Detailed Design

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

March 9, 2009

Detailed Design
Table of Contents:
Javanet

1.1 Product Overview and Summary

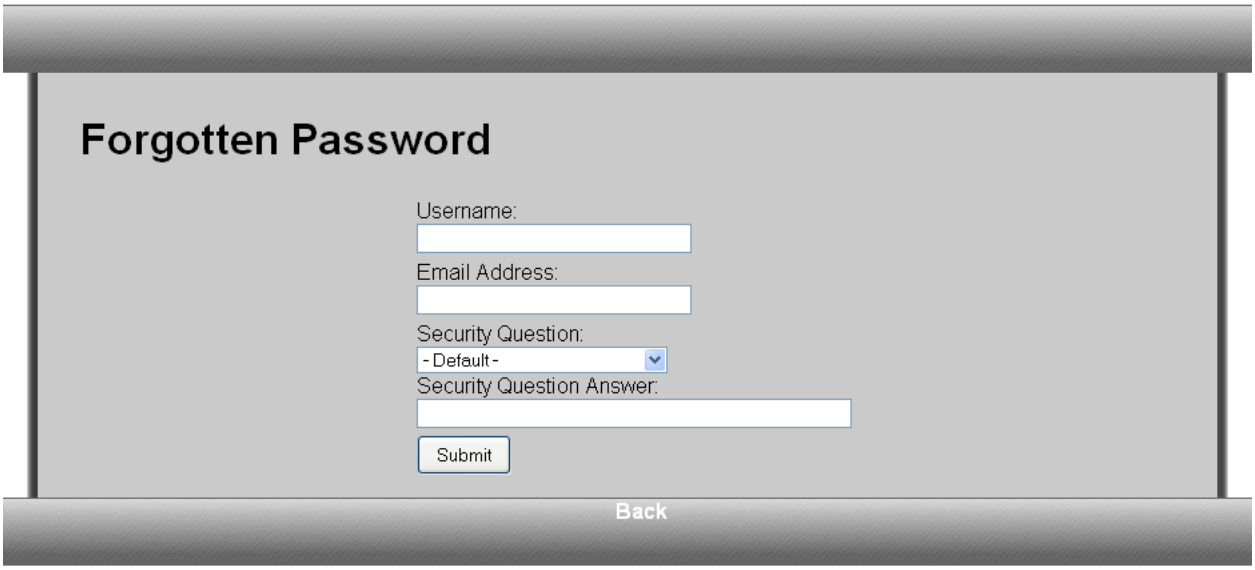
Our clients, Dr. Darren Lim and Mrs. Pauline White, have requested a web based application that will allow students enrolled in a Java based course to practice Java programming problems and take Java based quizzes and tests provided by their instructor. Our software program, Javanet, will also help instructors 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 an efficient solution. A solution that will allow instructors to easily upload, assign and grade Java programming problems. Additionally, student progress will be able to be tracked both immediately and gradually, by both the students themselves and the instructors.

1.2 User Displays and Command Summary

1.2.1 Login Page

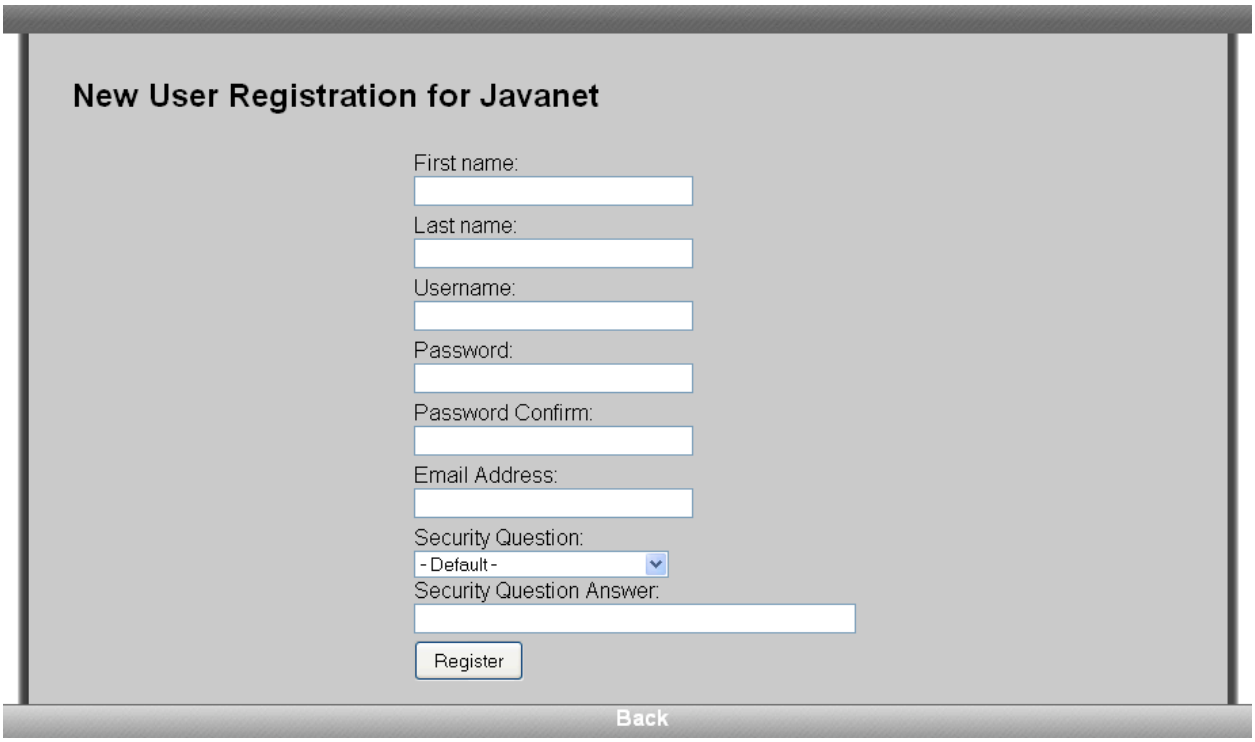
The login page is the first screen the user will see when attempting to login into Javanet. After entering a valid username and password combination, the user will be taken to the appropriate Welcome Page.

1.2.2 Reset Password Page



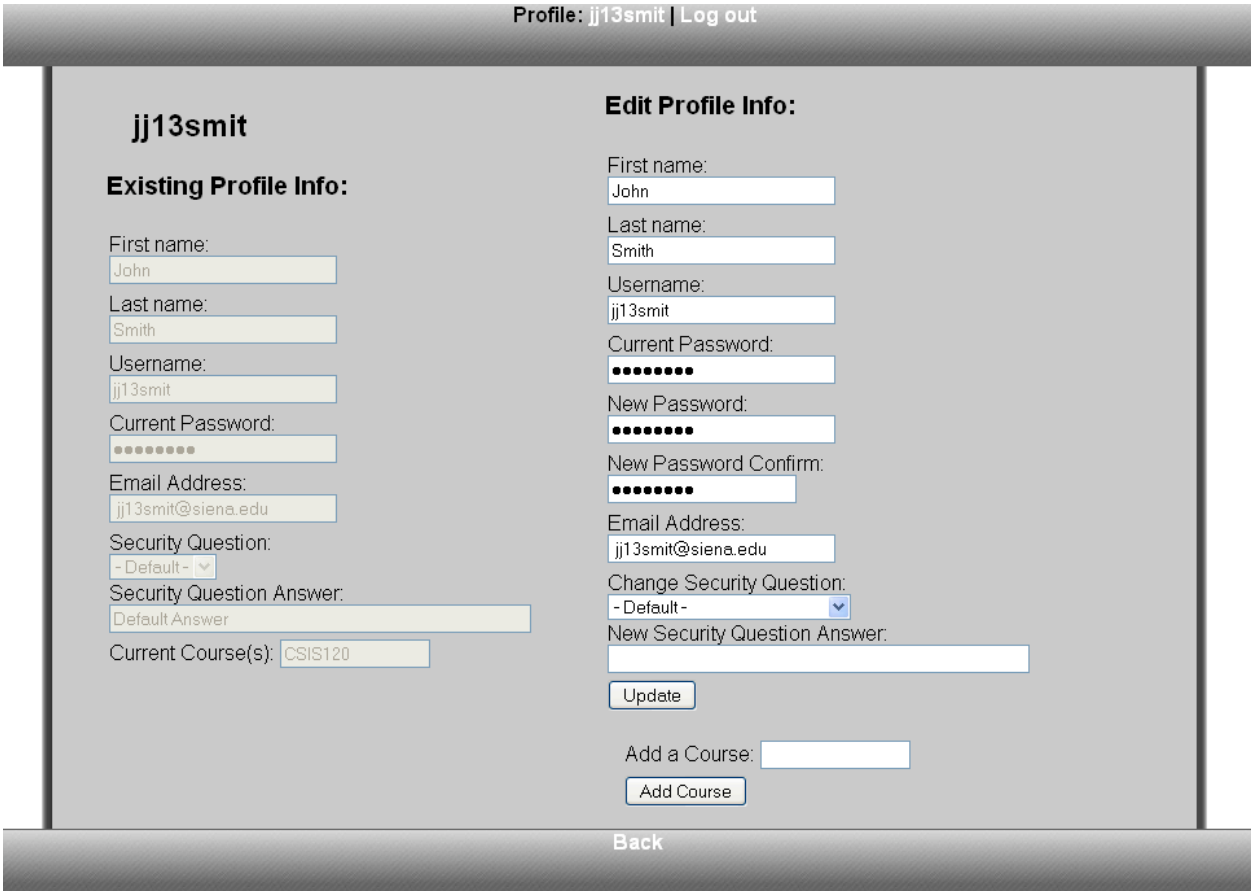
A user will be shown this page if the Forgot Password link is clicked from the Login Page. After entering a user name, email address, and selecting and answering a security question and answer, the user will be emailed a new password.

1.2.3 Student - Registration Page



Students are required to self-enroll. The Student Registration Page will be shown after a Student clicks the Register link from the Login Page. The Student will be required to enter a first name, last name, user name, password, confirmed password, email address, and select a security question and enter an answer.

1.2.4 Student - Profile Page

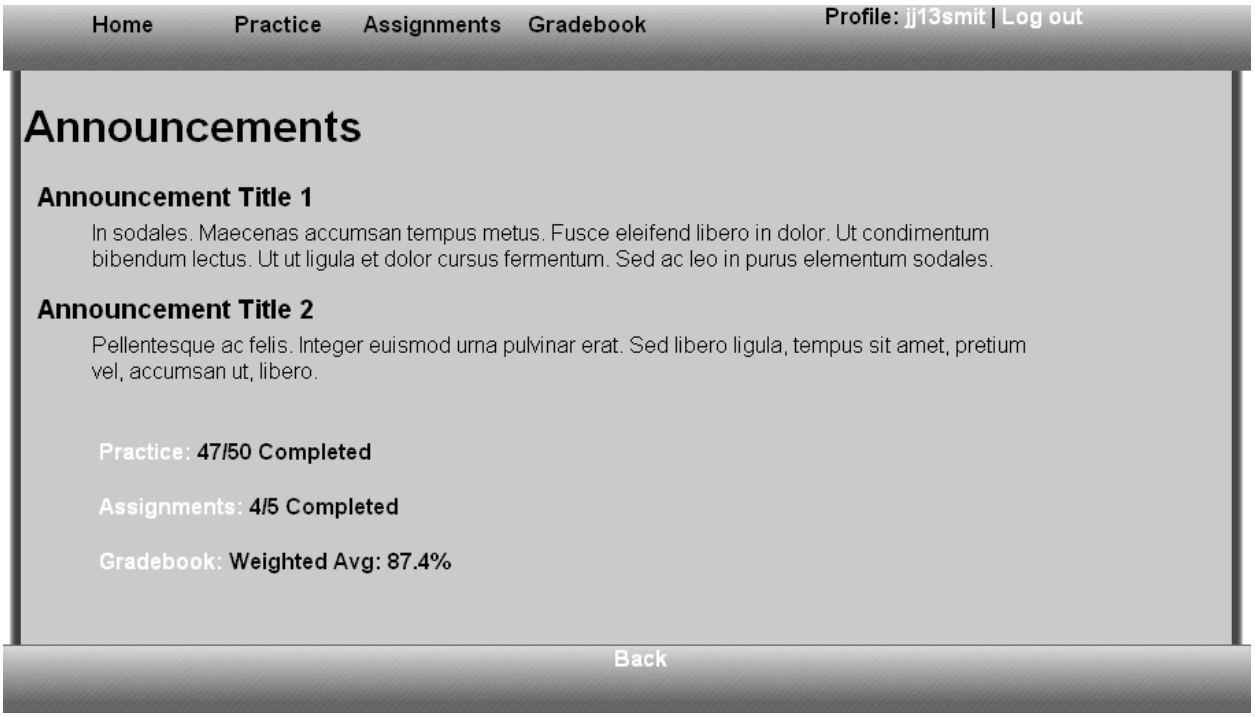


The Student Profile Page will be shown after clicking Register on the Student Registration Page, and anytime the user name link in the upper right corner is clicked. The Student Profile Page will allow a Student to view the existing profile information and edit the information available on this page: first name, last name, password, email address, security question, and security answer. The functionality for adding a course by entering a course key will also be available on this page.

1.2.5 Student - Welcome Page

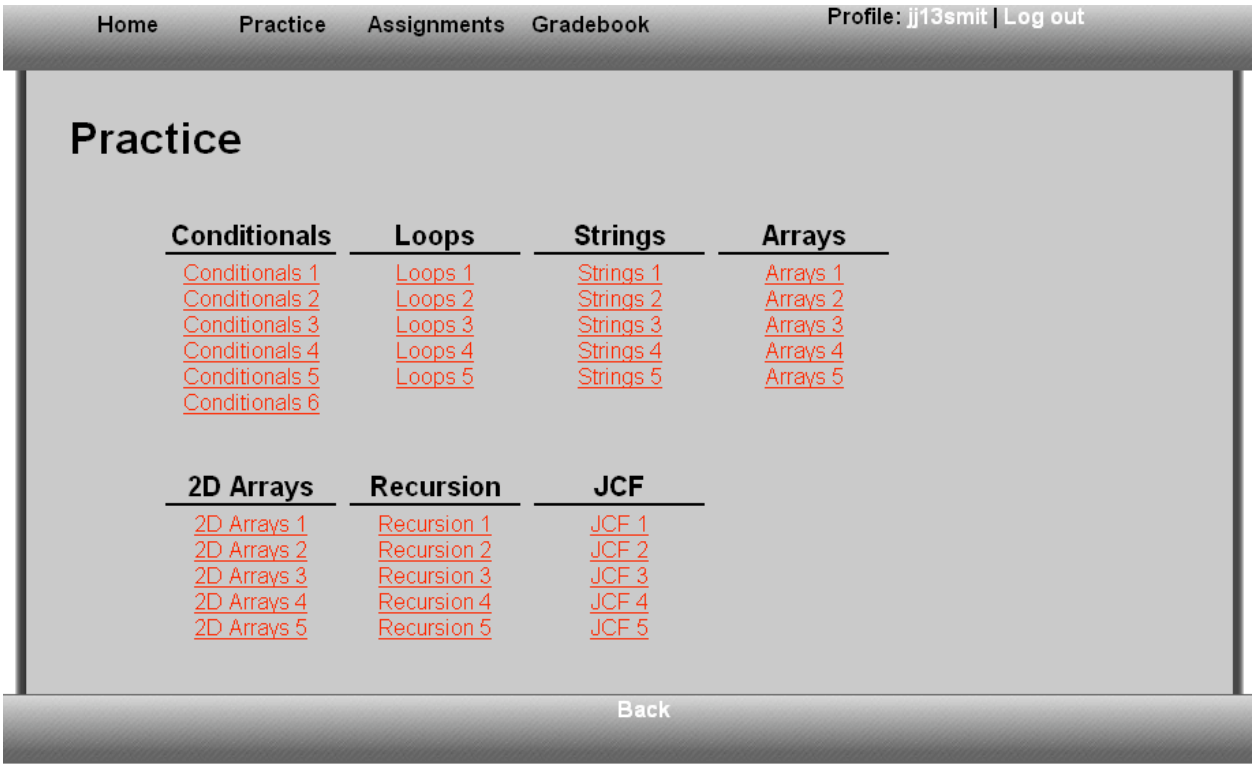
After logging in, a Student will be required to select which course (which they are currently or previously enrolled in) they would like to access.

1.2.6 Student - Home Page



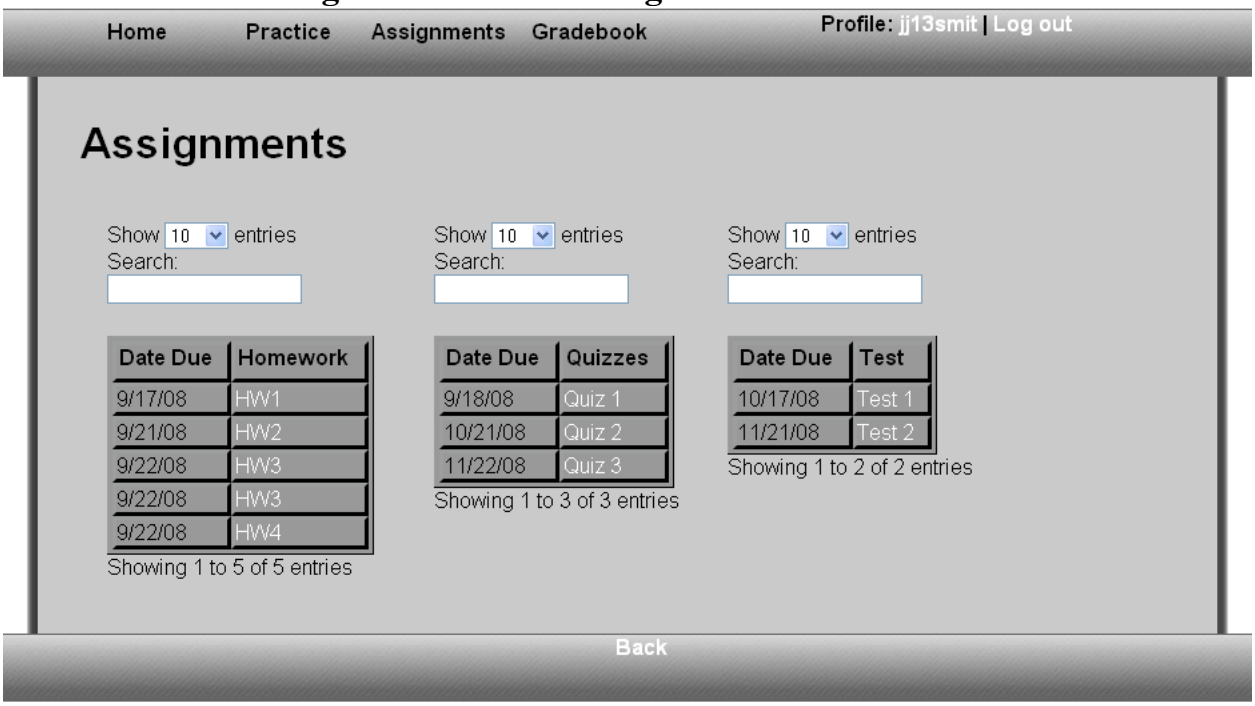
The Student Home page will be shown anytime the *Home* tab is clicked, or after selecting a course on the Student Welcome Page. A Student user will see announcements that have been posted by the Instructor, the Course Coordinator, or the Administrator. A Student will also be able to click on a Practice link, an Assignments link, or a Gradebook link.

