

# Requirements Specification

## Javanet

### *Clients:*

Dr. Darren Lim  
Assistant Professor  
Department of Computer Science  
Siena College

Mrs. Pauline White  
Visiting Instructor  
Department of Computer Science  
Siena College

## Phoenix Tech

*Prepared By:*  
**Jessica Edmonds**  
**Marc Iuliano**  
**Anthony Torres**  
**Erica Weisgerber**

**October 22, 2008**

# Table of Contents:

## Javanet

1.1	Product Overview and Summary .....	2
1.2	Development, Operating, and Maintenance Environments .....	2
1.3	Data Flow Diagrams .....	3
1.3.1	Data Flow Diagram Key .....	3
1.3.2	Context Diagram .....	4
1.3.3	Level 0 Diagram .....	5
1.3.4	Level 1 Diagram: Login .....	6
1.3.5	Level 1 Diagram: Manage Accounts .....	7
1.3.6	Level 1 Diagram: Manage Question/Question Sets .....	8
1.3.7	Level 1 Diagram: Answer Problem .....	9
1.3.8	Level 1 Diagram: View Grade Book .....	10
1.4	UML Use Case Diagram .....	11
1.4.1	UML Use Case Diagram Key .....	11
1.4.2	UML Use Case Diagram (for Javanet) .....	12
1.5	Use Case Narratives .....	13
1.6	Functional Requirements Inventory .....	16
1.7	Performance Requirements .....	19
1.8	Exception Handling .....	19
1.9	Early Subsets and Implementation Priorities .....	20
1.10	Foreseeable Modifications and Enhancements .....	20
1.11	Acceptance Criteria .....	20
1.12	Testing Requirements .....	20
1.13	Cross Reference Index .....	20
1.14	Sources of Information .....	21

### *Appendix*

2.1	Timeline (Gantt Chart) .....	22
2.2	Glossary of Terms .....	23

## **1.1 Product Overview and Summary**

Our clients, Dr. Darren Lim and Mrs. Pauline White, have requested a web based program that would help instructors to track their students' progress and provide students with a dynamic and interactive way of learning. Here at Phoenix Tech we believe we can offer our clients a solution that will allow instructors to easily upload, assign and grade Java programming problems. Our software will allow students to practice Java programming problems and take Java based quizzes and tests provided by their instructor. Ultimately, student progress will be able to be tracked both immediately and gradually, by both the students themselves and the instructors.

## **1.2 Development, Operating, and Maintenance Environments**

Javanet will be developed at Siena College in the Software Engineering laboratory. The Phoenix Tech team will be using a Dell Dimension 4550 running Microsoft Windows XP Service Pack 2, and an Apple iMac running Macintosh Operating System X 10.4.10 to do the primary development.

Javanet will be stored on, throughout development and production, a server consisting of an 8 core Intel Xenon CPU E5430 2.66GHz processor, 8 GB RAM, and roughly 94 GB of disk space available for use. The server is running a 64bit CentOS Linux release 5.2 operating system. It is running an Apache 2.2.9 web server, PHP 5.2.6, and MySQL 5.0.45, and Java Compiler 1.6.0\_10-rc.

Javanet will be accessible via three of the main internet browsers; Internet Explorer 7.0 and 8.0, the latest version of Firefox 2 and 3, and Safari 2.\* or better.

The users of Javanet will be able to access the application from any computer with any of the three internet browsers previously listed and will have full usage of the application.

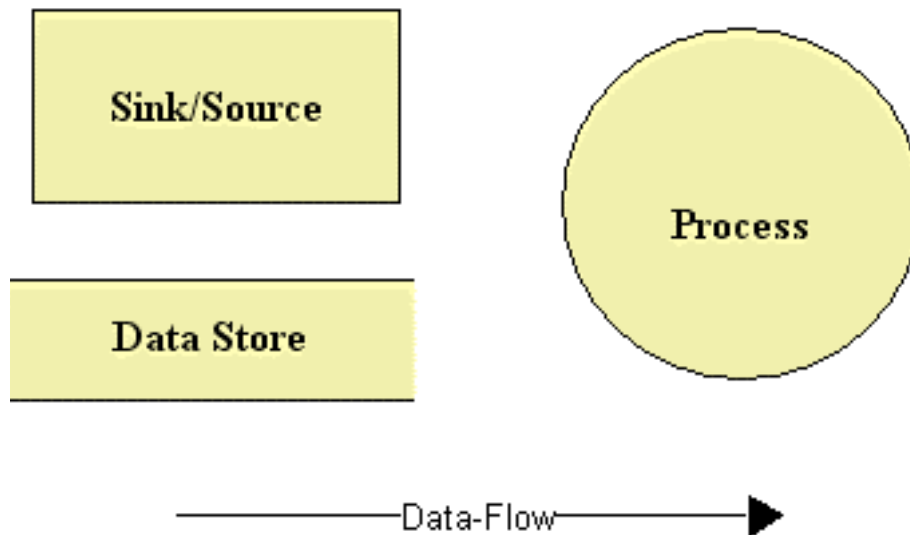
Maintenance of the Javanet application will be discussed at a future date.

## 1.3 Data Flow Diagrams

A Data Flow Diagram (DFD) is a diagram used by Software Engineers to show a graphical representation of the flow of data within a system. Data Flow Diagrams usually contain representations of users, data, processes, and data stores within a system. Data Flow Diagrams serve as a method for the Software Engineer to present information about data flow within a system that is meaningful for not only his/herself, the Software Engineer, but also someone who has not been trained in software engineering practices.

