (1974). *Up to the Mark: A Study of the Examination Game* (Guildford, Society for Research in Higher Education).
- Montgomery, B. (1986). An interactionist analysis of small group peer assessment, *Small Group Behavior*, 17, 19-37.

- Orsmand, P., Merry, S., and Reiling, K. (1997). A study in self-assessment: tutor and students' perceptions of performance criteria. *Assessment and Evaluation in Higher Education*, 22(4).
- Perkins, D. (1999). Many faces of constructivism. *Educational Leadership*, 6-11.
- Phillips, D.C. (1995). The good, the bad and the ugly: The many faces of constructivism. *Educational Researcher*, 24(7), 5-12.
- Piaget, J. (1928). *Judgment and reasoning in the child*. (London, Routledge & Kegan Paul).
- Rafiq, Y., and Fullerton, H. (1996). Peer assessment of group projects in civil engineering. *Assessment and Evaluation in Higher Education*, 21, 69-81.
- Sewell, A. (2002). Constructivism and student misconceptions. *Australian Science Teacher's Journal*, 48(4).
- Shen, J., Kung-E C., and Bieber, M. (2004a). Collaborative Examinations for Asynchronous Learning Networks, in preparation.
- Shen, J., Kung-E C., Bieber, M., and Hiltz, R. (2004b). Traditional In-class Examination vs. Collaborative Online Examination in Asynchronous Learning Networks: Field Evaluation Results, *Proceedings of the Tenth Americas Conference on Information Systems*, New York, New York, August 2004.
- Stefani, L. (1998). Assessment in partnership with learners. *Assessment and valuation in higher education*, 23(4)
- Tytler, R. (2002). Student conceptions research and changing views of learning. *Australian Science Teacher's Journal*, 48(3).
- Vygotsky, L. (1978). *Mind in society: the development of higher psychological processes*. (Cambridge, MA, Harvard University Press).
- Wilson, B. and Lowry, M. (2000). *New directions for adult and continuing education*, No.88.
- Wu, D., Bieber, M., Hiltz, R., and Han H.J. (2004). Constructivist Learning with Participatory Exams. *Proceedings of the 37th Hawaii International Conference on System Sciences*, IEEE Press, Washington, D.C., January 2004.