1.2.7 Student - Practice Page



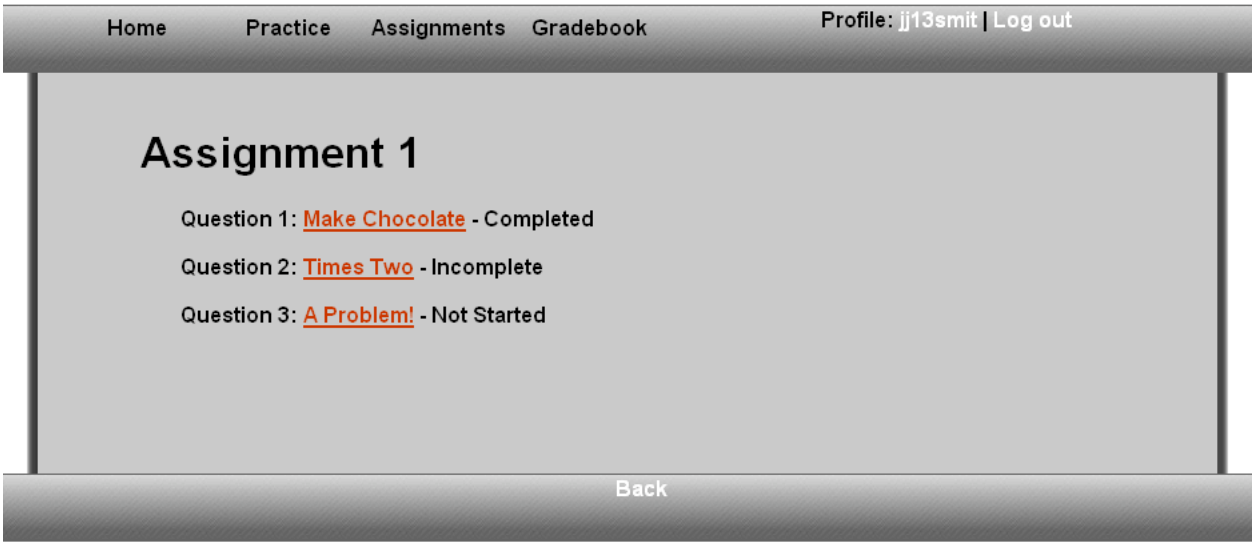
This page can be accessed by a Student either clicking on the *Practice* tab, or clicking on the Practice link on the Home Page. The Student Practice Page displays the categories of questions available and lists questions sets underneath.

1.2.8 Student - Assignment Selection Page



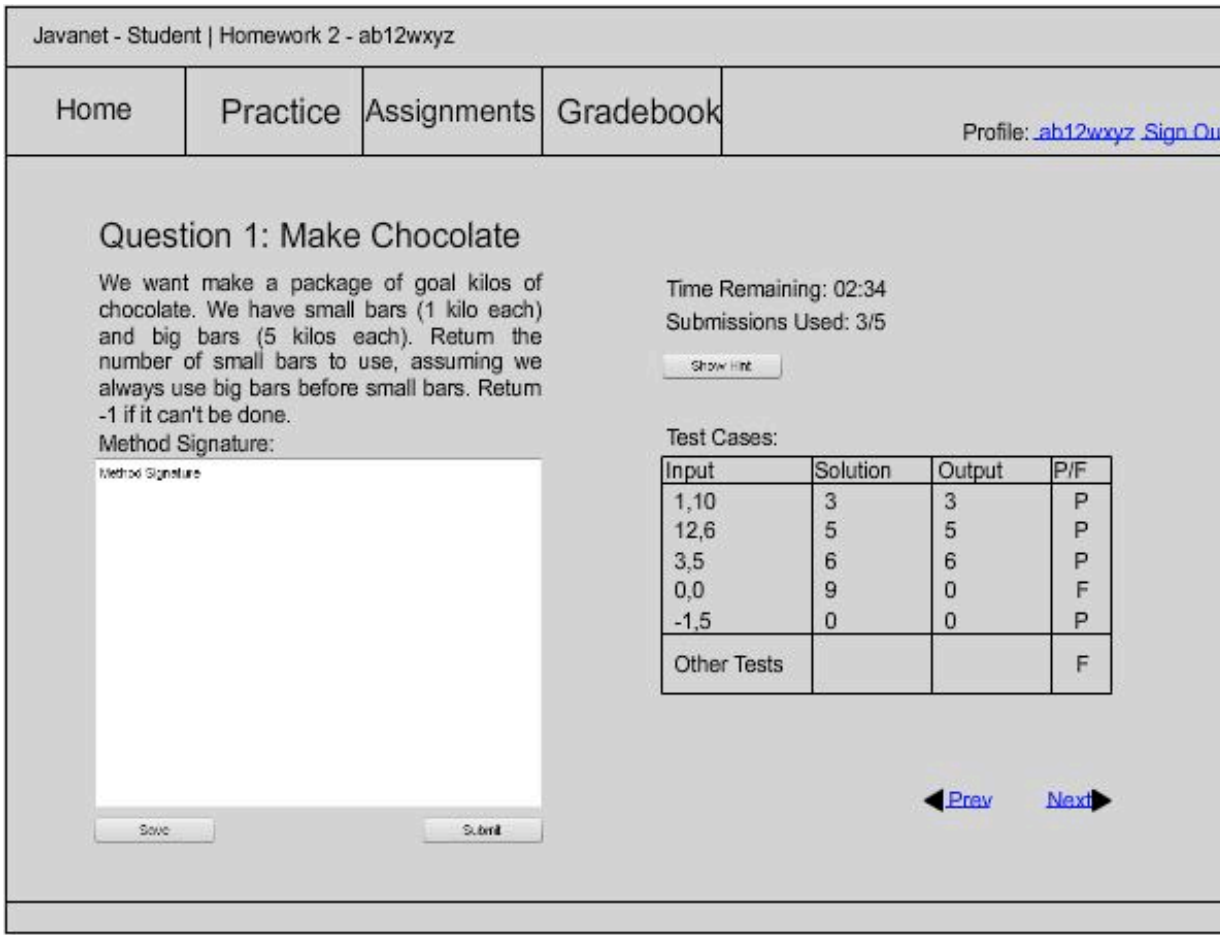
This page can be accessed by a Student either clicking on the *Assignments* tab, or clicking on the Assignments link on the Home Page. The Student Assignment Selection Page displays three tables: homework, quizzes, and tests. Each assignment is listed with its corresponding due date. Each assignment is also an active link that will redirect to the Student Assignment Page.

1.2.9 Student - Assignment Page – for a specific assignment



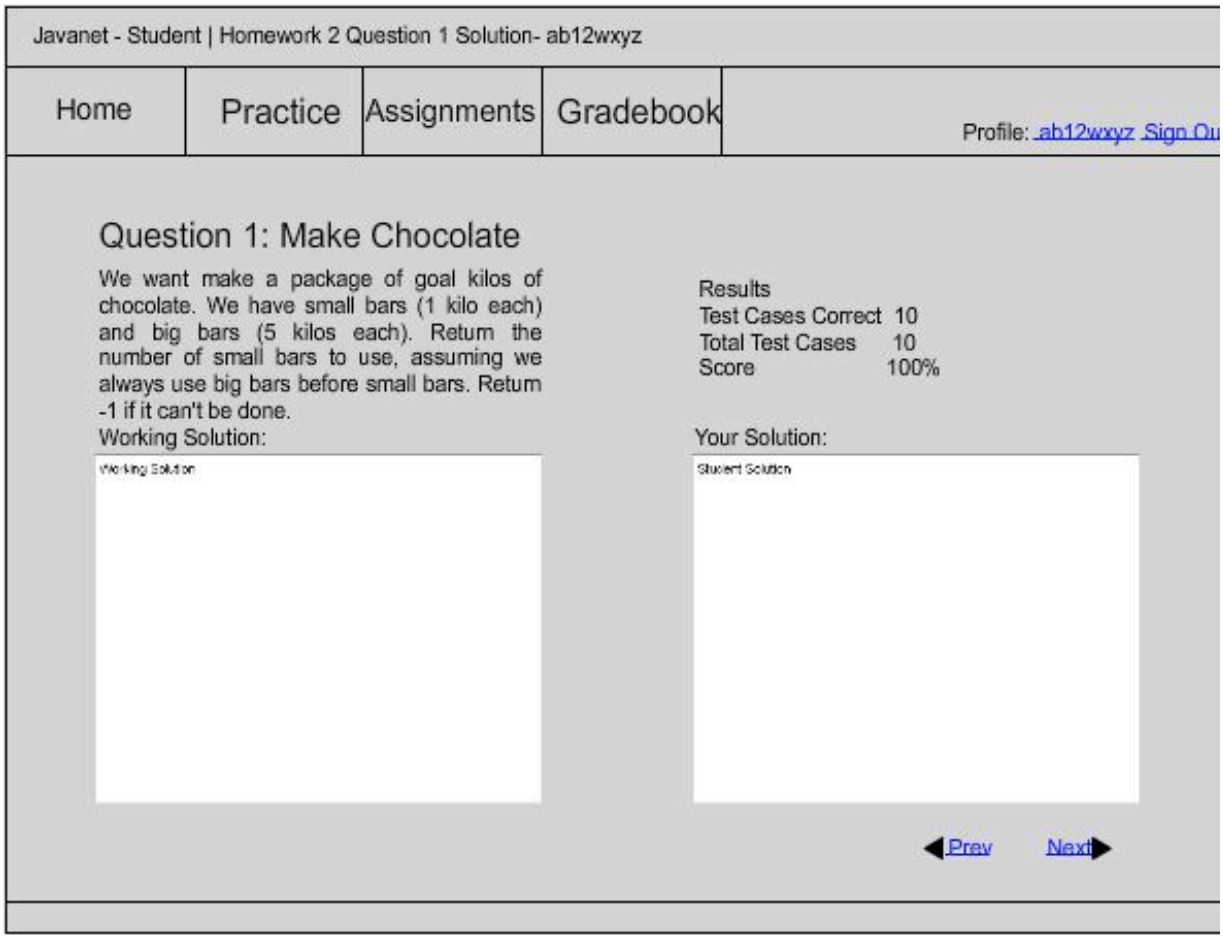
After clicking Homework 2, for example, from the Student Assignment Selection Page, a Student will be brought to this page. There are a list of questions, which are active links that will redirect to the Student Answer Question Page, that make up the assignment and a progress indicator.

1.2.10 Student - Answer Question Page



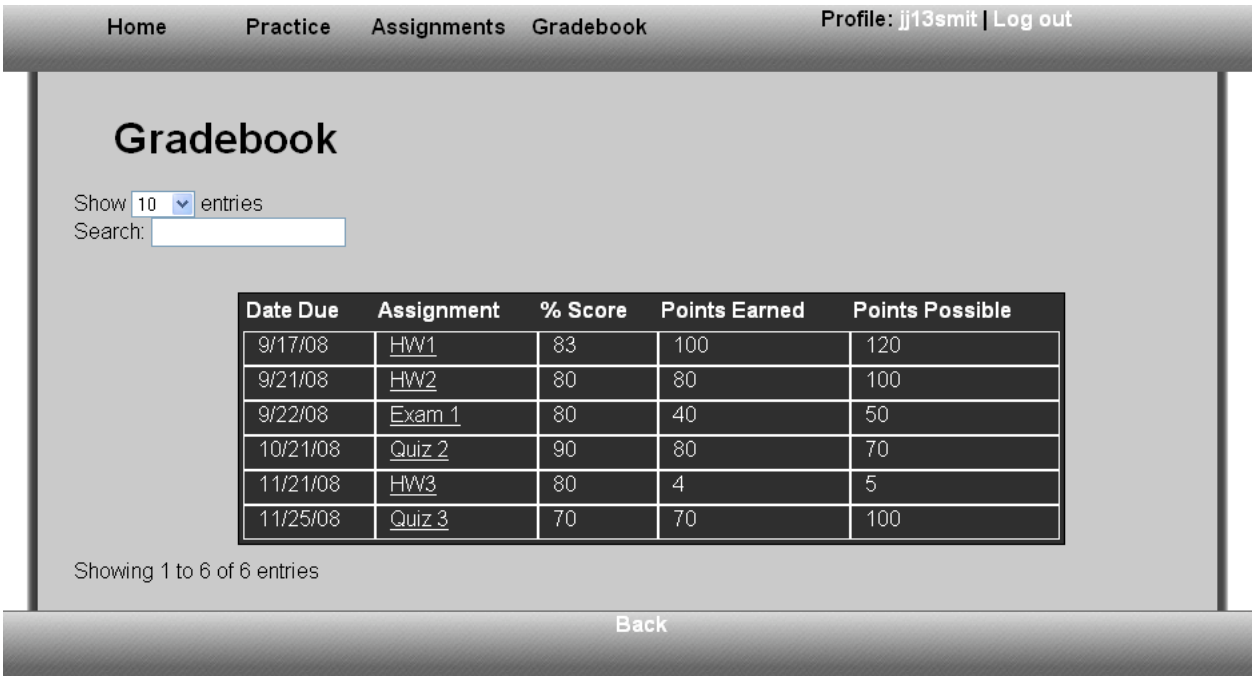
After a question is selected from the Student Assignment Page, the student will see this page. The question will be displayed along with a provided method signature and a text box to write the question solution. In addition, a Student can see how many submissions are remaining and how much time is remaining. A Student may also click Show Hint to view any hints provided by the Instructor. A Student can also Save their source code by clicking on the Save button. Viewable Test Cases are also provided with results.

1.2.11 Student - Solution Page



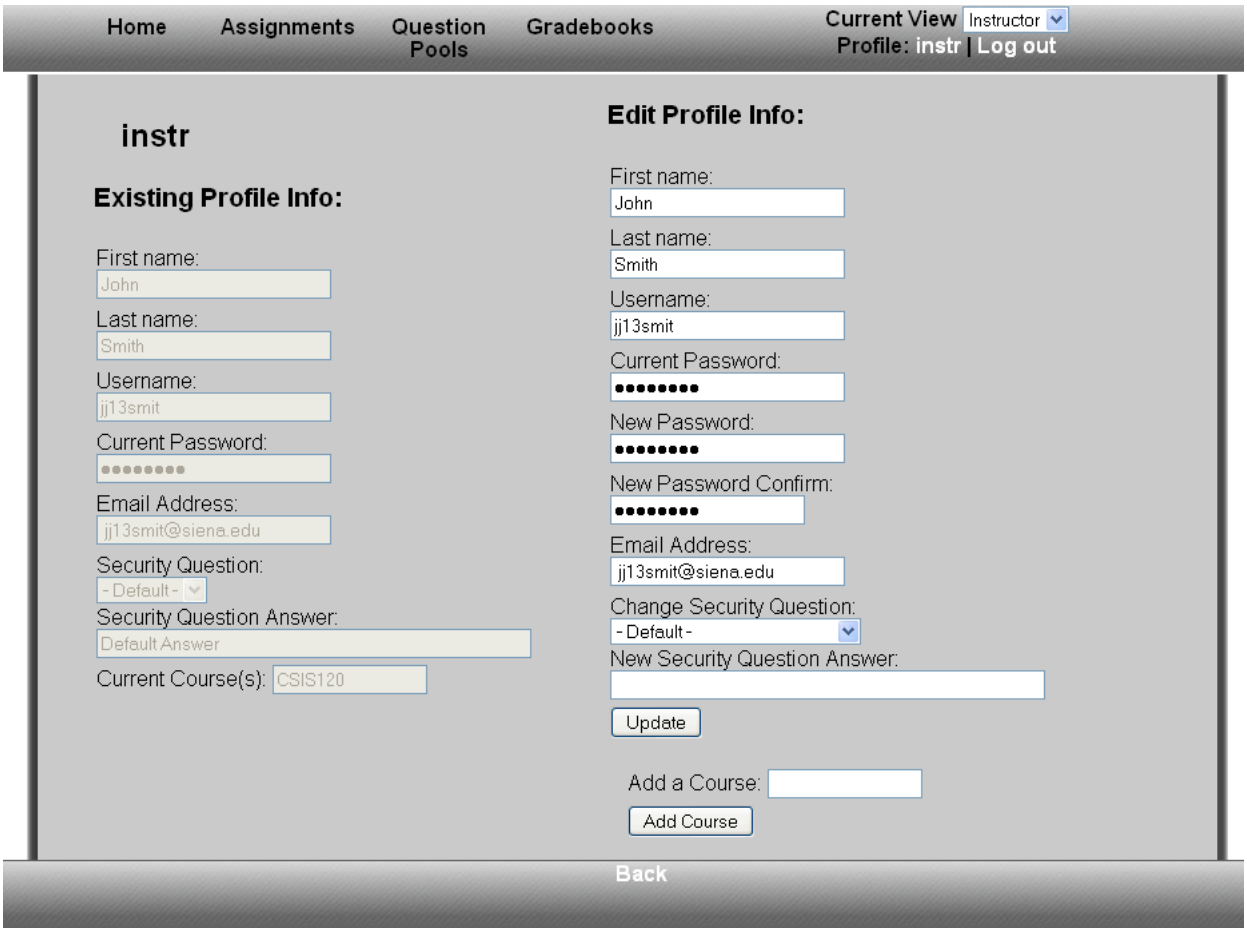
This page can be accessed after an assignment has been completed. This page will provide a Student user with a read only version of his or her solution to the question being viewed. This page will also provide the percentage correct, test cases correct, and the total number of test cases (viewable and non-viewable).