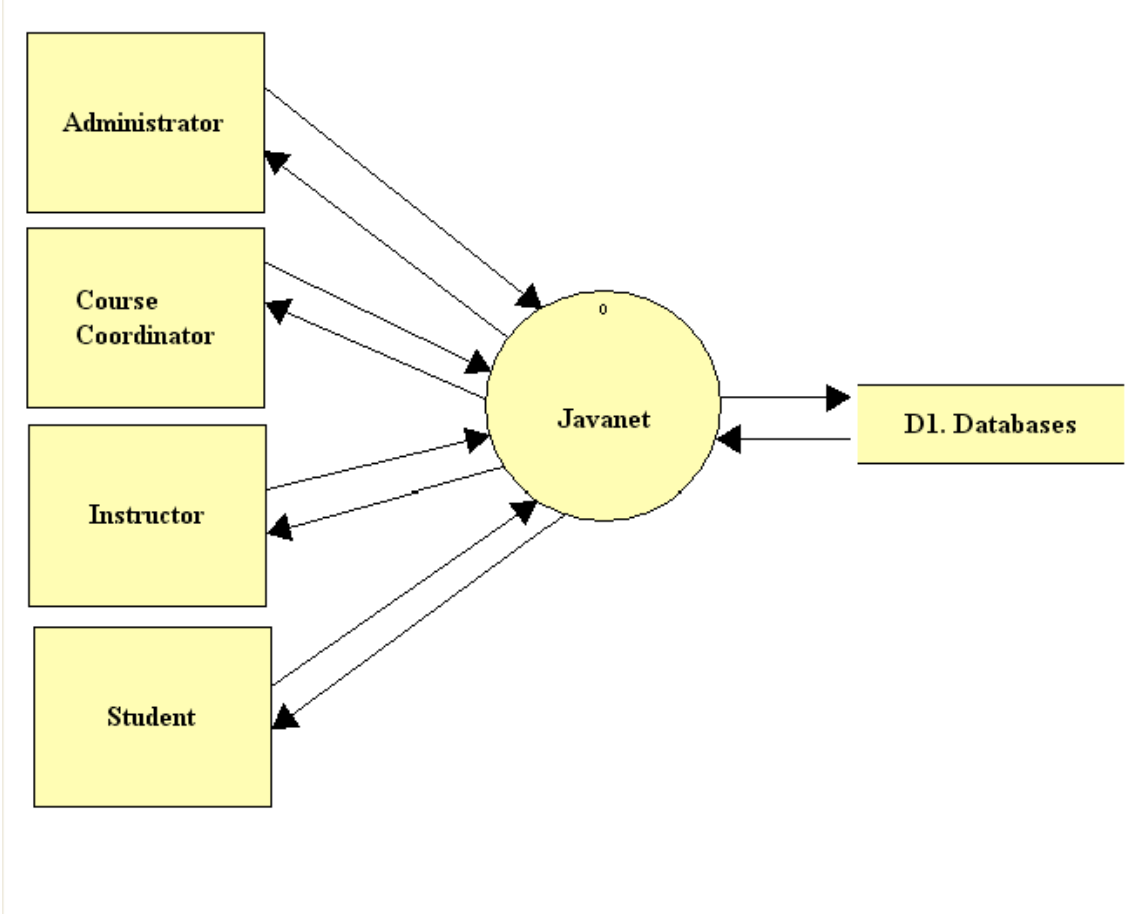
The Data Flow Diagrams presented in this document were developed by the Phoenix Tech team for the Javanet application. The Data Flow Diagrams for the Javanet application will follow the Yourdon style.

### 1.3.1 Data Flow Diagram Key

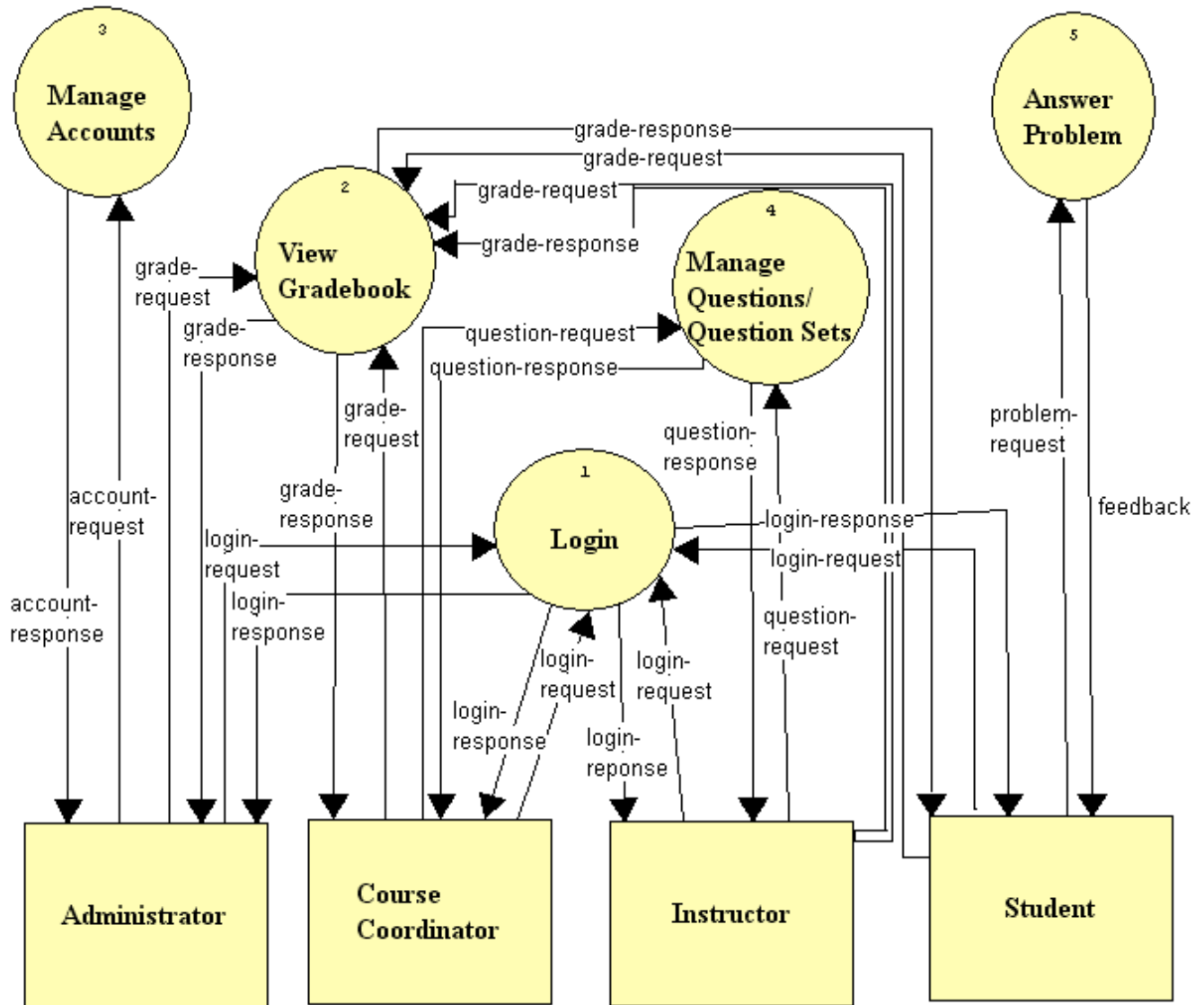


There are four major symbols used in the Data Flow Diagrams. Closed rectangles represent a source or sink. A source or sink is an outside user accessing the system. Circles represent the processes, or actions, that take place within the system. Open-sided rectangles represent data stores. A data store is a structure used to store information. In the case of Javanet, data stores are two databases; one storing user information and one storing question information. Finally, an arrow represents the transfer of data between sources or sinks, processes, and data stores.