1.2.12 Student - Gradebook Page



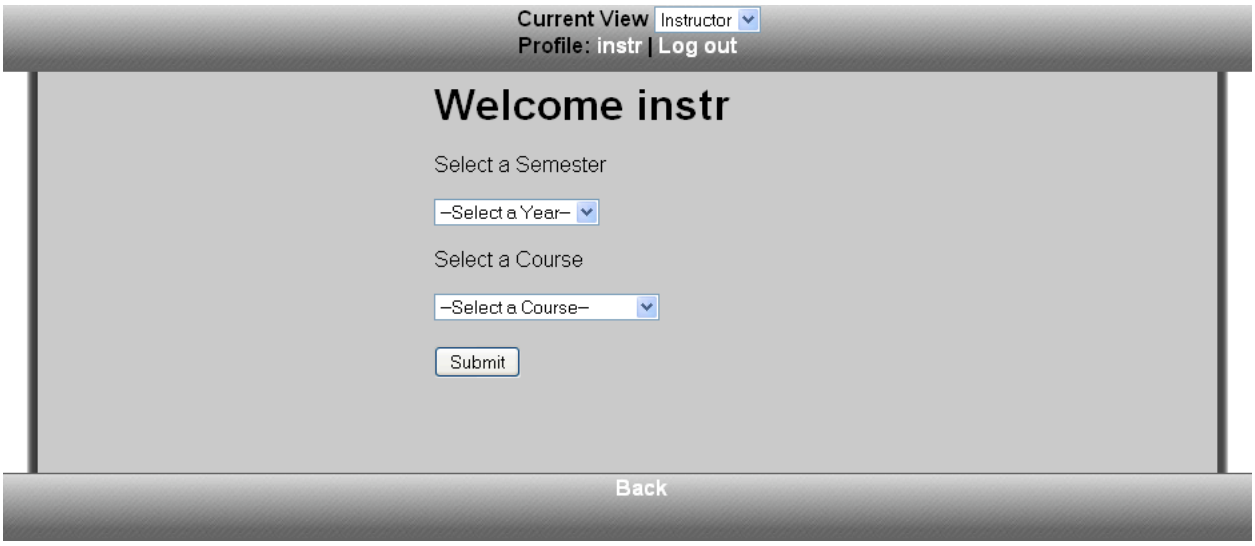
This page can be accessed by a Student either clicking on the *Gradebook* tab, or by clicking on the Gradebook link on the Student Home Page. In this case, the gradebook is ordered by due date, as displayed by the blue background of the Due Date tab. The gradebook may also be sorted by any of the other columns: assignment, %score, points earned, or points possible, by simply clicking on the heading. The Gradebook will show the percentage, points possible and points earned on each assignment given.

1.2.13 Instructor - Profile Page



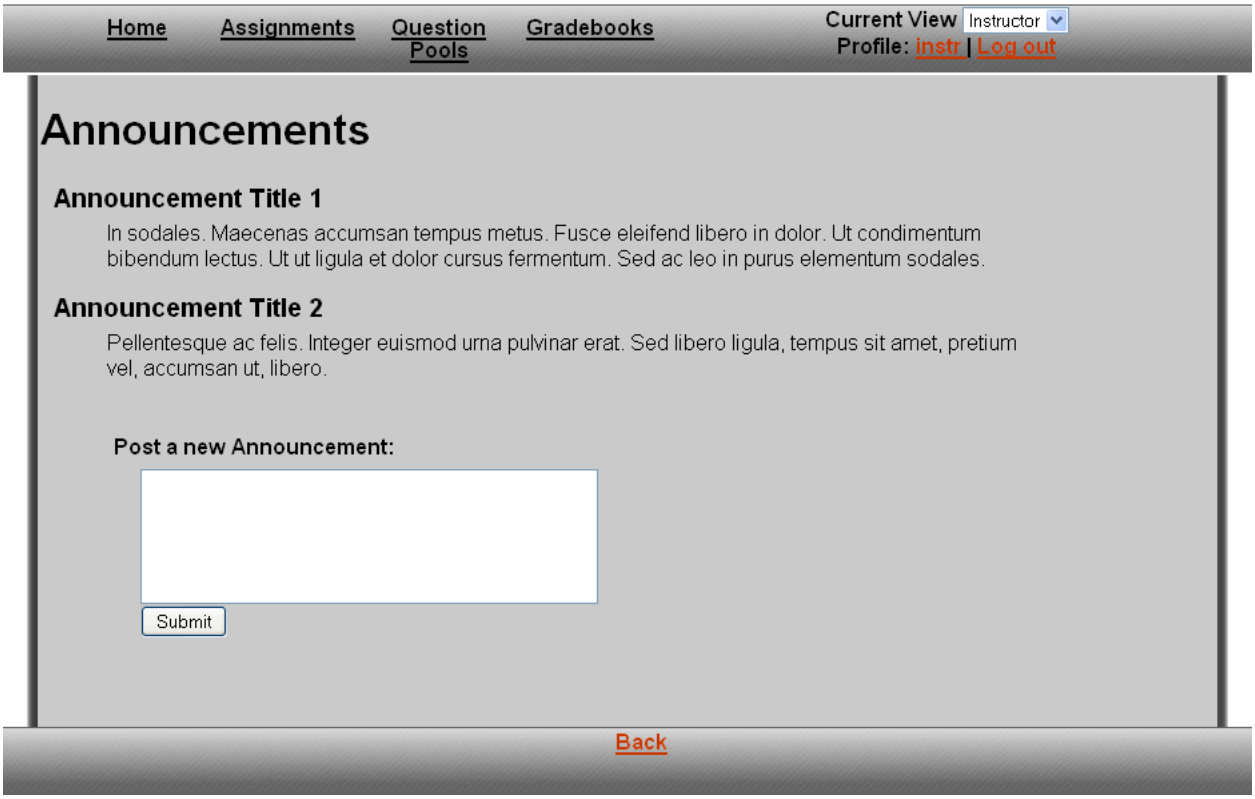
The Instructor Profile Page will be shown anytime the user name link in the upper right corner is clicked. The Instructor Profile Page will allow an Instructor to view the existing profile information and edit the information available on this page: first name, last name, password, email address, security question, and security answer. Additionally, an Instructor has the ability to join a specific course section or to copy course information from a past semester on this page.

1.2.14 Instructor - Welcome Page



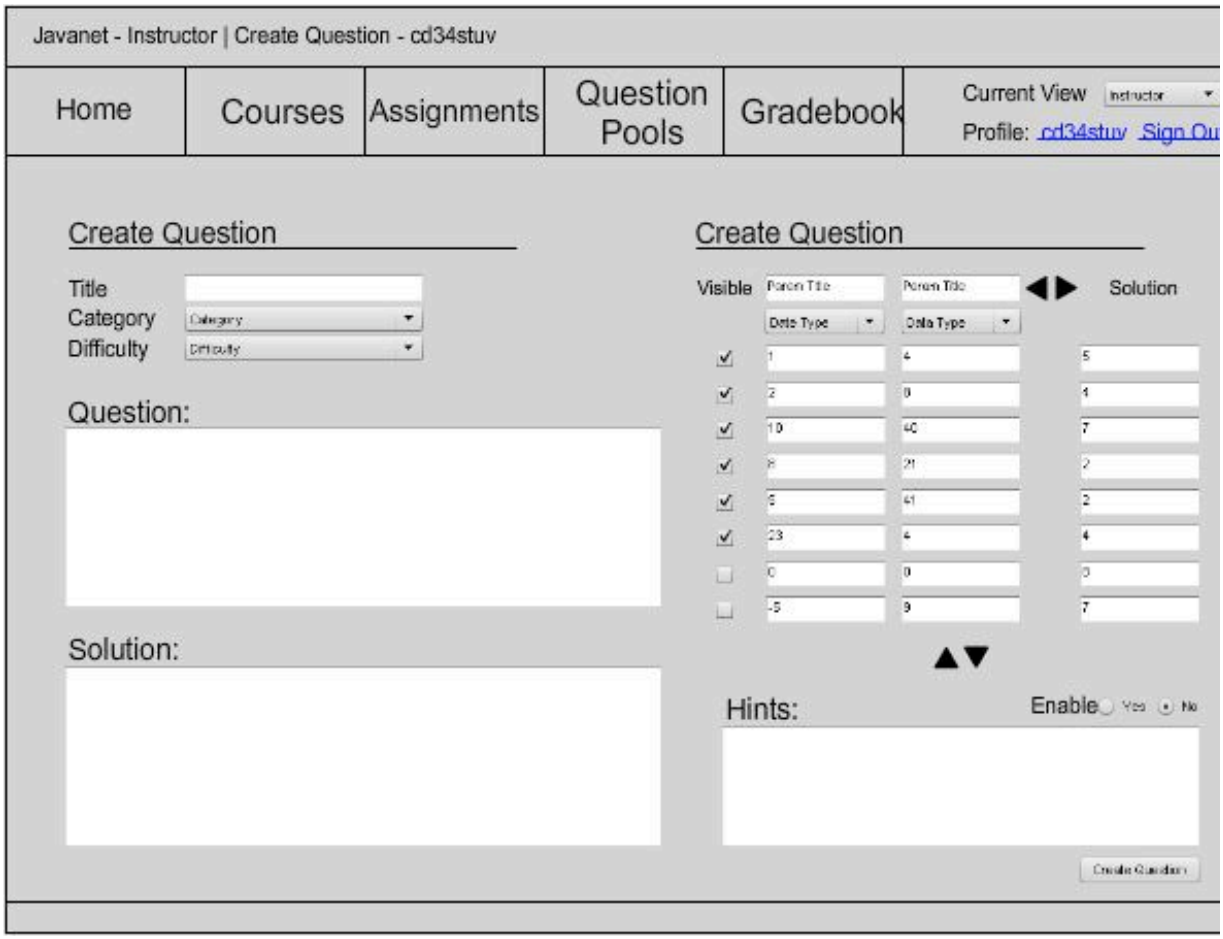
After logging in, an Instructor will be required to select which course they would like to access.

1.2.15 Instructor - Home Page



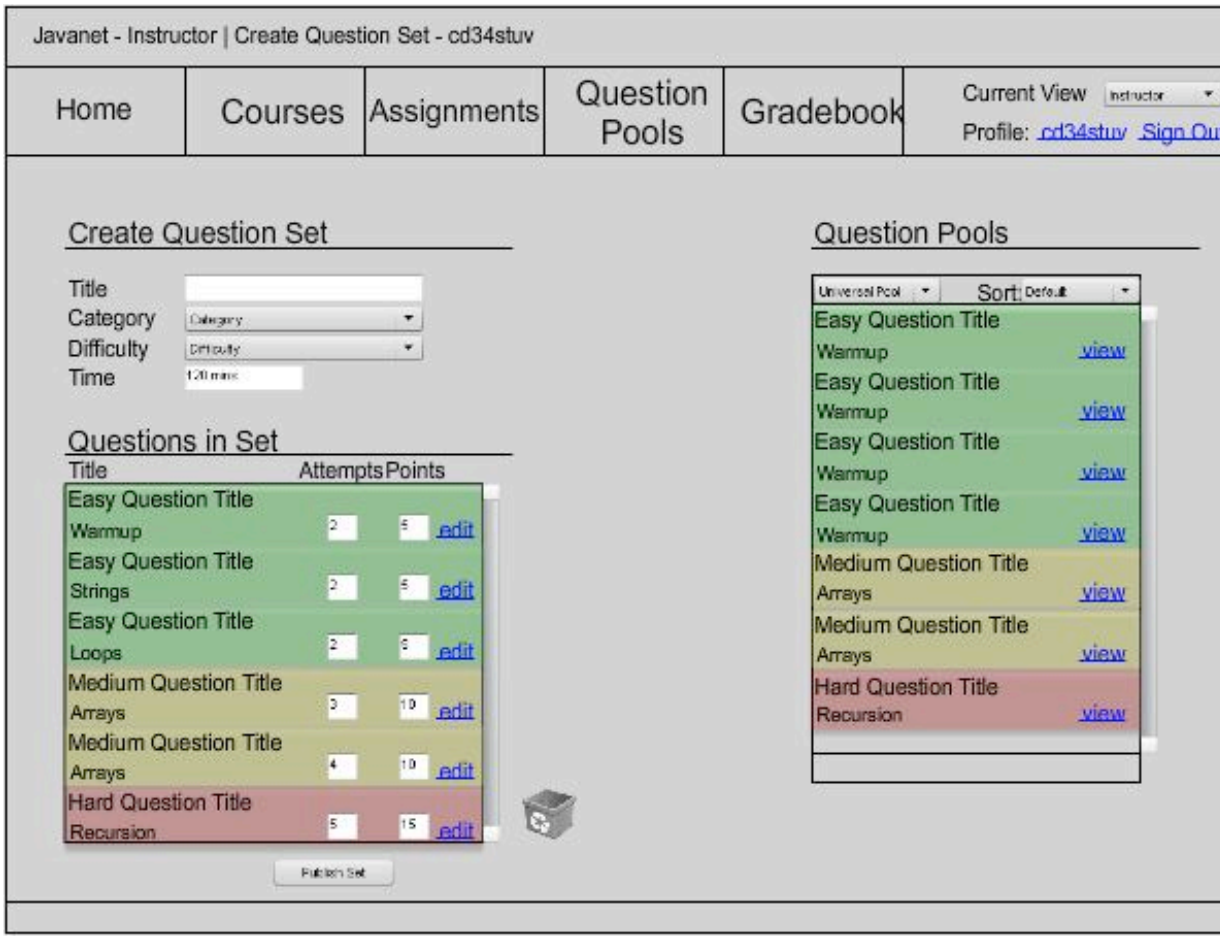
The Instructor Home page will be shown anytime the *Home* tab is clicked, or after selecting a course on the Instructor Welcome Page. An Instructor user will see the most recent announcements that have been posted by either themselves, the Course Coordinator, or the Administrator. An Instructor will also be able to edit or delete his or her own previous announcements.

1.2.16 Instructor - Create Question Page



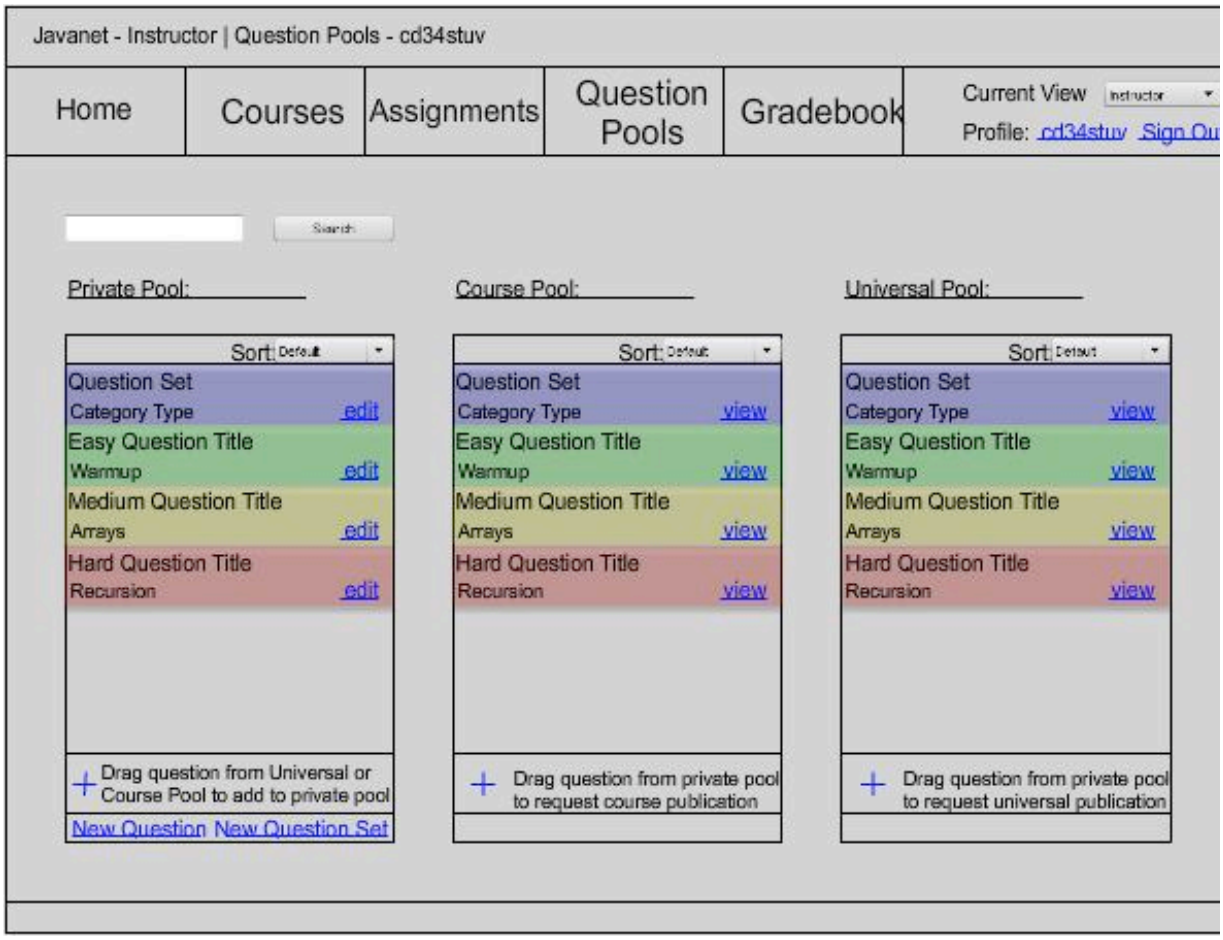
When creating a question, an Instructor will see this page. He or she can select a title, category, and difficulty for the question. An Instructor can then write the question and solution as well. An Instructor will also be able to add parameters for test sets and hints if it is desired. The arrows in the create test set section provide the functionality of creating more test sets and more parameters.

1.2.17 Instructor - Create Question Set Page



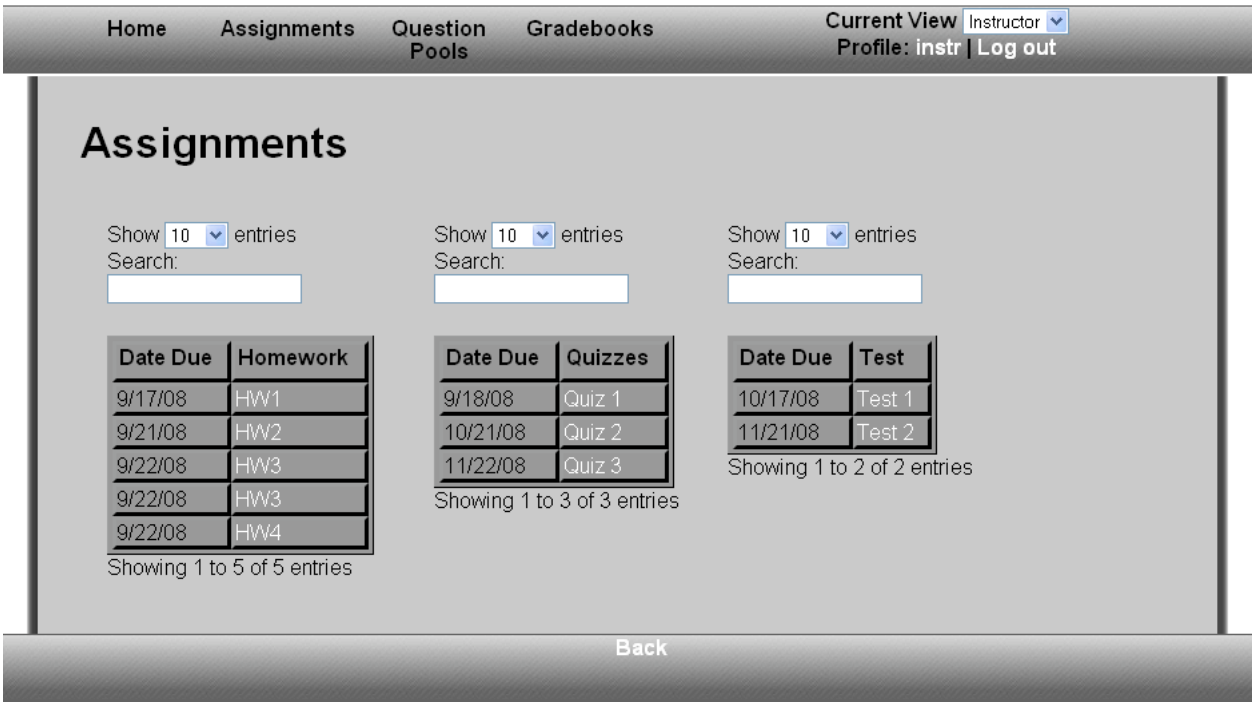
When creating a question set, an Instructor will see this page. An Instructor can create a title, select a category, select a difficulty level, and a time limit for completion. He or she will be able to edit each question in the set, as well as adjust the amount of attempts and, or points available. Instructors are able to build a question set with questions from any of their three pools.