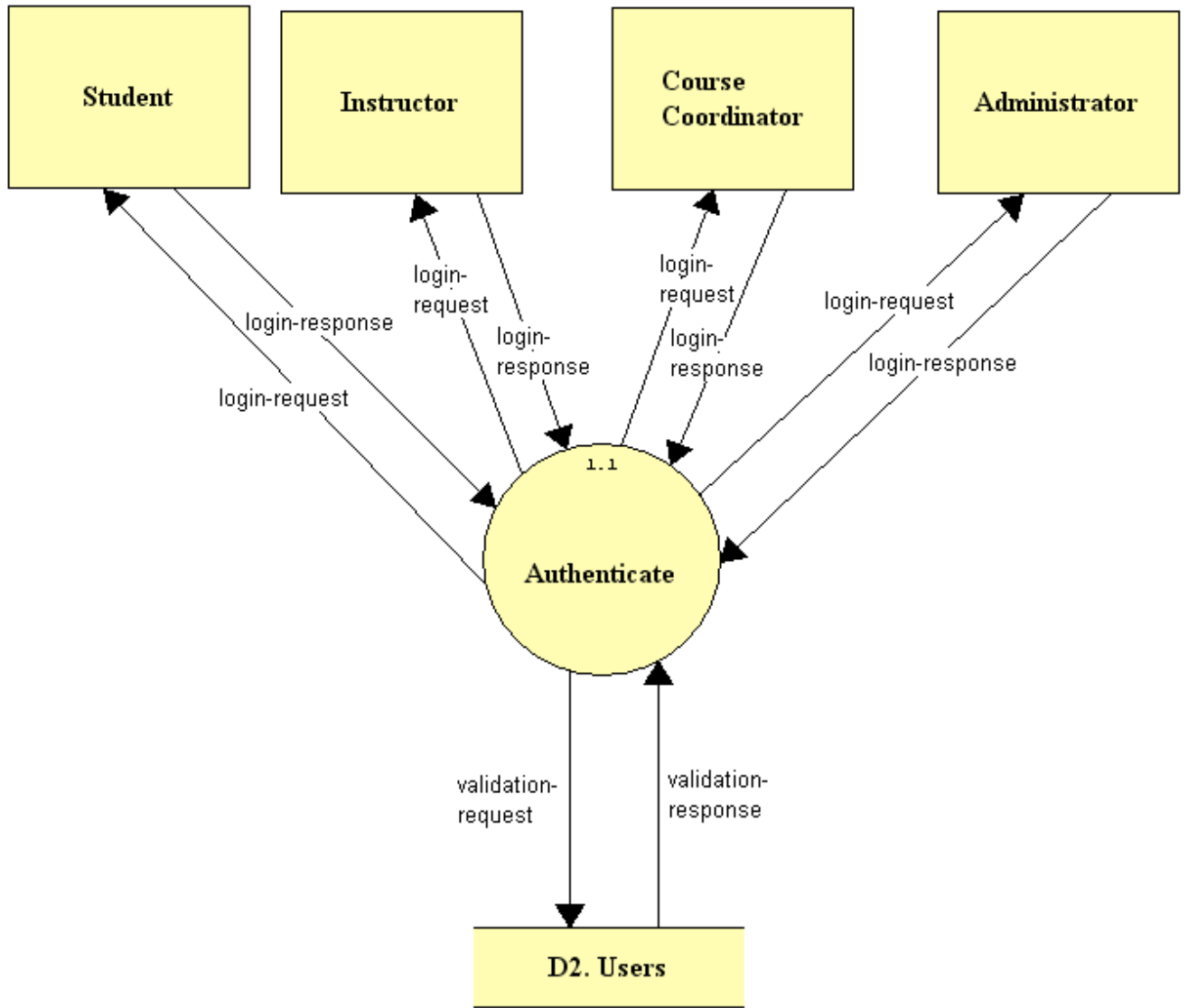
### 1.3.2 Context Diagram



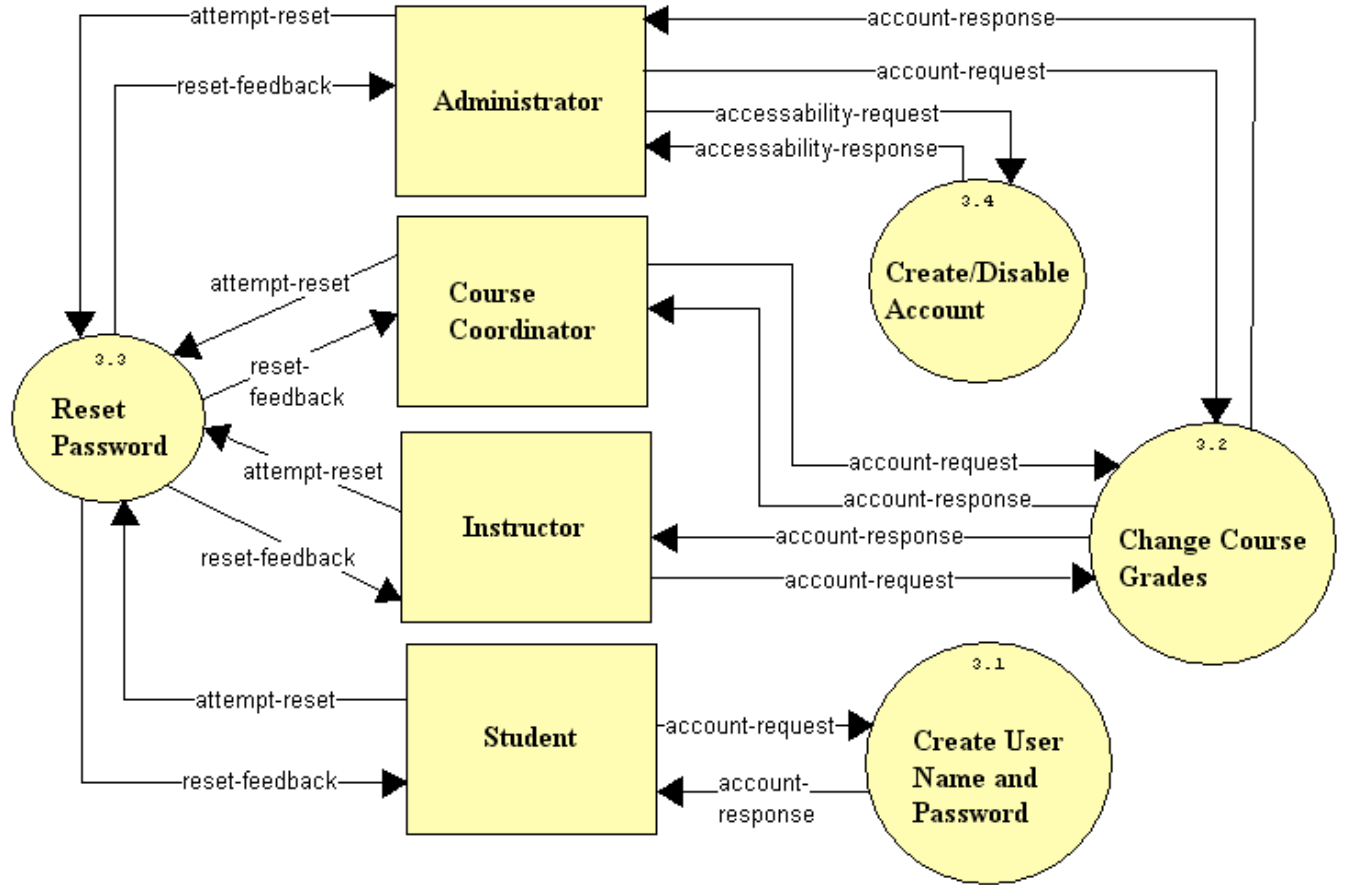
### 1.3.3 Level 0 Diagram



### 1.3.4 Level 1 Diagram: Login

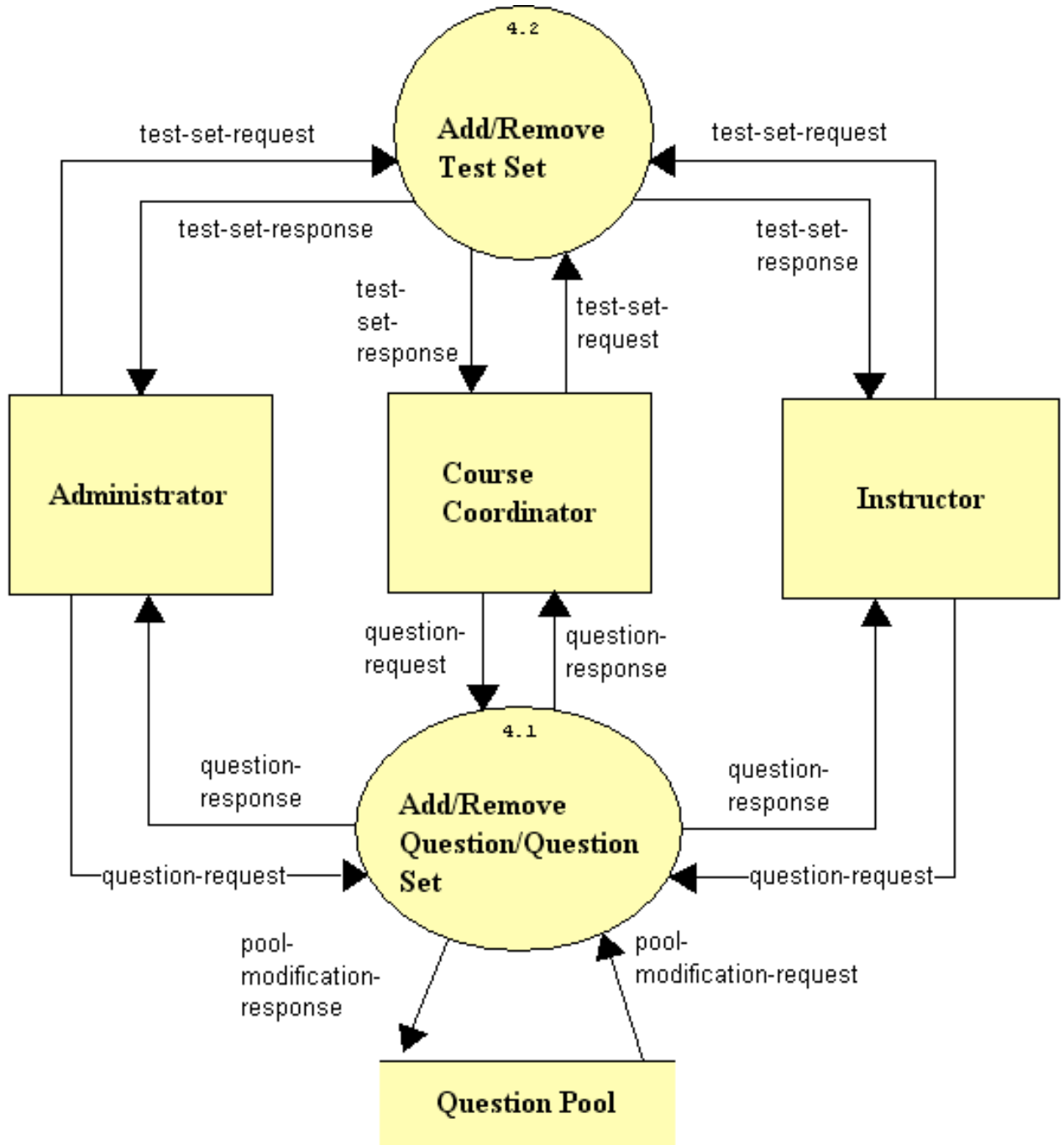


### 1.3.5 Level 1 Diagram: Manage Accounts

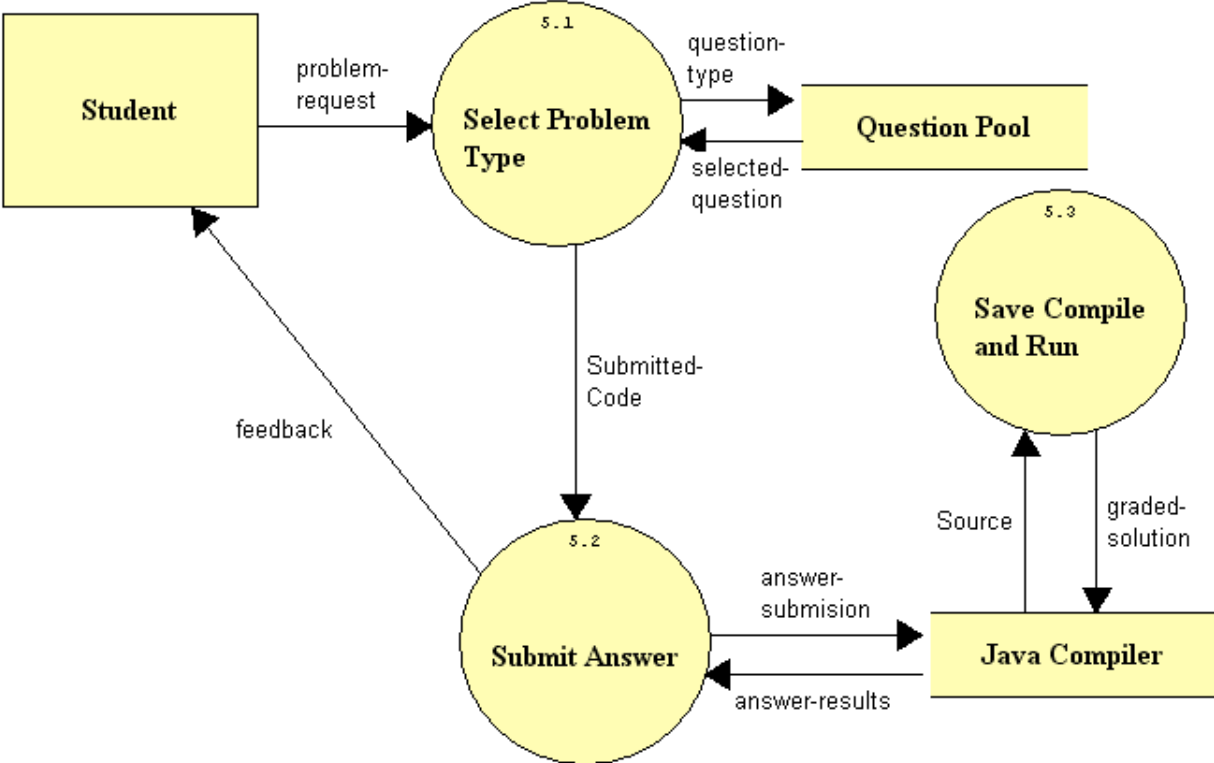




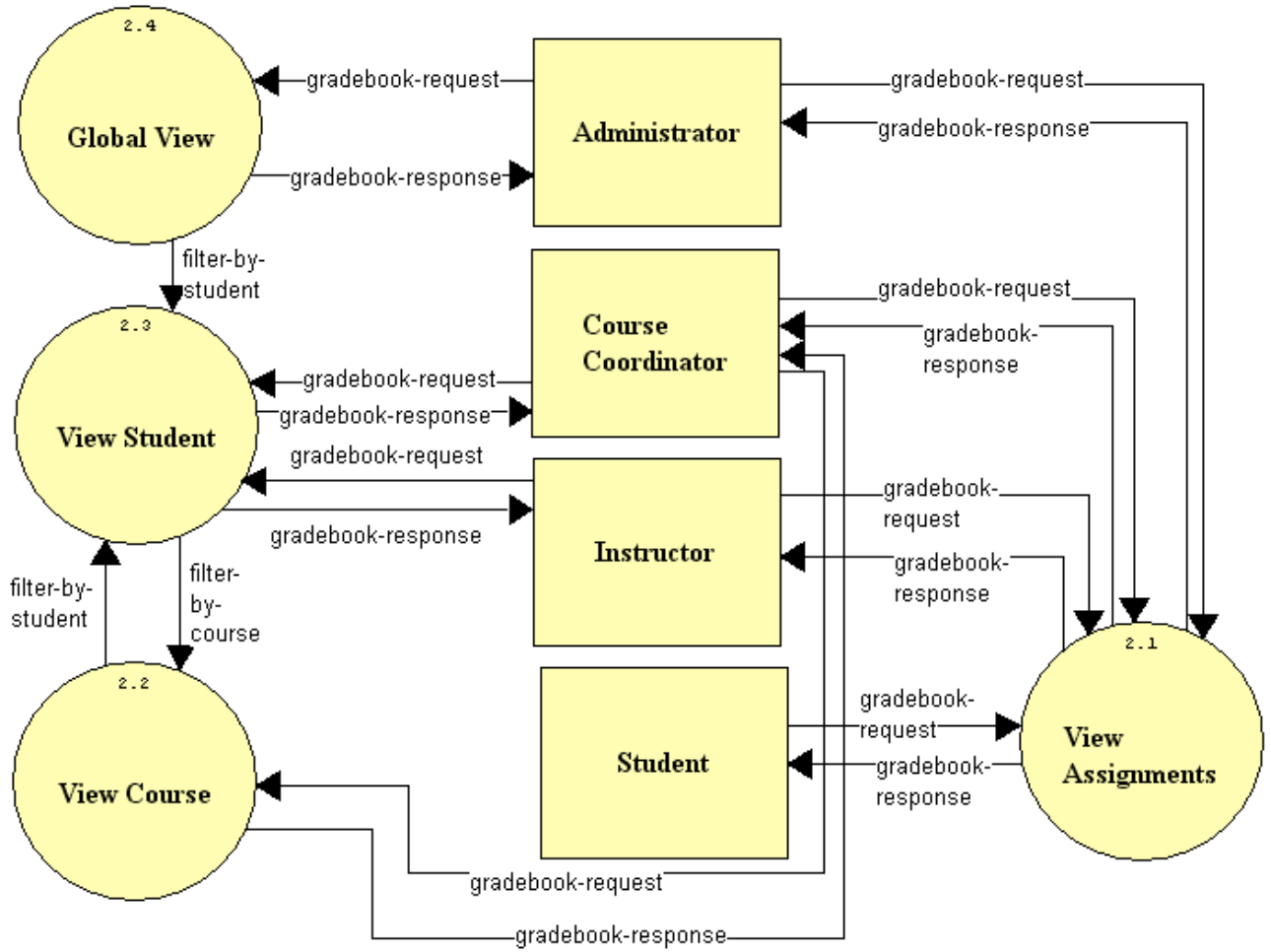
### 1.3.6 Level 1 Diagram: Manage Question/Question Sets



### 1.3.7 Level 1 Diagram: Answer Problem



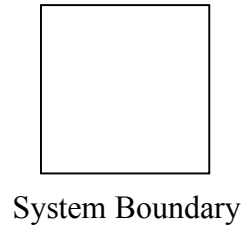
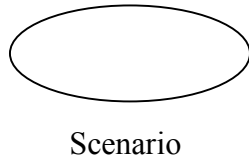
### 1.3.8 Level 1 Diagram: View Grade Book



## 1.4 UML Use Case Diagram

A UML Use Case Diagram is another diagram used by Software Engineers to show a graphical representation of the functionality provided by a system. Use Case Diagrams contain representations of actors and scenarios. An actor can be a human or non-human entity. A scenario is a specific use case. The ultimate goal of a Use Case Diagram is to show what functions are performed by the system and which actors use these functions.