1.2.18 Instructor - Pools Page



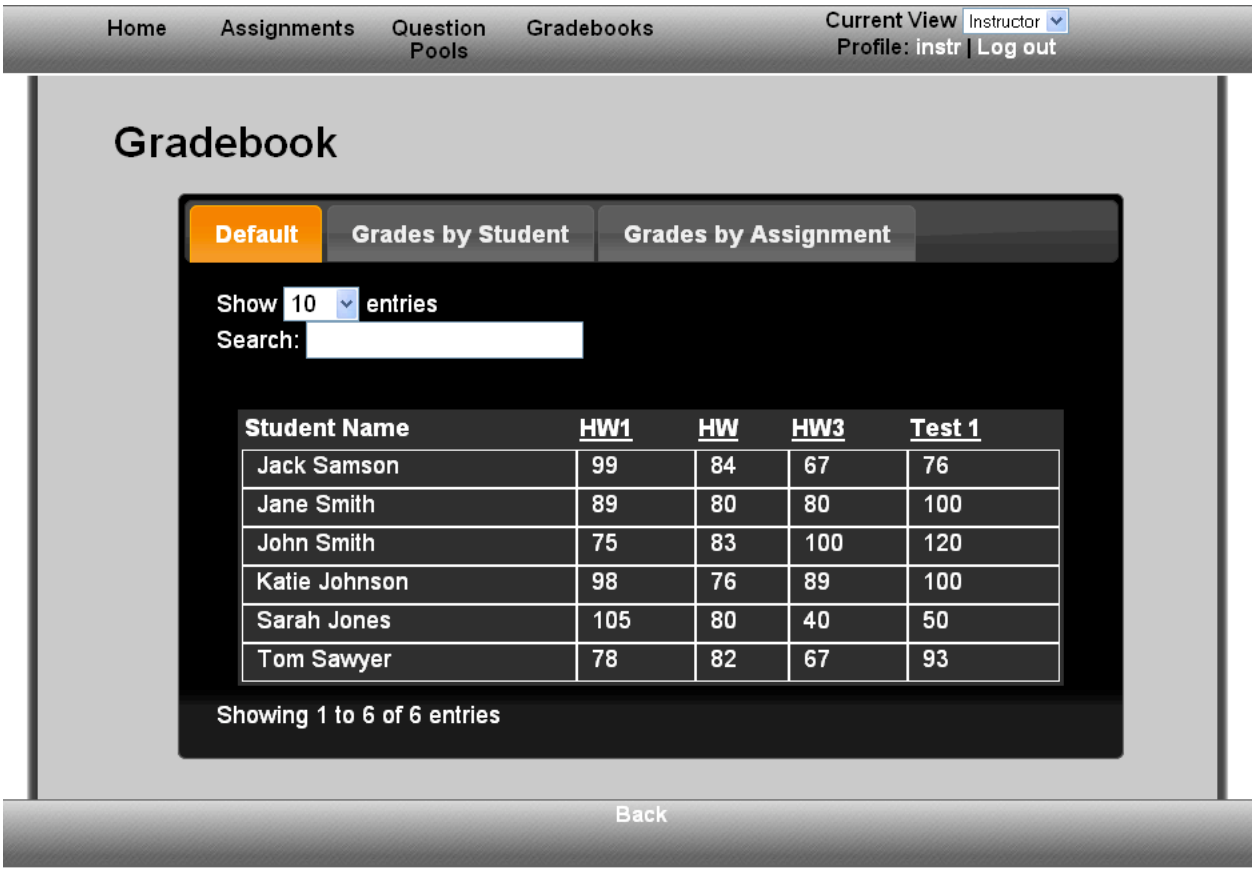
Selecting the *Question Pools* tab will bring you an Instructor this page. On this page, an Instructor may drag and drop questions to and from certain pools. An Instructor can also search the pools by entering a search string. In addition, he or she can decide to create a new question or question set here.

1.2.19 Instructor - Assignment Listing Page



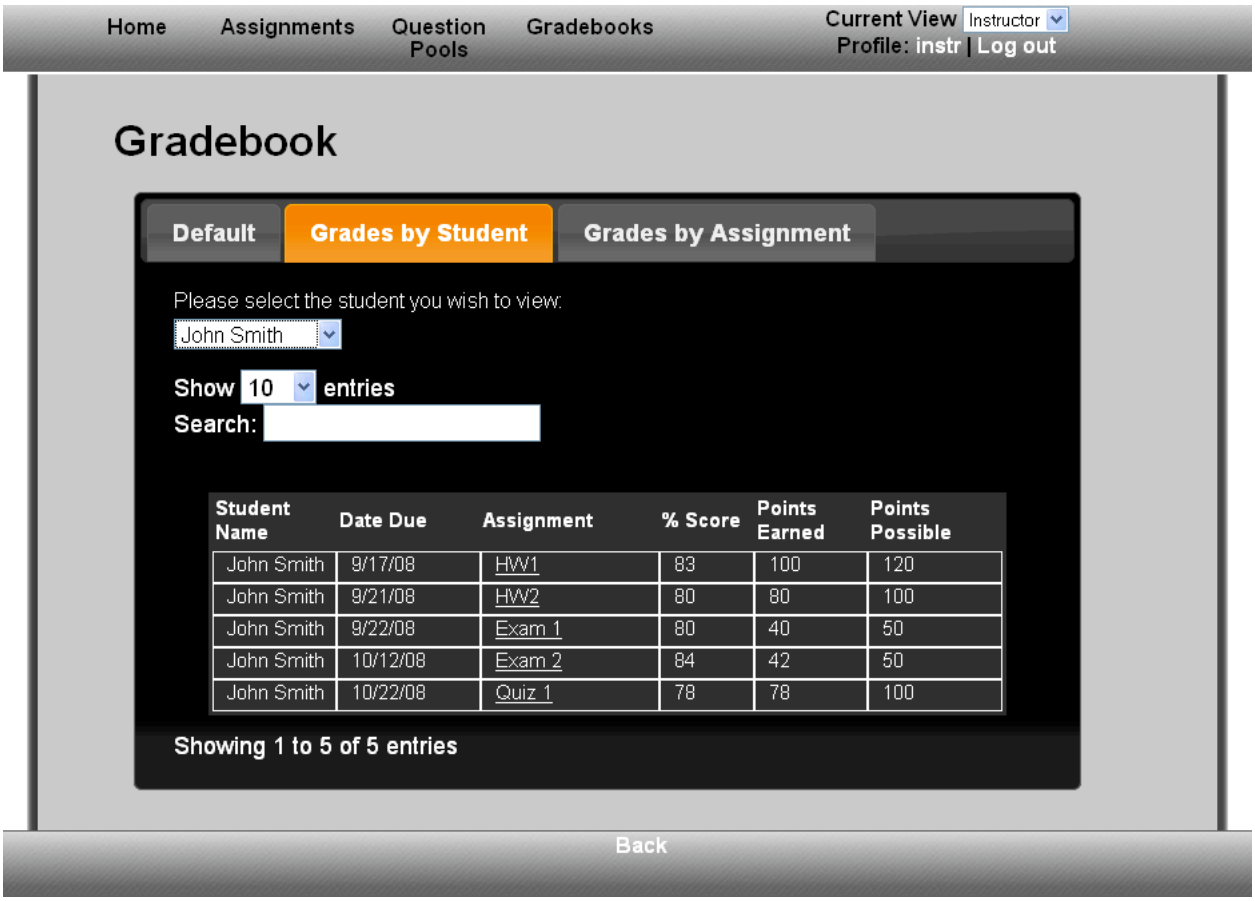
This page can be accessed by an Instructor clicking on the *Assignments* tab. This page displays three tables: homework, quizzes, and tests. Each assignment is listed with its corresponding due date. Each assignment is also an active link that will redirect to the Gradebook.

1.2.20 Instructor – Gradebook Page – default view



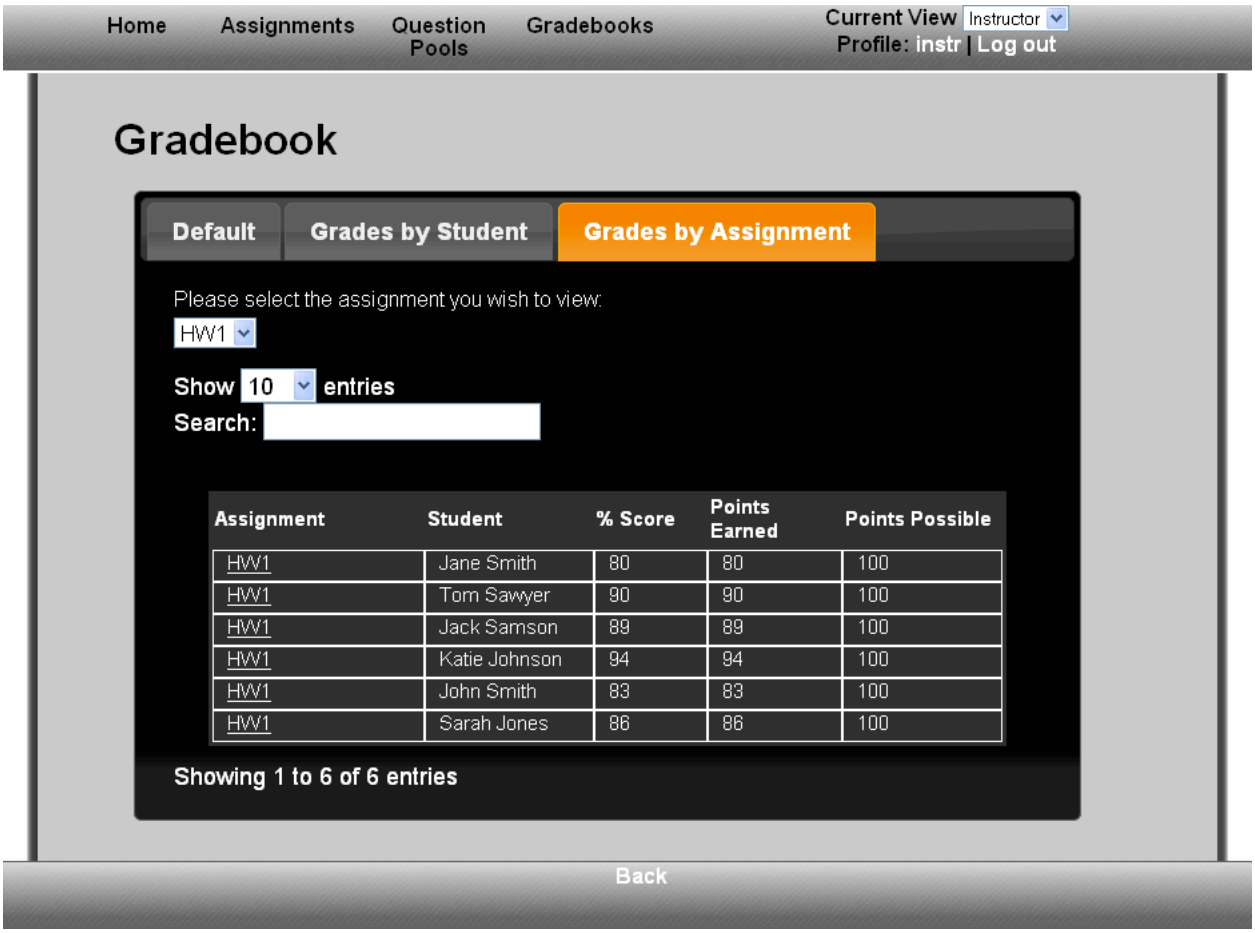
An Instructor can access this page by selecting the *Gradebook*. This page serves as the default Gradebook. This page will display the students and their averages on each assignment given. Instructors may also choose to view the Gradebook by Student or by Assignment. Each Student name and each assignment are also active links.

1.2.21 Instructor - Gradebook Page - sorted by Students



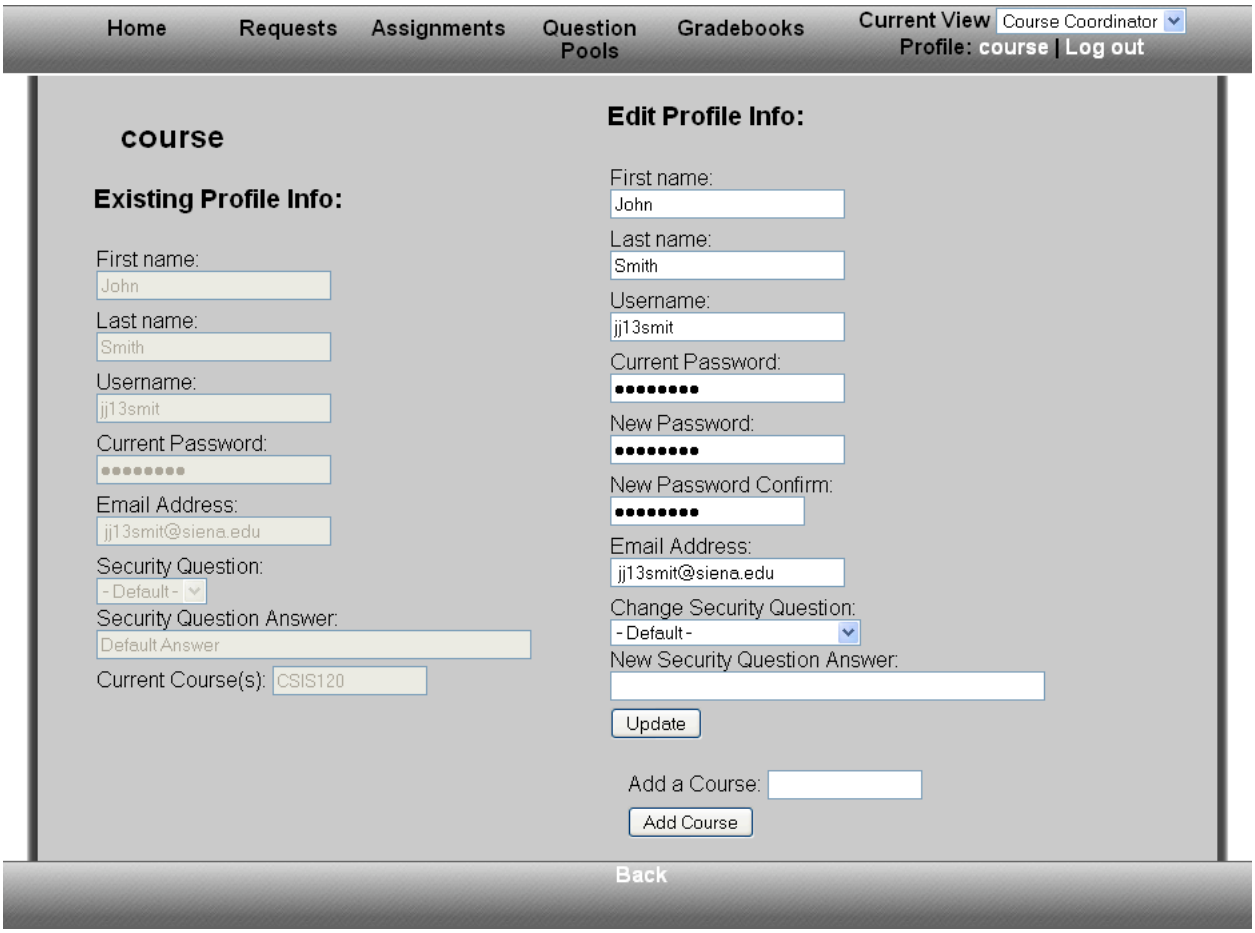
This page can be accessed from the default Gradebook by clicking on a Student name, or by clicking on the Grades By Student tab in the Gradebook. This page shows an Instructor each assignment with its corresponding score, points possible and points earned, for one particular student.

1.2.22 Instructor - Gradebook Page - sorted by Assignments



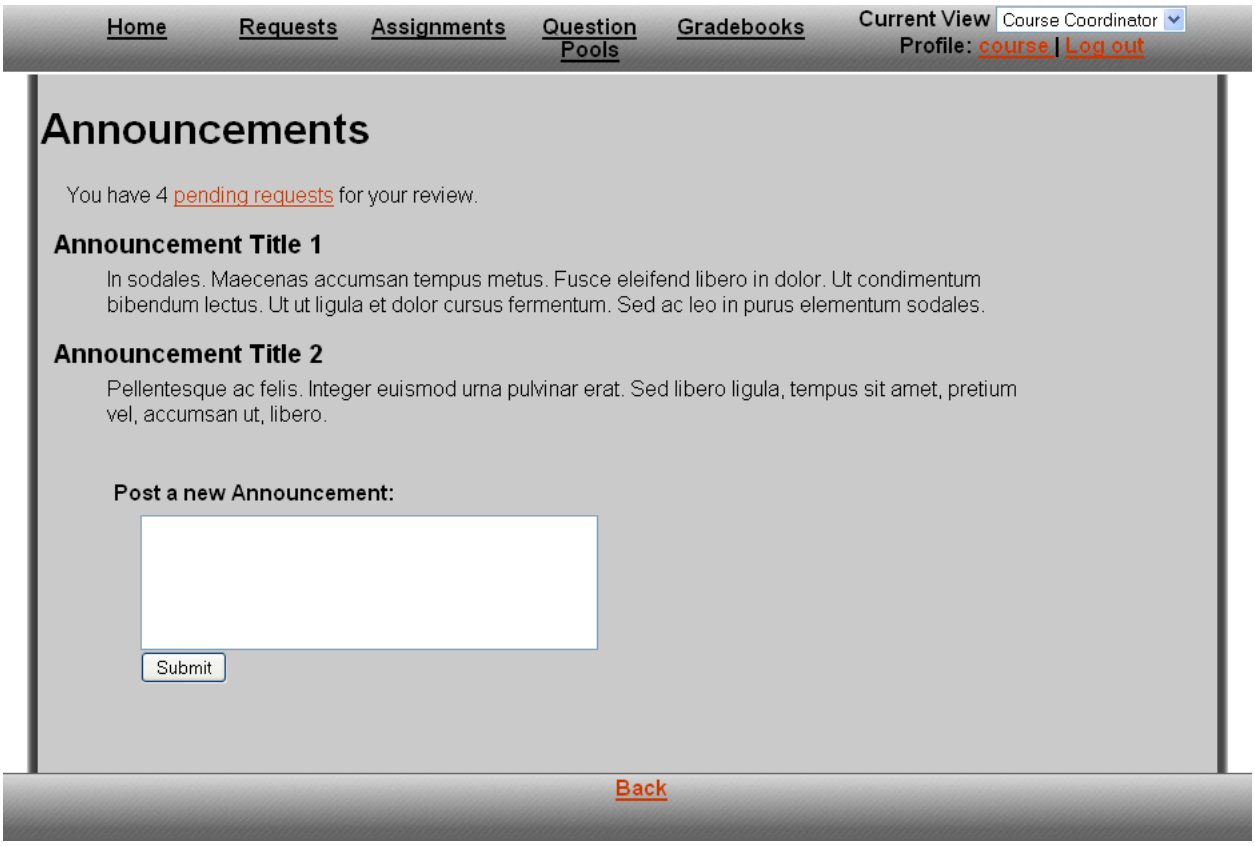
This page can be accessed from the default Gradebook by clicking on a particular assignment, or by clicking on the Grades By Assignment tab in the Gradebook. This page shows an Instructor all of the Students with their corresponding scores, points possible and points earned, for one particular assignment.

1.2.23 Course Coordinator - Profile Page



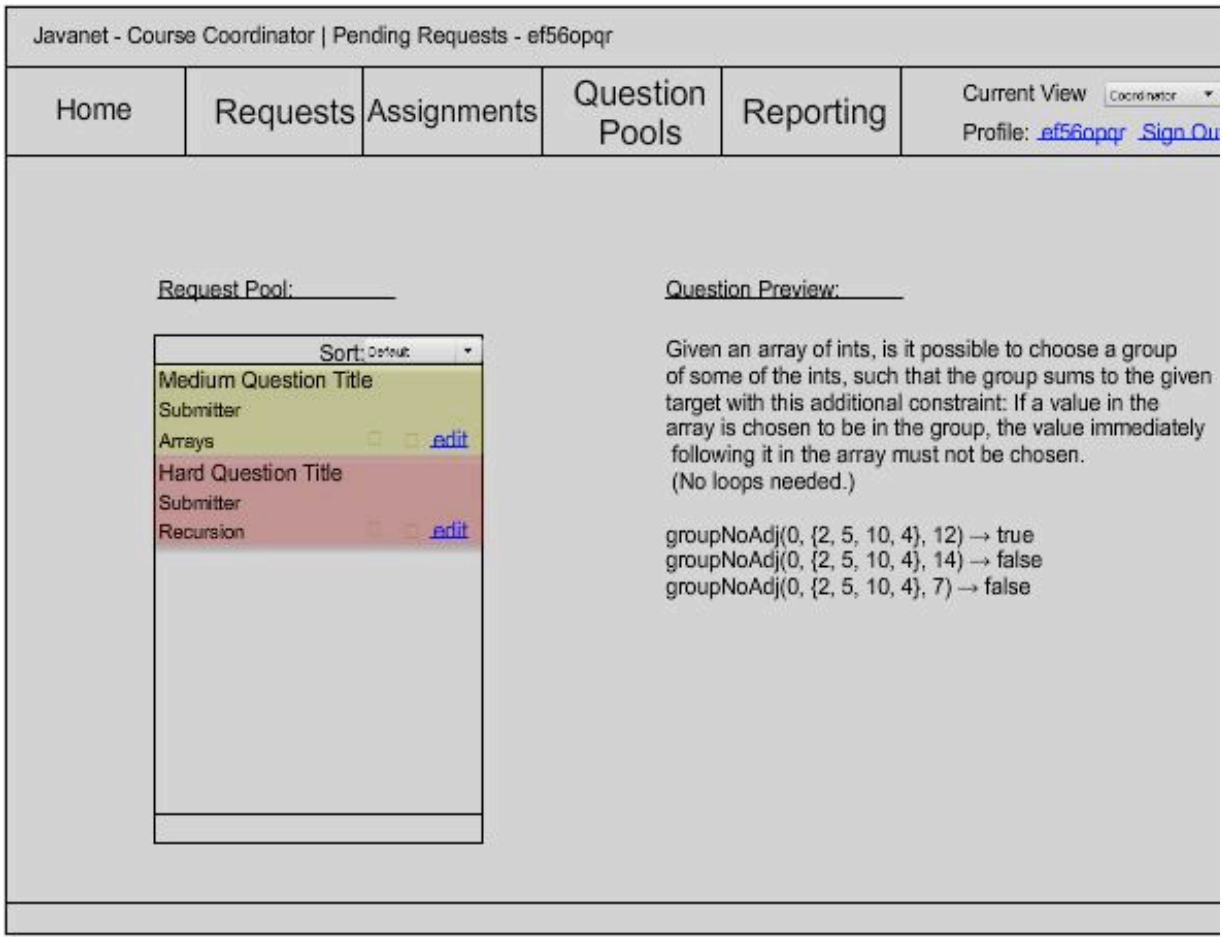
The Course Coordinator Profile Page will be shown anytime the user name link in the upper right corner is clicked. The Course Coordinator Profile Page will allow a Course Coordinator to view his or her existing profile information and edit the information on this page: first name, last name, password, email address, security question, and security answer.

1.2.24 Course Coordinator - Welcome Page



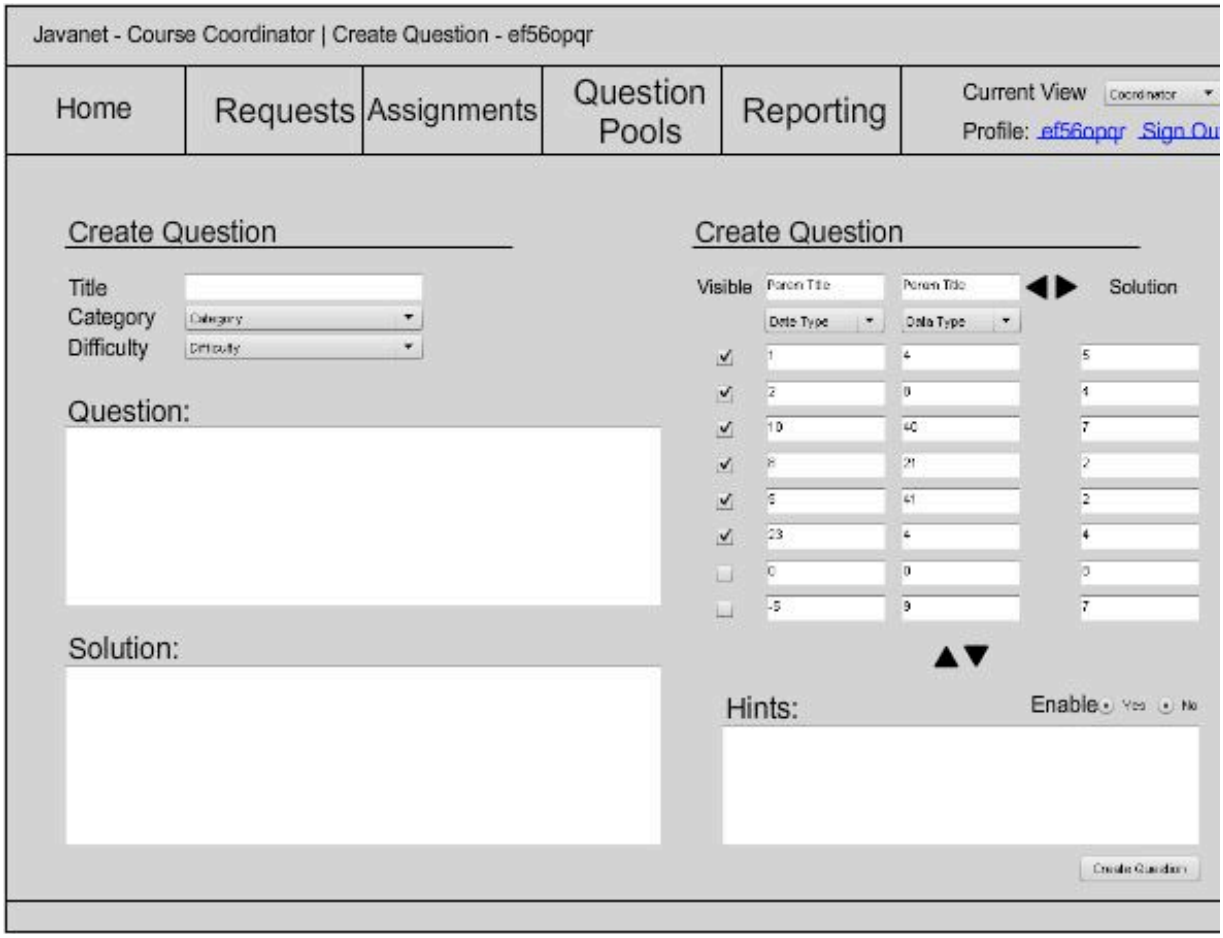
After logging in, the Course Coordinator will be brought to this page. He or she will see any pending requests, the most recent announcements posted and be able to post a new announcement for that course. The Course Coordinator may also edit or delete any announcements that they have posted.

1.2.25 Course Coordinator - Pending Requests Page



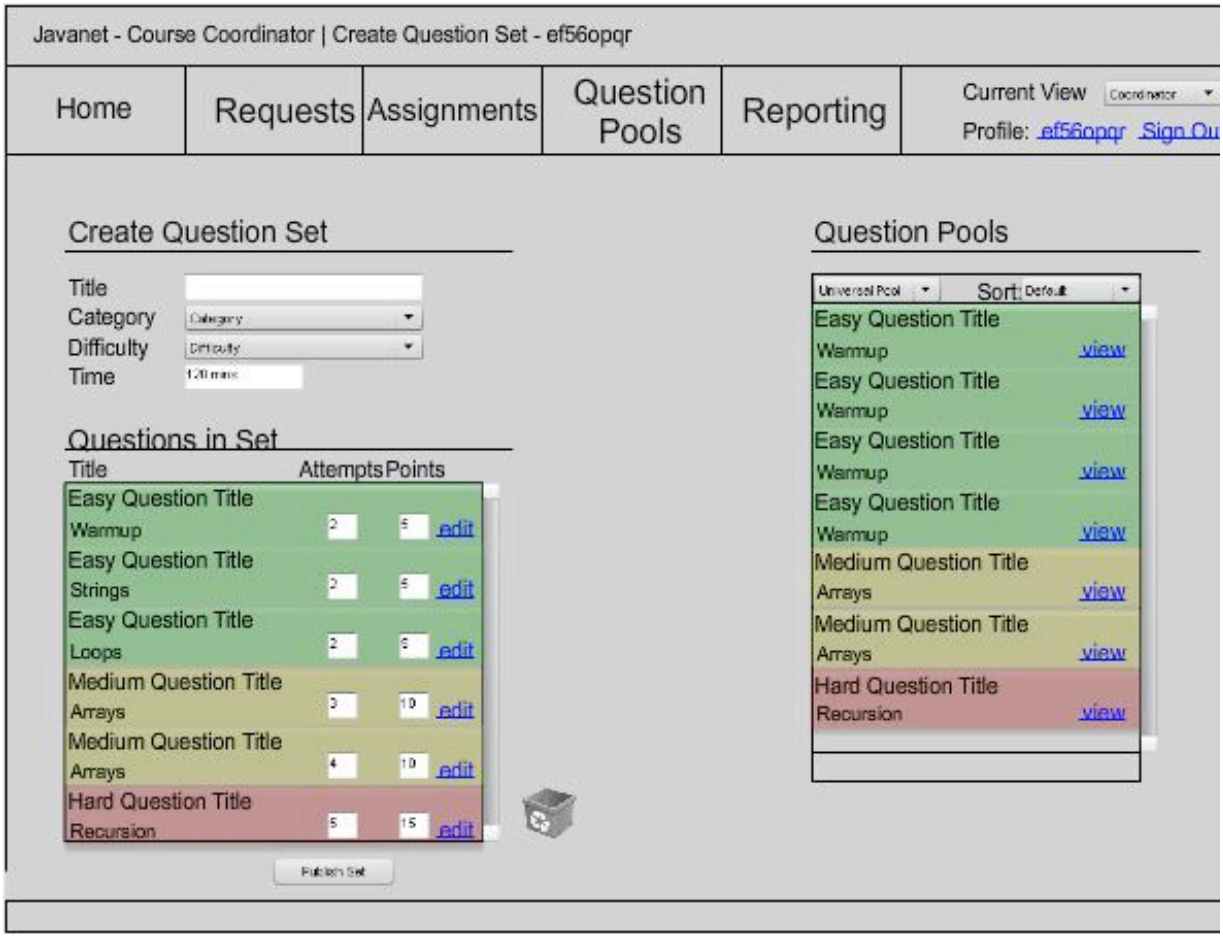
The Pending Requests Page will show the specific pending requests that the Course Coordinator has. He or she can choose to approve, deny, or edit a question or question set.

1.2.26 Course Coordinator - Create Question Page



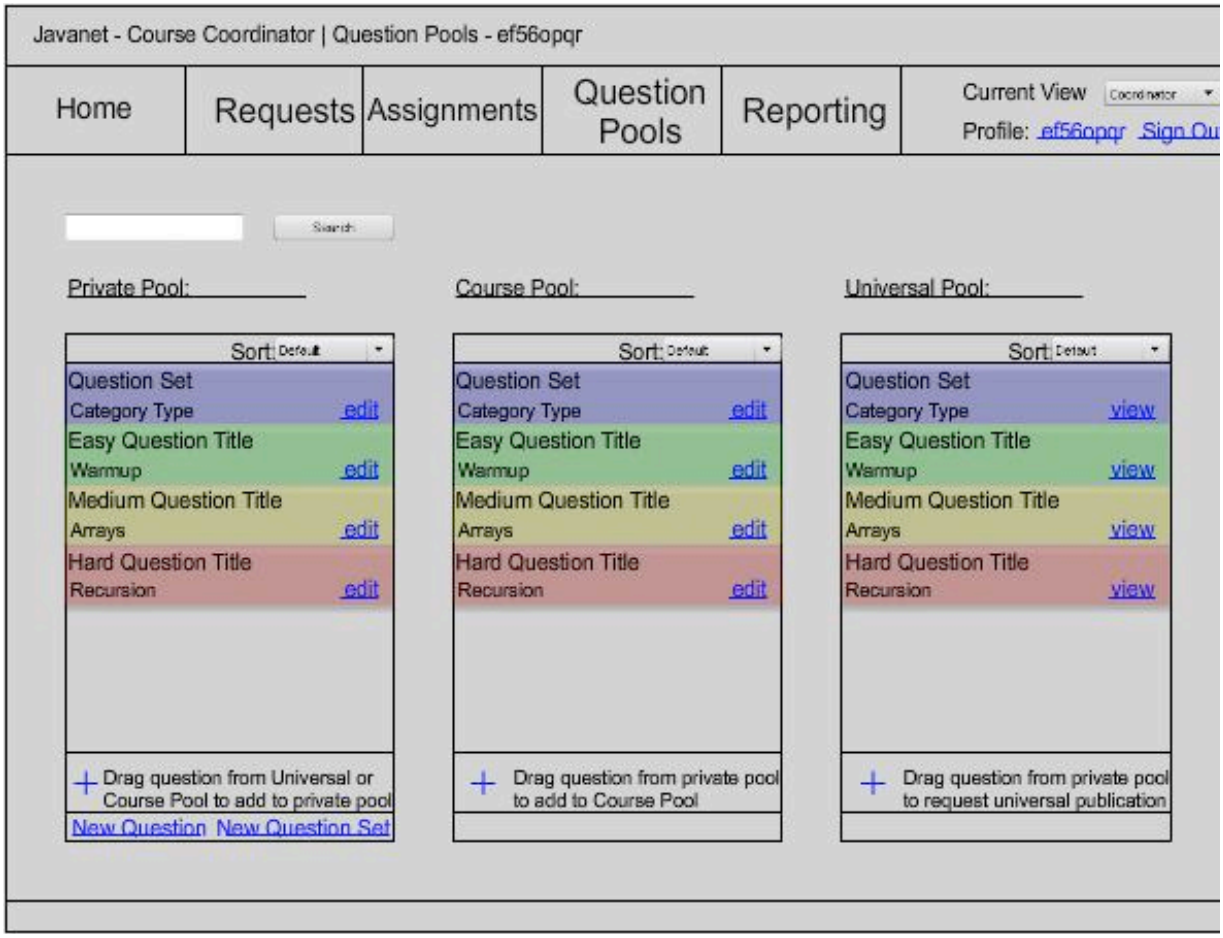
A Course Coordinator will see this page when creating a question. A Course Coordinator can select a title, category, and difficulty for the question. A Course Coordinator can then write the question and solution as well. A Course Coordinator will also be able to add parameters for test sets and hints if it is desired. The arrows in the create test set section provide the functionality of creating more test sets and more parameters.

1.2.27 Course Coordinator - Create Question Set Page



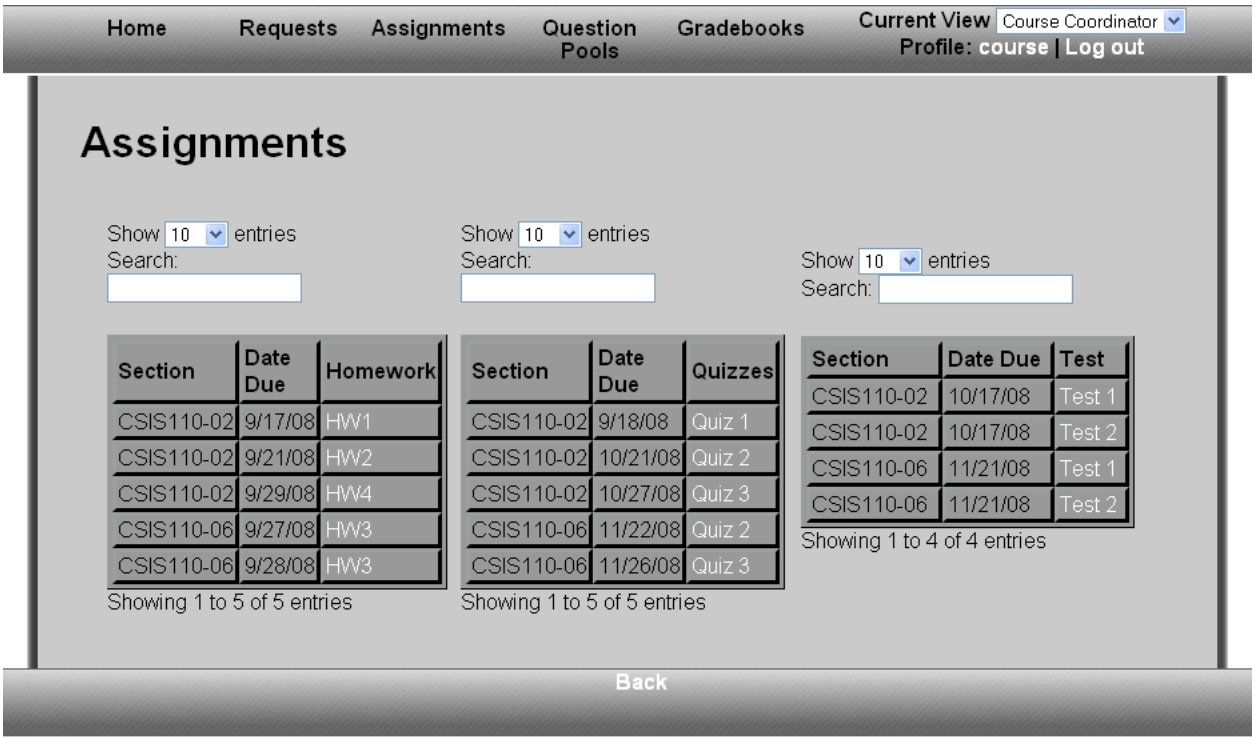
When creating a question set, a Course Coordinator will see this page. A Course Coordinator can create a title, select a category, select a difficulty level, and a time limit for completion. A Course Coordinator will be able to edit each question in the set, as well as adjust the amount of attempts and, or points available. Course Coordinators are able to build a question set with questions from any of their three pools.