### 1.4.1 UML Use Case Diagram Key



### 1.4.1 UML Use Case Diagram (for Javanet)



## 1.5 Use Case Narratives

### Administrator

The Administrator user will have the ability to login to and log out of the application through a web interface with a username and password provided by the application developers. Once logged into the application, the Administrator will have the ability to create Instructor and Course Coordinator accounts, and confirm Student accounts. The Administrator will also have the ability to deactivate user accounts, and manage and reset passwords. The Administrators can upload Instructor questions and question sets, upload Instructor test sets, and create questions and question sets to go into the Universal Pool. The Administrator will be able to submit statistical data to common content for all instructors to see. Administrators will be able to switch into any Instructor, Student, or Course Coordinator view mode. The Administrator will have the ability to view the grade book sorted by assignment, specific student, or course. Administrators will be able to view student progress, and assignment statistics across all courses. If necessary, the Administrator will also be able to change grades. The Administrator will also be able to archive student information from the database.

### Course Coordinator

The Course Coordinator will be able to log into and log out of the system. The Course Coordinator will be the coordinator of all sections of one specific course. The Course Coordinator will have the ability to submit statistical data to common content of the specific course that they are the coordinator for. This statistical data will be seen by all Instructors. The Course Coordinator can switch into instructor or student view modes. The Course Coordinator will also be able to create questions or question sets to go into the Course pool, import questions or question sets sent by Instructors to the Course Pool to be shared with other Instructors of the course, and import questions or question sets sent by Instructors to the Universal Pool to be available to all other Instructors. If necessary, the Course Coordinator will have the ability to change the grades of the Students of the course that they coordinate. The Course Coordinator will be able to view the grade book sorted by assignment, or specific Student. Across the course they are coordinating, the Course Coordinator may also view Student progress.

## **Instructor**

The Instructor user will have the ability to login to and log out of the application through a web interface with a username and password. The Instructor accounts will be created by the Administrator user. The Instructor users will be given an initial password. The Instructor will have the ability to change their password at anytime. Upon login the Instructor will be able to select what course section page they would like to view. As Student users identify themselves as being enrolled in a specific course, they will appear on the appropriate course page for the corresponding Instructor. The Instructor will have the ability to create new folders/tabs of practice questions, homework assignments, quizzes, and tests. When creating a question, the Instructor will have the following options: including hints for the Students, viewing Student source code, selecting a category for the question, selecting a difficulty level for a question, and selecting the number of compilations allowed for a question or question set. Instructors will have the ability to compile their own private pool of questions. The Instructor may choose to create and publish question sets from their private pool for use by the students enrolled in their course. Instructors may send the Course Coordinator their questions or question sets from their private pool to be published to the Course Pool to share with other Instructors of the course they are teaching. An Instructor may also send the Course Coordinator their questions or question sets to be published to the Universal Pool to be available to all other Instructors. Instructors may also import questions and question sets to their private pool from both the Course and Universal pools. The Instructor will also have the option of how they would like to review results: they can select a Student and scroll through that Student's assignments, or they can select an assignment and scroll through all of the Students. The Instructor will be able to create a test set for the Student solution to be tested against. The Instructor will also have the ability to override a Student's grade when they think the automatically generated grade is inappropriate. After an exam is administered and completed, the Instructor will have the ability to see test statistics across all Instructor sections for the corresponding course. The Instructor will also be able to give permission to a particular Student to view previous assignments, quizzes, and tests. Additionally, the Instructor will be able to click to a Student view in order to best assist Students when they are in need of help.

## **Student**

The Student user will have the ability to login to and log out of the application through a web interface with a username and password, once they have created an account. When creating an account the Student will be required to create a username and password. The Student will be able to change their password at any time. If the Student forgets their password, they will be able to click a link to an identity validation page to reset the password. The Student will also be able to edit various aspects of their account information at any time. The Student will then be required to enter their name and select what course they are enrolled in. Upon login the Student will be able to select what mode they would like to work in: practice, homework, quiz, or test. In practice mode, the Student will be able to select a category of problems to practice with. In homework mode, the Student will be able to complete the current homework assignment posted by their instructor. In both quiz and test mode, the Student will be able to complete the current quiz or test posted by their instructor. The Student will be required to compile and execute for submission in one step. In all cases, the Student will get instant feedback about their performance, on a category or set of questions, in the form of both a percentage of questions correct and the actual number of questions correct. Students will have the ability to look back at and practice with previous homework assignments, quizzes, and tests with the permission of the Instructor.



## 1.6 Functional Requirements Inventory

The following list outlines the required functionality to be included in the final solution.

Javanet will be a web-based application viewable on the four major browsers. These browsers are Internet Explorer 7.0 and 8.0, the latest version of Firefox 2 and 3, and Safari 2.\* or better.

- Index page displays a login screen common to all users.
- All other pages display a “logout” option for all users.

The requirements are listed according to user case, as follows:

### **Administrative User:**

- Will be able to log into system.
- Will be able to confirm Student accounts.
- Will be able to create Instructor accounts.
- Will be able to create Course Coordinator accounts.
- Will be able to manage user accounts, including passwords.
- Will be able to reset passwords of all users.
- Will be able to create questions and question sets to go into the Universal pool.
- Will be able to submit statistical data to common content for all instructors to see.
- Will be able to switch into Instructor view mode.
- Will be able to switch into Student view mode.
- Will be able to switch into Course Coordinator view mode.
- Will be able to view grade book sorted by assignment.
- Will be able to view grade book sorted by specific student.
- Will be able to view grade book sorted by course.
- Will be able to view student progress across all courses.
- Will be able to view assignment statistics across all courses.
- Will be able to change grades, if needed.
- Will be able to archive student information from the database.
- Will be able to deactivate accounts.
- Will be able to log out of the system.