1.2.28 Course Coordinator - Pools Page



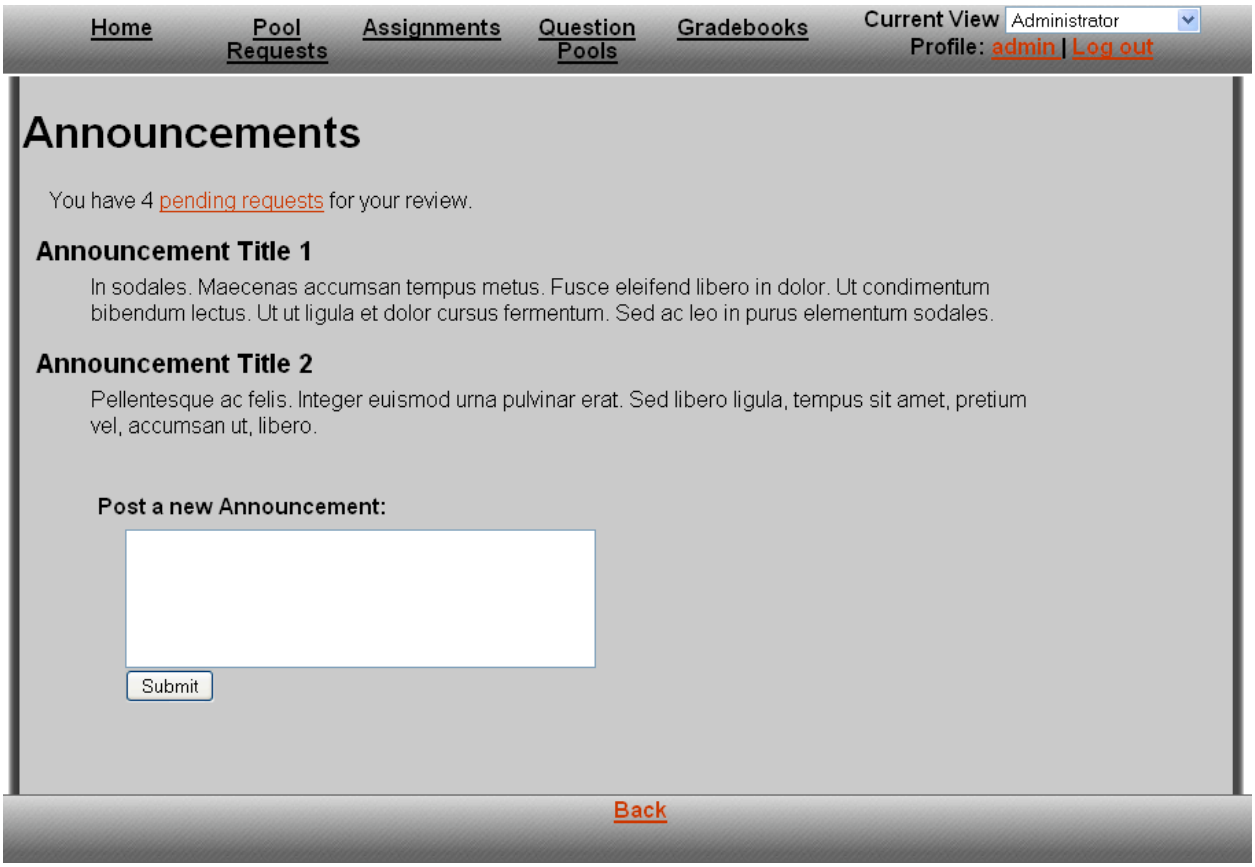
Selecting the *Question Pools* tab will bring a Course Coordinator to this page. On this page, a Course Coordinator may drag and drop questions to and from certain pools. A Course Coordinator can also search the pools by entering a search string. In addition, a course coordinator can decide to create a new question or question set by clicking links on this page.

1.2.29 Course Coordinator - Assignment Listing Page



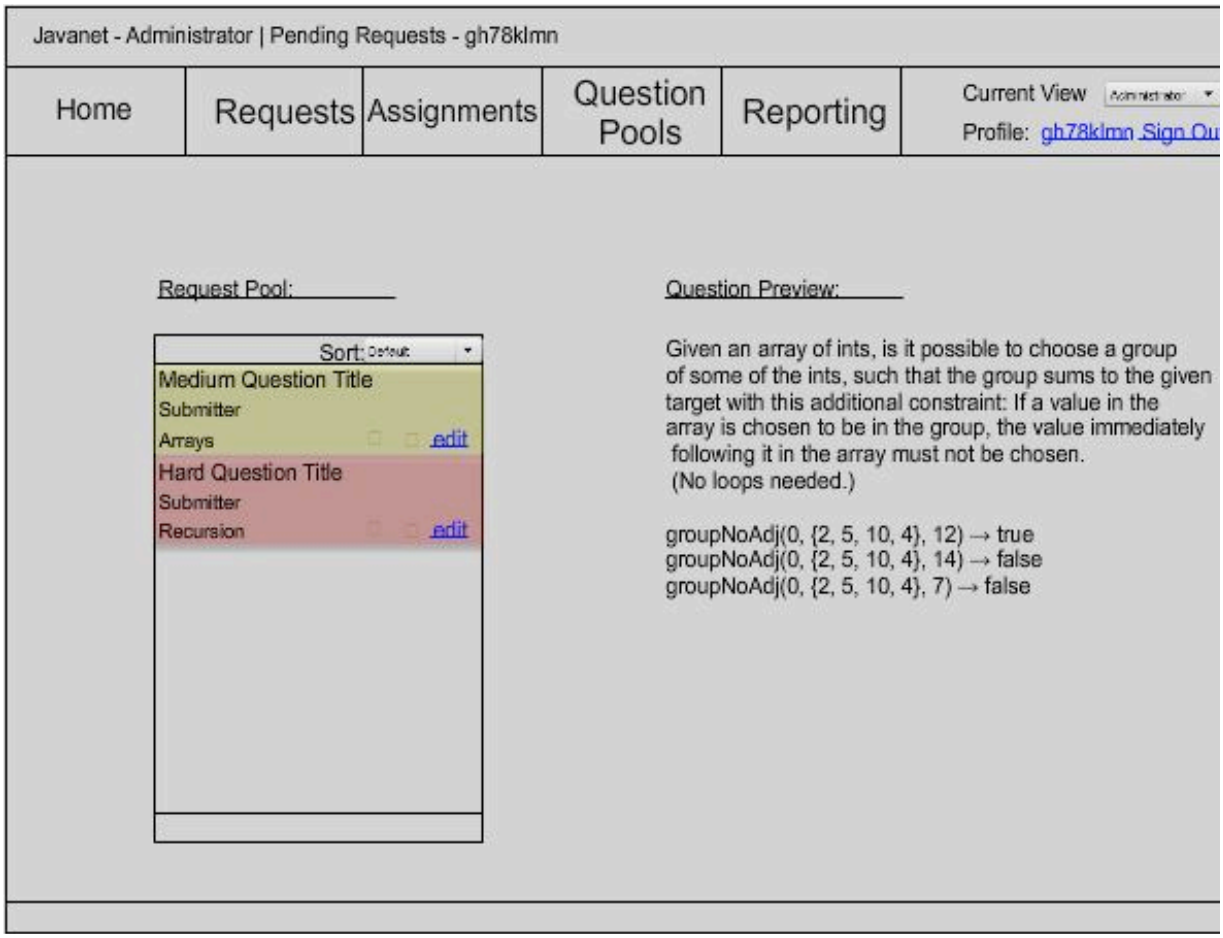
This page can be accessed by clicking the *Assignments* tab. On this page, a Course Coordinator can view the assignments for a specific course. This page will display three lists: homework, quizzes, and tests. Each list will display the due date and the section for which the assignment has been assigned.

1.2.30 Administrator - Welcome Page



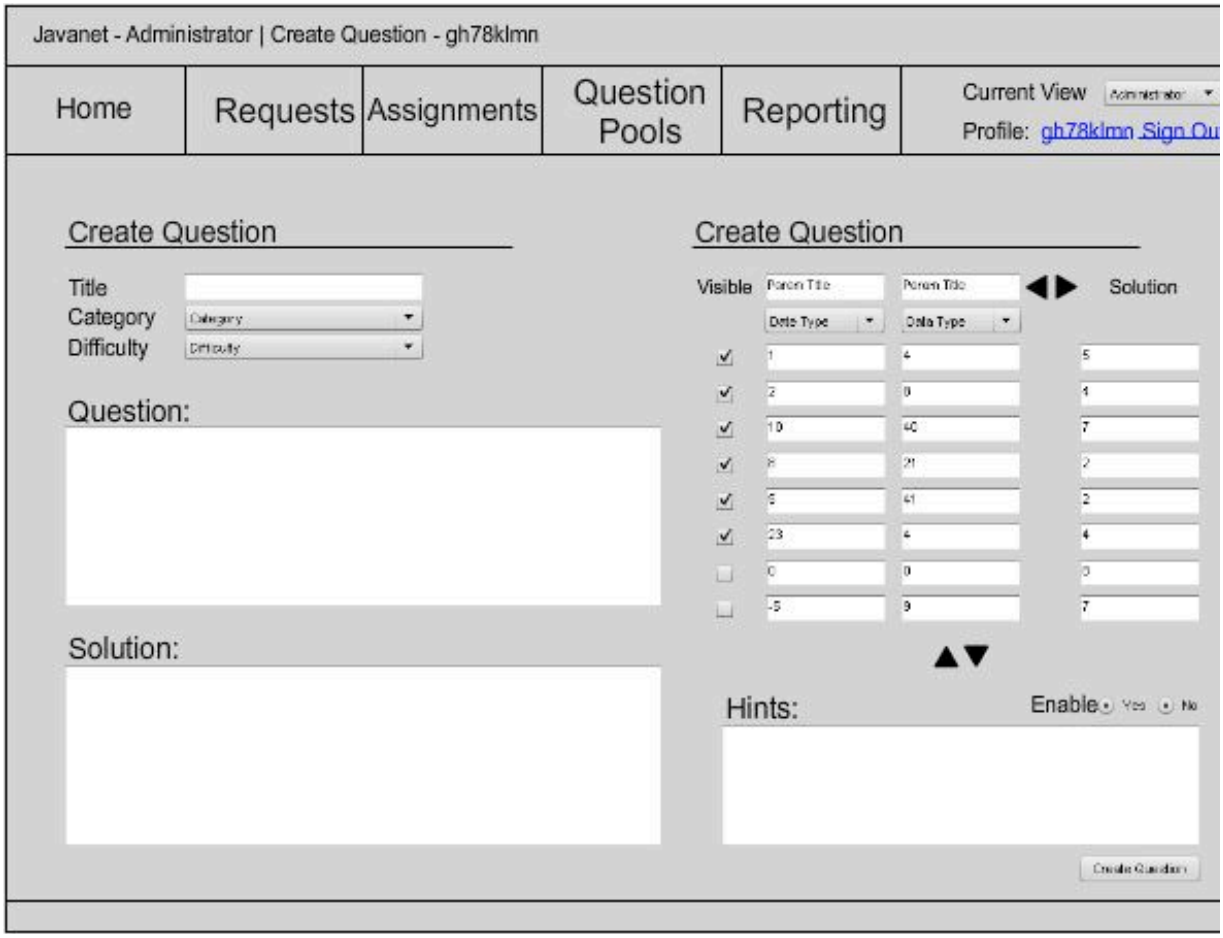
After logging in, the Administrator will be brought to this page. The Administrator will see any pending requests, the most recent announcements posted and be able to post a new announcement for the entire system of Javanet users. The Administrator may also edit or delete any announcements that they have posted.

1.2.31 Administrator - Pending Requests Page



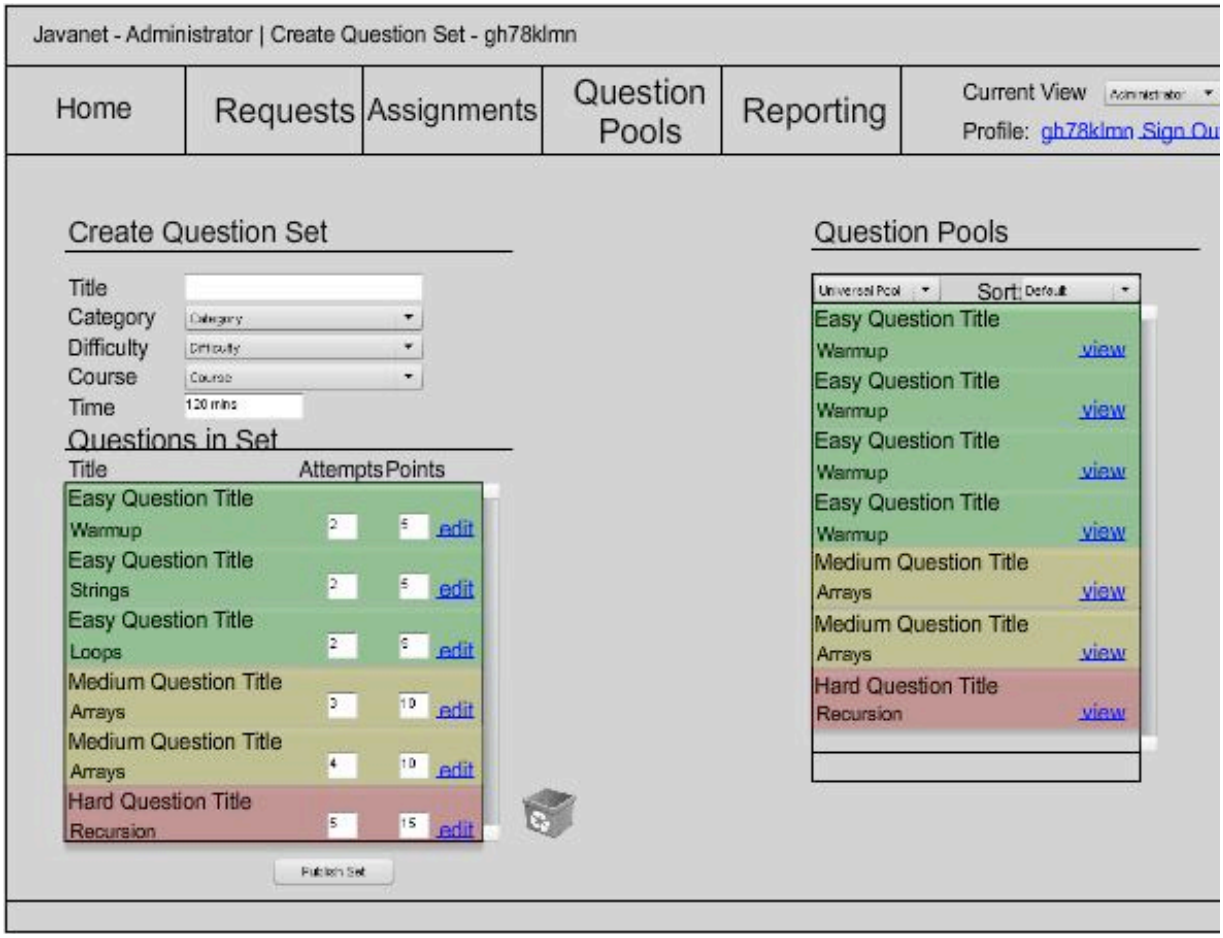
The Pending Requests Page will show the specific pending requests that the Administrator has. The Administrator can choose to approve, deny, or edit a question or question set.

1.2.32 Administrator - Create Question Page



The Administrator will see this page when creating a question. The Administrator can select a title, category, and difficulty for the question. The Administrator can then write the question and solution as well. The Administrator will also be able to add parameters for test sets and hints if desired. The arrows in the create test set section provide the functionality of creating more test sets and more parameters.

1.2.33 Administrator - Create Question Set Page



When creating a question set, the Administrator will see this page. The Administrator can create a title, select a category, select a difficulty level, and a time limit for completion. The Administrator will be able to edit each question in the set, as well as adjust the amount of attempts and, or points available. The Administrator is able to build a question set with questions from any of the pools.

1.3 Data Flow Diagrams

Please refer to the separate Data Flow Diagrams document.

1.4 Logical Data Dictionary

The Logical Data Dictionary is used to define the data flow in each diagram. It defines which diagrams the data flow is located in, the source, or the user that is sending the data flow request, and the destination, the process or data store that the data flow will either be manipulated, or saved.