### **Course Coordinator User:**

- Will be able to log into the system
- Will be able to submit statistical data to common content for all instructors of the specific course that they are the coordinator for to see
- Will be able to switch into instructor view mode.
- Will be able to switch into student view mode.
- Will be able to create questions or question sets to go into the Course pool.
- Will be able to import questions or question sets sent by Instructors to the Course Pool to be shared with other Instructors of the course.
- Will be able to import questions or question sets sent by Instructors to the Universal Pool to be available to all other Instructors.
- Will be able to change grades for students of the course that they are the coordinator for, if needed.
- Will be able to view grade book sorted by assignment.
- Will be able to view grade book sorted by specific student.
- Will be able to view student progress across the course that they are the coordinator for.
- Will be able to log out of the system.

## **Instructor User:**

- Will be able to login to the application through a web interface with a username and password.
- Will be able to change their password at any time.
- Will be able to select what course section page to view on login.
- Will be able to create new folders/tabs for practice questions.
- Will be able to create new folders/tabs for homework assignments.
- Will be able to create new folders/tabs for quizzes and tests.
- Will have the following options when creating a question:
  - Including hints for the Students.
  - Selecting a category for the question.
  - Selecting a difficulty level for a question.
- Will be able to amass their own private pool of questions.
- Will be able to create and publish question sets from their private pool for use by the students enrolled in their course.
- Will be able to submit question sets to the Course Coordinator of a specific course, to be published from their private pool to the Course Pool to share with other Instructors of the specific course.
- Will be able to submit question sets to the Course Coordinator of a specific course, to be published from their private pool to the Universal Pool to share with other all other Instructors.
- Will be able to import questions and question sets to their private pool from the Course pool.
- Will be able to import questions and question sets to their private pool from the Universal pool.
- Will be able to choose how they would like to review results:
  - Select a Student and scroll through that Student's assignments.
  - Select an assignment and scroll through all of the Students.
- Will be able to create a test set for the Student solution to be tested against.
- Will be able to override a Student's grade when the automatically generated grade is inappropriate.
- Will be able to see statistics across all Instructor sections for the corresponding course after an exam is administered and completed.
- Will be able to see statistics across all Instructor sections for the corresponding course after a homework assignment is administered and completed.
- Will be able to give permission to a particular Student to view previous assignments, quizzes, and tests.
- Will be able to click to a Student view in order to best assist Students when they are in need of help.
- Will be able to log out of the system.

## **Student:**

- Will be able to self enroll by creating a username and password.
- Will receive an email confirmation from the Administrator.
- Will be able to log in with the confirmed username and password through a web interface:
  - An incorrect log in will display an appropriate error message.
  - A link to an identity validation page will be provided if password is forgotten.
- Will be able to view and edit various account information.
- Will be able to enroll into their appropriate course.
- Will be able to select a “practice” tab.
- Will be able to select a “homework” tab.
- Will be able to select a “quiz” tab.
- Will be able to select a “test” tab.
- Will be able to compile and submit solutions to problems.
- Will be able to view their grades.
- Will be able to log out of the system.

## **1.7 Performance Requirements**

Javanet will be a web based user interface that will allow for cross platform compatibility, primarily for use on Windows XP or Vista, Mac OSX, or Linux OS. However assuming any arbitrary device has a web browser with internet access and full JavaScript support, these devices will also be able to interact completely with Javanet

Javanet will be designed to be viewed in a 1024x768 resolution. It will also be tested on Internet Explorer 7.0 and 8.0, the latest version of Firefox 2 and 3, and Safari 2.\* or better.

## **1.8 Exception Handling**

In the process of utilizing Javanet, unanticipated errors are always a possibility even if users comply with the system. Javanet will therefore be prepared to handle the following exceptions:

- In the case of an incorrect log-in, with either the user name or password being entered, the user will be asked to re-enter his or her user name and password, with no lock out limit.
- If a user has forgotten their user name or password, a link will be provided so that the user may receive an email containing their user name and password.
- To ensure no miscommunication when submitting an answer, students will receive confirmation feedback when their submission is successful.

## **1.9 Early Subsets and Implementation Priorities**

The critical components of Javanet are:

- The ability of Administrators to manage user accounts
- The ability of Instructors to upload, post and publish questions and question sets
- The ability of the Instructors and Students to choose between practice, test/quiz, and homework modes
- The ability of Students to submit assignments
- The ability of Javanet to automatically grade students' code
- The ability of Instructors to view their students' grades

## **1.10 Foreseeable Modifications and Enhancements**

Possible modifications to the software may include several of the following:

- Provide more sophisticated statistical analysis of data.
- Provide the ability to combine different sections, classes, and individual students into one set of result data.
- Provide the ability to compare sections, classes and individual student data.

## **1.11 Acceptance Criteria**

Javanet will meet all of the Functional Requirements listed in the Functional Requirements Inventory section of this document. Different levels of testing will be administered throughout development. Testing will take place after the completion of our Detailed Design, which will include a full test plan.

## **1.12 Testing Requirements**

Testing will attempt to ensure that Javanet meets all functional requirements. Non-functional requirements are difficult to measure and will be evaluated by the degree of the clients' and developers' satisfaction. The functional requirements will be broken down into functional modules, called units. These units and their features will be tested for accurate and inaccurate input and output. A preliminary test plan will be provided in the Preliminary Design phase, and a complete test plan will be available in the Detailed Design phase. Test results will be presented in the Acceptance Test stage of our project.

## **1.13 Cross Reference Index**

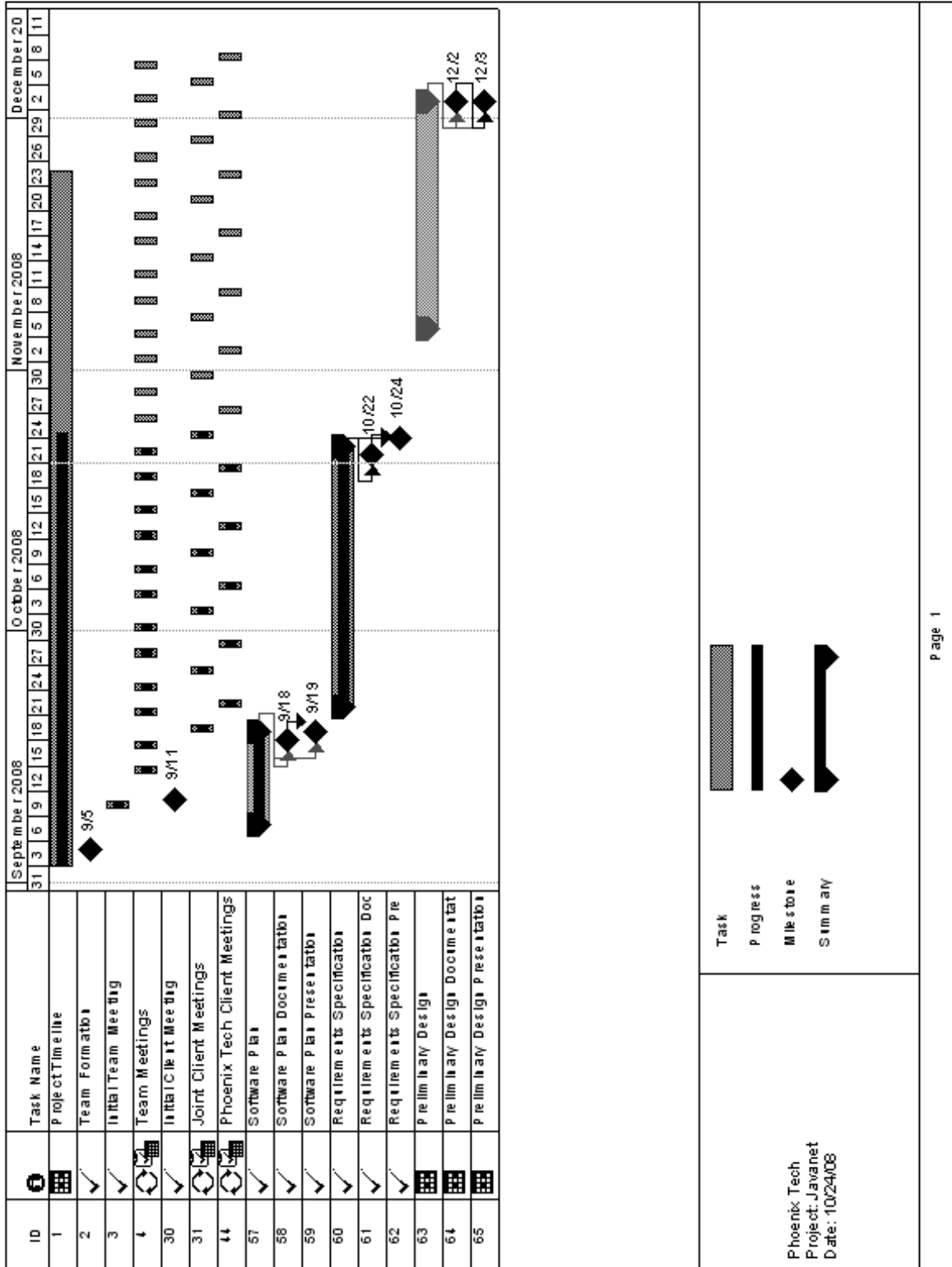
A Cross Reference Index will be available in the next phase of development of our system, the Preliminary Design. A finalized set of Data Flow Diagrams are essential for the Cross Reference Index and will have been developed during the Preliminary Design phase.

## **1.14 Sources of Information**

The information presented in this Requirements Specification document was obtained through meetings with our clients, Dr. Darren Lim and Mrs. Pauline White. Information was also acquired from Dr. Tim Lederman's Software Engineering lectures, Mr. Ken Swarner and various Internet resources. Additional information was collected from Requirements Specifications documents created in previous years by students who were enrolled in the Software Engineering course.

# Appendix

## 2.1 Enlarged Timeline (Gantt Chart)



## 2.2 Glossary of Terms

*All glossary definitions provided by Wikipedia.org.*

### **API**

An application programming interface (API) is a set of functions, procedures or classes that an operating system, library or service provides to support requests made by computer programs.

### **Classic Waterfall Model**

The waterfall model is a sequential software development model (a process for the creation of software) in which development is seen as flowing steadily downwards (like a waterfall) through the phases of requirements analysis, design, implementation, testing (validation), integration, and maintenance.

### **Course Pool**

Collection of question sets that an Instructor of the Pool's specific course has published to the pool. These published questions are then given permission to join the pool of existing question sets in the pool by the Course Pool Administrator.

### **CSS**

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document, including SVG and XUL.

### **Gantt Chart**

A Gantt chart is a type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.

### **GUI**

A graphical user interface (GUI) is a type of user interface which allows people to interact with electronic devices like computers, hand-held devices (MP3 Players, Portable Media Players, Gaming devices), household appliances and office equipment.

### **Hardware**

Hardware is a general term that refers to the physical artifacts of a technology. It may also mean the physical components of a computer system,

### **Java**



Java refers to a number of computer software products and specifications from Sun Microsystems that together provide a system for developing application software and deploying it in a cross-platform environment.

### **Linear Sequential Model**

The Linear Sequential Model is a software process model that involves a systematic progression through analysis, design, coding, testing and maintenance phases. It is also referred to as the "waterfall model".

**Mac OS X version 10.4 “Tiger”** was the fifth major release of Mac OS X, Apple’s desktop and server operating system for Macintosh computers. Tiger was released as the successor to Mac OS X v10.3 “Panther”

### **MySQL**

MySQL is a relational database management system (RDBMS) which has more than 11 million installations. The program runs as a server providing multi-user access to a number of databases.

### **Operating System**

An operating system (commonly abbreviated *OS* and *O/S*) is the software component of a computer system that is responsible for the management and coordination of activities and the sharing of the resources of the computer.

### **PHP**

PHP is a computer scripting language. Originally designed for producing dynamic web pages, it has evolved to include a command line interface capability and can be used in standalone graphical applications.

### **Private Instructor Pool**

A specific Instructor’s own personal collection of questions and question sets they have created/uploaded to their private pool that Students and other Instructors do not have access to.

### **Software**

Software is a general term used to describe a collection of computer programs, procedures and documentation that perform some tasks on a computer system.

### **Source Code**

In computer science, source code (commonly just source or code) is any sequence of statements or declarations written in some human-readable computer programming language. Source code is written in a programming language, which is usually a simplified form of the English language to reduce ambiguity. Source code allows the programmer to communicate with the computer using a reserved number of instructions

## **Universal Pool**

A collection of question sets that may have been published by any Instructor of any course. The question sets are added to the pool only when the Universal Pool Administrator has given permission.

## **Workstation**

A workstation, or *engineering workstation*, is a high-end microcomputer designed for technical or scientific applications. Workstations are intended primarily to be used by one person at a time, although they are commonly connected to a local area network and run multi-user operating systems.

## **Windows Vista**

Windows Vista is a line of operating systems developed by Microsoft for use on personal computers, including home and business desktops, laptops, Tablet PCs, and media center PCs.

## **XHTML**

The *Extensible Hypertext Markup Language*, or XHTML, is a markup language that has the same depth of expression as HTML, but also conforms to XML syntax.