Gradebook Request

Locations:

Level 0 Diagram — Destination: View Gradebook

Source: Administrator

Level 0 Diagram — Destination: View Gradebook

Source: Instructor

Level 0 Diagram — Destination: View Gradebook

Source: Course Coordinator

Level 1 View Gradebook — Destination: Select Course

Source: Administrator

Level 1 View Gradebook — Destination: Select Course

Source: Course Coordinator

Level 1 View Gradebook — Destination: Select Course

Source: Instructor

Level 1 View Gradebook — Destination: View Gradebook

Source: Select Course

Level 1 View Gradebook — Destination: View Gradebook

Source: Student

Gradebook Request Cont'd

Level 2 View Gradebook Destination: Global View

Source: Administrator

Level 2 View Gradebook Destination: View Assignments

Source: Administrator

Level 2 View Gradebook Destination: View Student

Source: Course Coordinator

Level 2 View Gradebook Destination: View Student

Source: Instructor

Level 2 View Gradebook Destination: View Course

Source: Course Coordinator

Level 2 View Gradebook Destination: View Assignments

Source: Student

Level 2 View Gradebook Destination: View Assignments

Source: Instructor

Level 2 View Gradebook Destination: View Assignments

Source: Course Coordinator

Gradebook Response

Locations:

Level 0 Diagram Destination: View Gradebook

Source: Administrator

Level 0 Diagram Destination: View Gradebook

Source: Instructor

Gradebook Response Cont'd

Level 0 Diagram	Destination: View Gradebook
	Source: Course Coordinator
Level 1 View Gradebook	Destination: View Gradebook
	Source: Administrator
Level 1 View Gradebook	Destination: View Gradebook
	Source: Course Coordinator
Level 1 View Gradebook	Destination: View Gradebook
	Source: Instructor
Level 1 View Gradebook	Destination: View Gradebook
	Source: Student
Level 2 View Gradebook	Destination: Administrator
	Source: Global View
Level 2 View Gradebook	Destination: Course Coordinator
	Source: View Student
Level 2 View Gradebook	Destination: Instructor
	Source: View Student
Level 2 View Gradebook	Destination: Course Coordinator
	Source: View Course
Level 2 View Gradebook	Destination: Administrator
	Source: View Assignment
Level 2 View Gradebook	Destination: Course Coordinator
	Source: View Assignment
Level 2 View Gradebook	Destination: Instructor
	Source: View Assignment

Gradebook Response Cont'd

Level 2 View Gradebook Destination: Student

Source: View Assignment

Filter by Student

Locations:

Level 2 View Gradebook Destination: View Student

Source: Global View

Level 2 View Gradebook Destination: View Student

Source: View Course

Filter by Course

Locations:

Level 2 View Gradebook Destination: View Course

Source: View Student

Course Request

Locations:

Level 1 View Gradebook Destination: Course List

Source: Select Course

Course Response

Locations:

Level 1 View Gradebook Destination: Select Course

Source: Course List

Account Response

Locations:

Level 0 Diagram Destination: Course Coordinator

Source: Manage Accounts

Account Response Cont'd

Level 0 Diagram Destination: Administrator

Source: Manage Accounts

Level 0 Diagram Destination: Instructor

Source: Manage Accounts

Level 2 Manage Accounts Destination: Administrator

Source: Change Course Grades

Level 2 Manage Accounts Destination: Course Coordinator

Source: Change Course Grades

Level 2 Manage Accounts Destination: Instructor

Source: Change Course Grades

Level 2 Manage Accounts Destination: Student

Source: Create User Name and Password

Account Request

Locations:

Level 0 Diagram Destination: Manage Accounts

Source: Course Coordinator

Level 0 Diagram Destination: Manage Accounts

Source: Administrator

Level 0 Diagram Destination: Manage Accounts

Source: Instructor

Level 2 Manage Accounts Destination: Change Course Grades

Source: Administrator

Level 2 Manage Accounts Destination: Change Course Grades

Source: Course Coordinator

Account Request Cont'd

Level 2 Manage Accounts Destination: Change Course Grades

Source: Instructor

Level 2 Manage Accounts Destination: Create User Name and Password

Source: Student

Feedback

Locations:

Level 0 Diagram Destination: Student

Source: Answer Problem

Level 1 Answer Problem Destination: Student

Source: Submit Answer

Problem Request

Locations:

Level 0 Diagram Destination: Answer Problem

Source: Student

Level 1 Answer Problem Destination: Select Problem Type

Source: Student

Question Request

Locations:

Level 0 Diagram Destination: Manage Question/Question Sets

Source: Administrator

Level 0 Diagram Destination: Manage Question/Question Sets

Source: Instructor

Question Request Cont'd

Level 0 Diagram Destination: Manage Question/Question Sets

Source: Course Coordinator

Level 1 Manage Question/Question Sets
Destination: Add/Remove Question/Question Set

Source: Course Coordinator

Level 1 Manage Question/Question Sets
Destination: Add/Remove Question/Question Set

Source: Administrator

Level 1 Manage Question/Question Sets
Destination: Add/Remove Question/Question Set

Source: Instructor

Question Response

Locations:

Level 0 Diagram Destination: Administrator

Source: Manage Question/Question Sets

Level 0 Diagram Destination: Instructor

Source: Manage Question/Question Sets

Level 0 Diagram Destination: Course Coordinator

Source: Manage Question/Question Sets

Level 1 Manage Question/Question Sets Destination: Course Coordinator

Source: Add/Remove Question/Question Set

Level 1 Manage Question/Question Sets Destination: Administrator

Source: Add/Remove Question/Question Set

Level 1 Manage Question/Question Sets Destination: Instructor

Source: Add/Remove Question/Question Set

Login Request

Locations:

Level 0 Diagram Destination: Login

Source: Administrator

Level 0 Diagram Destination: Login

Source: Course Coordinator

Level 0 Diagram Destination: Login

Source: Instructor

Level 0 Diagram Destination: Login

Source: Student

Level 1 Login Destination: Authenticate

Source: Student

Level 1 Login Destination: Authenticate

Source: Instructor

Level 1 Login Destination: Authenticate

Source: Course Coordinator

Level 1 Login Destination: Authenticate

Source: Administrator

Level 2 Login Destination: Sanatize Strings

Source: Student

Level 2 Login Destination: Sanatize Strings

Source: Administrator

Level 2 Login Destination: Sanatize Strings

Source: Course Coordinator

Login Request Con't

Level 2 Login Destination: Sanatize Strings

Source: Instructor

Login Response

Locations:

Level 0 Diagram Destination: Administrator

Source: Login

Level 0 Diagram Destination: Course Coordinator

Source: Login

Level 0 Diagram Destination: Instructor

Source: Login

Level 0 Diagram Destination: Student

Source: Login

Level 1 Login Destination: Student

Source: Authenticate

Level 1 Login Destination: Instructor

Source: Authenticate

Level 1 Login Destination: Course Coordinator

Source: Authenticate

Level 1 Login Destination: Administrator

Source: Authenticate

Level 2 Login Destination: Student

Source: Match Passwords

Level 2 Login Destination: Administrator

Source: Match Passwords

Login Response Con't

Level 2 Login Destination: Course Coordinator

Source: Match Passwords

Level 2 Login Destination: Instructor

Source: Match Passwords

Locate User

Locations:

Level 2 Login Destination: Users

Source: Sanatize Strings

User Information

Locations:

Level 2 Login Destination: Match Passwords

Source: Users

Create New Account

Locations:

Level 1 Manage Accounts Destination: Create Accounts

Source: Administrator

Level 1 Manage Accounts Destination: Enroll

Source: Student

Edit Account Request

Locations:

Level 1 Manage Accounts Destination: Edit Profile

Source: Administrator

Level 1 Manage Accounts Destination: Edit Profile

Source: Student

Edit Account Request Con't

Level 1 Manage Accounts Destination: Edit Profile

Source: Course Coordinator

Level 1 Manage Accounts Destination: Edit Profile

Source: Instructor

Edit Account Response

Locations:

Level 1 Manage Accounts Destination: Administrator

Source: Edit Profile

Level 1 Manage Accounts Destination: Instructor

Source: Edit Profile

Level 1 Manage Accounts Destination: Course Coordinator

Source: Edit Profile

Level 1 Manage Accounts Destination: Student

Source: Edit Profile

Account Creation Response

Locations:

Level 1 Manage Accounts Destination: Administrator

Source: Create Accounts

Accessibility Request

Locations:

Level 2 Manage Accounts Destination: Create/Disable Accounts

Source: Administrator

Accessibility Response

Locations:

Level 2 Manage Accounts Destination: Administrator

Source: Create/Disable Accounts

Attempt Reset

Locations:

Level 2 Manage Accounts Destination: Reset Password

Source: Administrator

Level 2 Manage Accounts Destination: Reset Password

Source: Course Coordinator

Level 2 Manage Accounts Destination: Reset Password

Source: Instructor

Level 2 Manage Accounts Destination: Reset Password

Source: Student

Reset Feedback

Locations:

Level 2 Manage Accounts Destination: Administrator

Source: Reset Password

Level 2 Manage Accounts Destination: Course Coordinator

Source: Reset Password

Level 2 Manage Accounts Destination: Instructor

Source: Reset Password

Level 2 Manage Accounts Destination: Student

Source: Reset Password

Test Set Request

Locations:

Level 1 Manage Question/Question Sets Destination: Add/Remove Test Set

Source: Administrator

Test Set Request Cont'd

Level 1 Manage Question/Question Sets Destination: Add/Remove Test Set

Source: Instructor

Level 1 Manage Question/Question Sets Destination: Add/Remove Test Set

Source: Course Coordinator

Test Set Response

Locations:

Level 1 Manage Question/Question Sets Destination: Administrator

Source: Add/Remove Test Set

Level 1 Manage Question/Question Sets Destination: Instructor

Source: Add/Remove Test Set

Level 1 Manage Question/Question Sets Destination: Course Coordinator

Source: Add/Remove Test Set

Pool Modification Request

Locations:

Level 1 Manage Question/Question Sets

Destination: Add/Remove Question/Question Set

Source: Question Pool

Pool Modification Response

Locations:

Level 1 Manage Question/Question Sets Destination: Question Pool

Source: Add/Remove Question/Question Set

Question Type

Locations:

Level 1 Answer Problem — Destination: Question Pool

Source: Select Problem Type

Selected Question

Locations:

Level 1 Answer Problem — Destination: Select Problem Type

Source: Question Pool

Submitted Code

Locations:

Level 1 Answer Problem — Destination: Submit Answer

Source: Select Problem Type

Source

Locations:

Level 1 Answer Problem — Destination: Save Compile and Run

Source: Java Compiler

Graded Solution

Locations:

Level 1 Answer Problem — Destination: Java Compiler

Source: Save Compile and Run

Answer Submission

Locations:

Level 1 Answer Problem — Destination: Java Compiler

Source: Submit Answer

Answer Results

Locations:

Level 1 Answer Problem Destination: Submit Answer

Source: Java Compiler

1.5 Logical Data Stores

1.5.1 Data Stores Key

The following abbreviations are used in the Data Stores:
PK: Primary Key
NN: Not Null
U: Unique

1.5.2 Data Stores

Table	Name	Type	Restrictions	Description
Users	uID	int	PK	unique id number of the user - auto-generated
Users	uName	varchar(32)	NN, U	the user's username
Users	uPassword	varchar(44)	NN	the user's password - encrypted
Users	uFname	varchar(32)	NN	the user's first name
Users	uLname	varchar(32)	NN	the user's last name
Users	uEmail	varchar(64)	NN, U	a user's email address
Users	uSecurityQuestion	varchar(128)	NN	question to be answered by the user before resetting a password
Users	uSecurityAnswer	varchar(128)	NN	user answer to the security question for identity validation
Table	**Name**	**Type**	**Restrictions**	**Description**

Users	uType	tinyint	NN	type of the user - 1 for Admin, 2 for Course Coordinator, 3 for Instructor, 4 for Student

Enrollment	userID, courseID	int, int	PK	user id and the course that they are enrolled in

Announcements	aID	int	PK	unique id number for the announcement - auto-generated
Announcements	aTitle	varchar(128)		title of the announcement
Announcements	aText	text	NN	text of the announcement published to all welcome pages of users associated with the creator
Announcements	aCreatorID	int	NN	user id of the creator of the announcement
Announcements	aCreationDate	date	NN	date an announcement was created
Announcements	aExpireDate	date		date an announcement expires
Announcements	aVisible	bool		determines if an announcement is visible or not (hiding/showing to users), default will be true

Courses	cID	int	PK	unique id number for a course, used to register for a specific course and section
Courses	cYear	int	NN	the year the course was taught
Courses	cSemester	varchar(32)	NN	the semester the course was taught, eg. FALL
Courses	cSection	varchar(8)	NN	the section of the course
Courses	cName	varchar(16)	NN	the course name and number, eg. CSIS120

Questions	qID	int	PK	unique id number for the question
Questions	qTitle	varchar(128)	NN	title of the question
Questions	qCategory	varchar(32)	NN	category of the question, eg. Arrays, Loops, Recursion
Questions	qDifficulty	varchar(16)	NN	name of the method, used to generate the method signature
Questions	qText	text	NN	description of the problem for the student to solve
Questions	qHints	text		any hints, including urls, that the instructor has given
Table	**Name**	**Type**	**Restrictions**	**Description**
Questions	qShowHint	bool		determines if the hints will be displayed to the student, the default will be false

Questions	qReturnType	varchar(32)	NN	the expected data type passed back by the function
Questions	qCreatorID	int	NN	user id of the user who created the question
Questions	qMethodName	varchar(64)	NN	name of the method, used to generate the method signature
Questions	qSetID	int		id of the question set that the question belongs to

QuestionSet	qsID	int	PK	unique id number for the question set
QuestionSet	qsCategory	varchar(32)	NN	category of the question set, eg. Arrays, Loops, Recursion
QuestionSet	qsDifficulty	varchar(16)	NN	difficulty of the question set, eg. Easy, Medium, Hard
QuestionSet	qsTitle	varchar(128)		title of the question set, eg. HW1, Test2
QuestionSet	qsCreatorID	int	NN	user id of the user who created the question set
QuestionSet	qsCourseID	int	NN	id of the course that the question set is associated with

QuestionParams	qpParentQuestion, qpOrderNum	int, int	PK	the question that the parameter is associated with and the order in which it will be passed into the function
QuestionParams	qpType	varchar(32)	NN	the data type of the parameter for the question, used to generate the method signature
QuestionParams	qpName	varchar(32)	NN	the name of the parameter for the question, used to generate the method signature

QuestionTestCases	qtcParentQuestion, qtcOrderNum	int, int	PK	the question that the test case is associated with and the order in which it will appear in the listing of test cases

Table	Name	Type	Restrictions	Description
QuestionTestCases	qtcValue	varchar(128)	NN	the value of the test case defined by the question creator
QuestionTestCases	qtcExpectedReturnVal	varchar(128)	NN	the value that is expected to be returned after the code is compiled and run
QuestionTestCases	qtcVisibility	bool		determines whether the test case is visible to the student user, default will

				be true

PublishedQuestions	pqID	int	PK	unique id number for the published question
PublishedQuestions	pqTitle	varchar(128)		title of the published question, eg. HW1, Test2
PublishedQuestions	pqCategory	varchar(32)	NN	category of the published question, eg. Arrays, Loops, Recursion
PublishedQuestions	pqDifficulty	varchar(16)	NN	difficulty of the published question, eg. Easy, Medium, Hard
PublishedQuestions	pqText	text	NN	description of the published question for the student to solve
PublishedQuestions	pqHint	text		any hints, including urls, that the instructor has given for the published question
PublishedQuestions	pqShowHint	bool		determines if the hints will be displayed to the student for the published question, the default will be false
PublishedQuestions	pqReturnType	varchar(32)	NN	the expected data type passed back by the function of the published question
PublishedQuestions	pqCreatorID	int	NN	user id of the user who created the published question
PublishedQuestions	pqMethodName	varchar(64)	NN	name of the method, used to generate the method signature for the published question
PublishedQuestions	pqPointsPossible	int		the maximum number of points possible for the published question
PublishedQuestions	pqNumAttempts	int		the maximum number of attempts allowed for the published question
PublishedQuestions	pqTimeLimitSeconds	int		the maximum number of seconds allowed for the published question
Table	**Name**	**Type**	**Restrictions**	**Description**
PublishedQuestions	pqDisable	bool		determines if the published question is visible and able to be used for a question set, the default will be false
PublishedQuestions	pqsID	int	NN	id of the question set that the published question belongs to
PublishedQuestions	pqSolution	text		the instructor solution to the published question, if

				provided by the instructor
PublishedQuestions	pqOwnerID	int	NN	user id of the user who owns the published question
PublishedQuestions	pqType	varchar(32)	NN	the type of the published question, eg. HW, Test, Practice
PublishedQuestions	pqExpirationDate	date		the date that the published question will expire
PublishedQuestions	pqCreationDate	date	NN	the date that the published question was created

PublishedQuestionSets	pqsID	int	PK	unique id number for the published question set
PublishedQuestionSets	pqsCategory	varchar(32)	NN	category of the published question set, eg. Arrays, Loops, Recursion
PublishedQuestionSets	pqsDifficulty	varchar(16)	NN	difficulty of the published question set, eg. Easy, Medium, Hard
PublishedQuestionSets	pqsTitle	varchar(128)		title of the published question set, eg. HW1, Test2
PublishedQuestionSets	pqsCreatorID	int	NN	user id of the user who created the published question set
PublishedQuestionSets	pqsOwnerID	int	NN	user id of the user who owns the published question set
PublishedQuestionSets	pqsType	varchar(32)	NN	the type of the published question set, eg. HW, Test, Practice
PublishedQuestionSets	pqsDueDate	date		the date that the published question set is due, if applicable
PublishedQuestionSets	pqsPointsPossible	int		the total number of points possible for a student to receive on the published question set, if applicable
PublishedQuestionSets	pqsNumAttempts	int		the total number of attempts allowed to a student for the published question set, if applicable
Table	**Name**	**Type**	**Restrictions**	**Description**
PublishedQuestionSets	pqsTimeLimit	int		the time, in seconds, allotted to a student to complete the published question set, if applicable
PublishedQuestionSets	pqsVisiblity	bool		determines if the published question set is active/visible to student users, default is true
PublishedQuestionSets	pqsExpirationDate	date		the date that the published question set will expire and must be completed by
PublishedQuestionSets	pqsCreationDate	date	NN	the date that the published question set was created

PublishedQuestionSets	pqsWeight	int		the weight of a particular published question set (assignment), for grading purposes, if applicable
PublishedQuestionSets	pqsCourseID	int	NN	id of the course that the question set is associated with

StudentSolutions	ssuID, sspqID, sspqsID	int, int, int	PK	the student user id and the specific published question and published question set that the student code is associated with
StudentSolutions	ssSourceCode	text		the source code submitted by a specific student for a specific question in a question set
StudentSolutions	ssPointsReceived	int		the total number of points assigned to the student solution by the automatic grading system

Requests	rID	int	PK	unique id number for the request
Requests	rRequesterID	int	NN	user id of the person requesting publication
Requests	rqsID	int	NN	id of the question set to be published
Requests	rApprovalStatus	bool		status of the request, eg. 0 for false (denied), 1 for true (approved)
Requests	rApproverID	int		user id of the course coordinator or administrator who has approved or denied the request

Table	Name	Type	Restrictions	Description
PublishedQuestionParams	pqpParentQuestion, pqpOrderNum	int, int	PK	the published question that the parameter is associated with and the order in which it will be passed into the function
PublishedQuestionParams	pqpType	varchar(32)	NN	the data type of the parameter for the published question, used to generate the method signature
PublishedQuestionParams	pqpName	varchar(32)	NN	the name of the parameter for the published question, used to generate the method signature

PublishedQuestionTestCases	pqtcParentQuestion, pqtcOrderNum	int, int	PK	the published question that the test case is associated with and the order in which it will

				appear in the listing of test cases
PublishedQuestionTestCases	pqtcValue	varchar(128)	NN	the value of the test case for the published question, defined by the question creator
PublishedQuestionTestCases	pqtcExpectedReturnVal	varchar(128)	NN	the value that is expected to be returned after the solution for the published question is compiled and run
PublishedQuestionTestCases	pqtcVisibility	bool		determines whether the test case is visible to the student user for the published question, default will be true

1.6 Logical Format of Data Files & Databases

All database tables and relationships are being created by our team specifically for this project. None of our database information is already in existence or being derived from an external source such as our client. Further information regarding our database design can be found in the ***Logical Data Stores*** section of this document.

2.1 Development & Production 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.

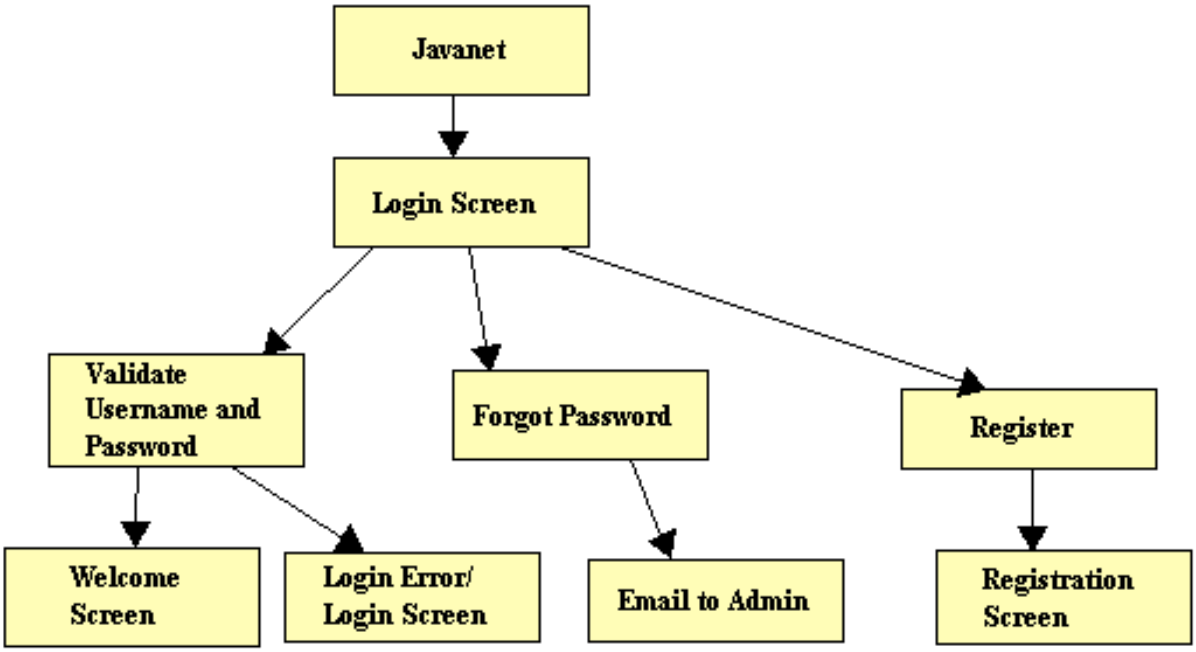
Javanet will be accessible via three of the main internet browsers; Internet Explorer 7.0 and 8.0, the latest version of Firefox 2 and 3, and Safari 2.* or better.

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.

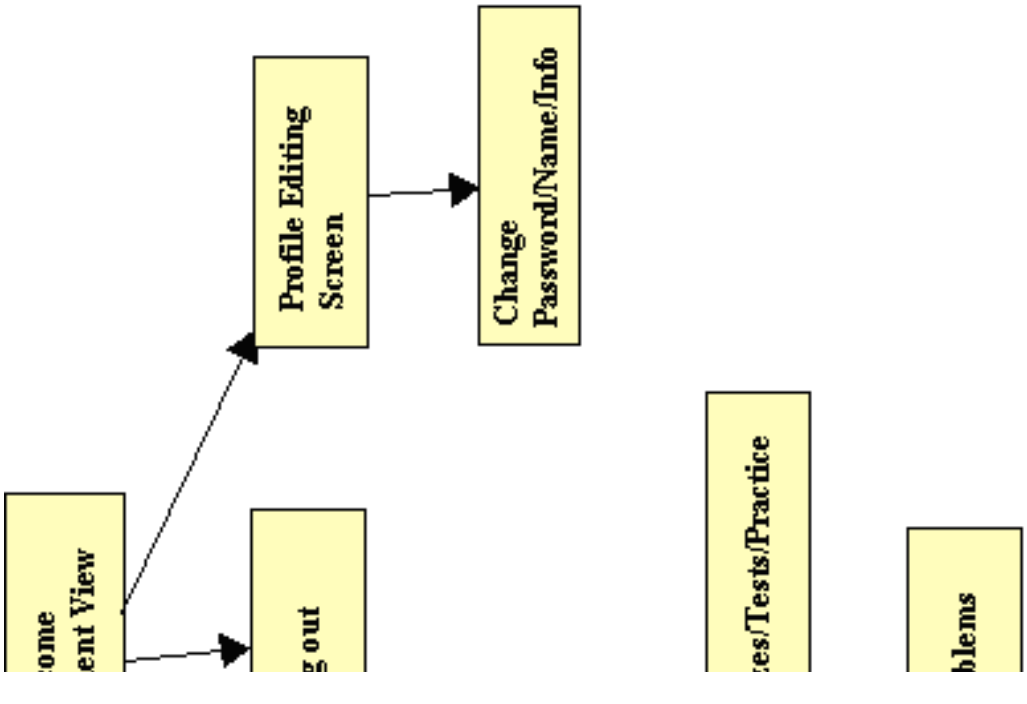
2.2 Structure Diagrams

A structure diagram is the hierarchical representation of how all users, processes, and data flow will take place.



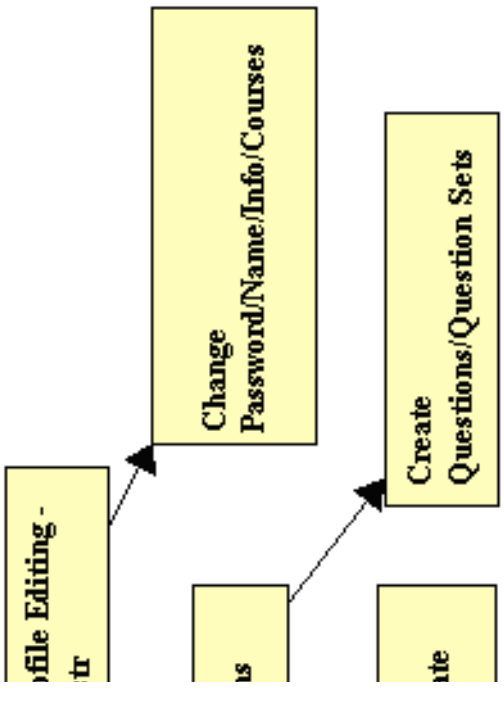
A structure diagram for the Javanet system.

2.2.1 Student



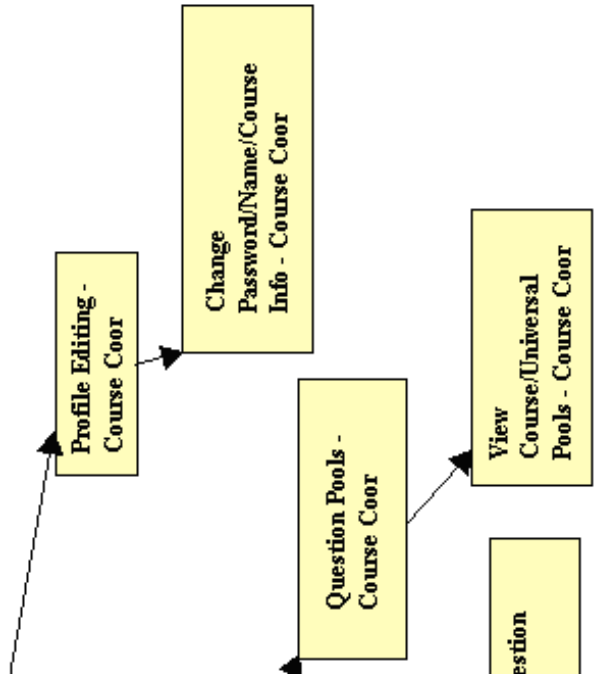
A structure diagram for the Student user.

2.2.2 Instructor



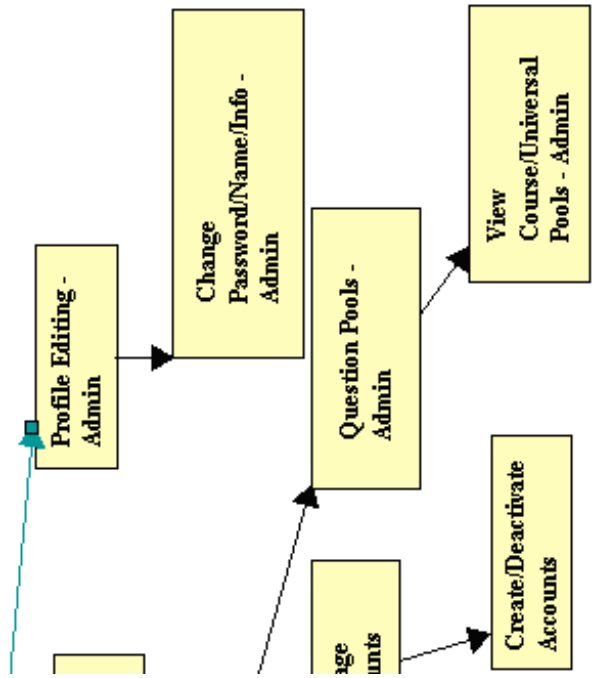
A structure diagram for the Instructor user.

2.2.3 Course Coordinator



A structure diagram for the Course Coordinator user.

2.2.4 Administrator



A structure diagram for the Administrator user.

2.3 Use Case Narratives

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.

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

Instructor cont'd

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.

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 viewable to all Instructors. The Course Coordinator can switch into Instructor or Student view mode. The Course Coordinator will be able to create questions or question sets to go into the Course Pool. The Course Coordinator 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, 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.

Administrator

The Administrator user will have the ability to login to and log out of the application through a web interface using 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. The Administrator will have the ability to deactivate user accounts, and manage and reset passwords. The Administrator 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 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 be able to change grades. The Administrator will be able to archive student information from the database.

2.4 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:

Student:

- Will be able to self enroll by creating a username (Siena email based) and password.
- 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 upon clicking username link.
- Will be able to enroll into their appropriate course.
- Will be able to select a "Practice" tab.
    - Will be able to select practice question sets under given categories.
    - A check mark will be rewarded once the questions in a question set are completed.
    - A smile face will be rewarded once a category is completed.
- Will be able to select an "Assignments" tab.
    - Will be able to choose from chronologically ordered assigned homework.
    - Will be able to choose from chronologically ordered assigned quizzes.
    - Will be able to choose from chronologically ordered assigned tests.
- Will be able to write, compile (or save) and submit solutions to questions.
    - If hints for questions have been enabled, the option to view those hints is available.
    - Test cases and their solutions can be seen on the right-hand side of the screen.
    - A read only version of Students' question solutions will be available once they have been submitted and graded.
- Will be able to view their grades via a "Gradebook" tab.
    - Links for each assignment will redirect students to a summary page.
- 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 select what course section page to view on login.
- Will be able to view and edit various account information upon clicking username link.
- Will be able to post a new announcement for students of that course to see upon login.
- Will be able to select "Courses" tab to switch into another course.
- Will be able to select an "Assignments" tab.
    - Will be able to choose from chronologically ordered assigned homework redirecting to the gradebook.

    - Will be able to choose from chronologically ordered assigned quizzes redirecting to the gradebook.
    - Will be able to choose from chronologically ordered assigned tests redirecting to the gradebook.
- Will be able to select a "Question Pool" tab.
    - A search function will be available to filter through the Universal, Course, and Private pools.
    - Will be able to drag and drop questions from Universal and Course pools to Private pool.
    - Will be able to drag and drop questions from Private pool to Universal or Course pools to request publication from Administrator.
    - Links under private pool will allow creation of a new question and or question set.
- Will have the following options when creating a question set:
    - Creating a title.
    - Selecting a category for the question set.
    - Selecting a difficulty level for a question set.
    - Creating a time limit.
    - Editing each question in the question set or dragging into trash bin for unwanted questions.
    - Adjusting the number of attempts allowed and points for each question.
- Will have the following options when creating a question:
    - Creating a title.
    - Selection a category for the question set.
    - Selecting a difficulty level for a question set.
    - Text box to enter question.
    - Text box to enter solution.
    - Entering and selecting parameter types and their solutions.
    - Enabling hints.
- Will be able to select a "Gradebook" tab.
    - Sub tabs will include: "Default", "Grades By Student", and "Grades By Assignment".
- Will be able to override a Student's grade when the automatically generated grade is inappropriate.

Instructor User cont'd

- 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.

Course Coordinator User:

- Will be able to log into the system.

- Will be able to switch into Instructor and 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.

Administrative User:

- Will be able to log into system.
- Will be able to create Instructor and Course Coordinator accounts.
- Will be able to change current view mode to Course Coordinator, Instructor, or Student.
- Will be able to edit Course Coordinator, Instructor, and Students account information while in their own view mode.
- Will be able to select a "Requests" tab.
    - Will be able to edit, approve, or deny questions pending publication.
    - A question preview will be displayed on the right-hand side.
- Will be able to select a "Question Pool" tab.
    - A search function will be available to filter through the Universal, Course, and Private pools.
    - Will be able to drag and drop questions from Universal and Course pools to Private pool.

    - Will be able to drag and drop questions from Private pool to Universal or Course pools to request publication from Administrator.
    - Links under private pool will allow creation of a new question and or question set.
- Will have the following options when creating a question set:
    - Creating a title.
    - Selecting a category for the question set.
    - Selecting a difficulty level for a question set.
    - Creating a time limit.
    - Editing each question in the question set or dragging into trash bin for unwanted questions.
    - Adjusting the number of attempts allowed and points for each question.
- Will have the following options when creating a question:
    - Creating a title.
    - Selection a category for the question set.
    - Selecting a difficulty level for a question set.
    - Text box to enter question.
    - Text box to enter solution.
    - Entering and selecting parameter types and their solutions.
    - Enabling hints.
- Will be able to create questions and question sets to go into the Universal pool.
- Will be able to change grades, if needed.
    - Switching view modes to either a Course Coordinator or Instructor.
- 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.

2.5 Non-Functional Requirements Inventory

- The system should also be aesthetically pleasing.
- The system should have a consistent user interface
- The system should have a reasonable compilation time
- The system should adhere to Shneiderman's 8 Golden Rules (located in the Glossary Section 5.2)

3.1 Packaging Specifications

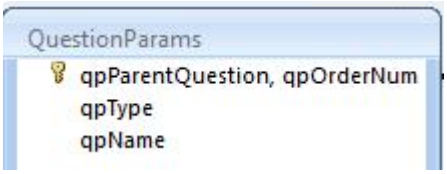
The users will be supplied with the website pointing to the Javanet welcome screen. It will be accessible through any of the three main Internet browsers, Internet Explorer 7+, Mozilla Firefox versions 2.0 and higher and Macintosh Safari 2.0 and higher. All the

necessary software, files, database tables, and features of Javanet will be preloaded on Siena Computer Science web server, Oraserv.

3.2 Physical Data Structure Specifications

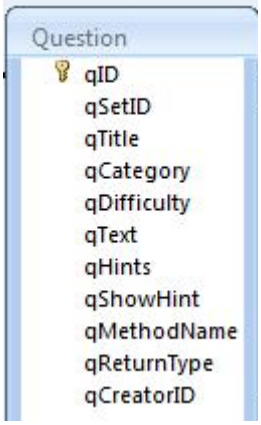
QuestionParams

This table holds the parameters of the signature for a specific question that has been created.



Question

This table holds the information pertaining to a specific question that has been created.

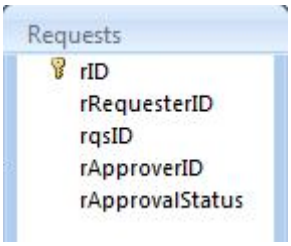


Announcements

This table holds information pertaining to a specific announcement that has been posted by a user.

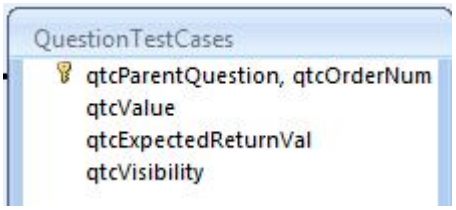
Requests

This table holds question pool request information.



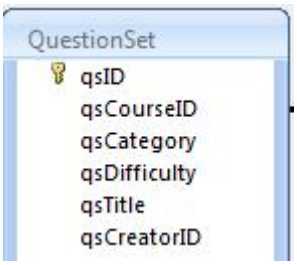
QuestionTestCases

This table holds information about question solution test cases and their respective parent question.



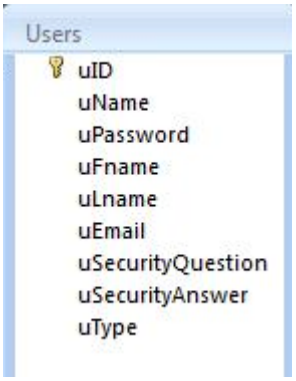
QuestionSet

This table holds information regarding a specific question set that has been created.



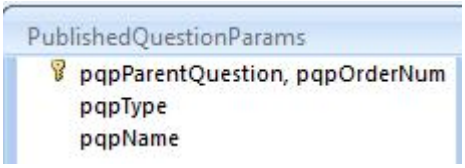
Users

This table holds information about a user that has registered with the system.



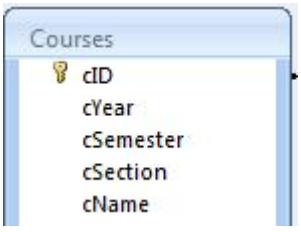
PublishedQuestionParams

This table holds information about a published question's parameters.



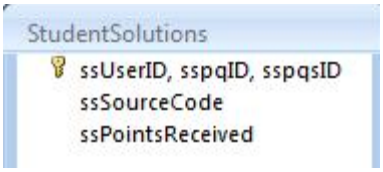
Courses

This table holds information about a specific course.



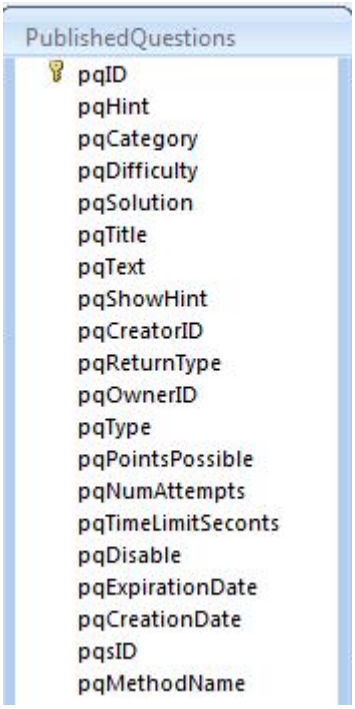
StudentSolutions

This table holds information pertaining to a specific student's solution to a specific question.



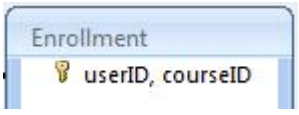
PublishedQuestions

This table holds information about published questions.



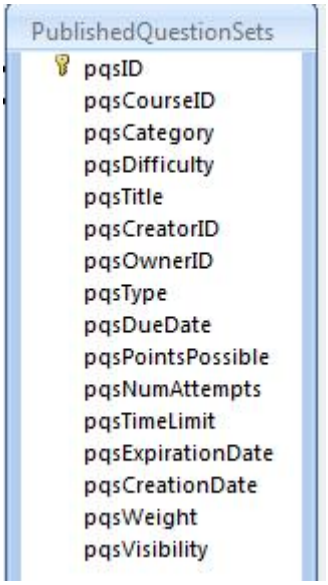
Enrollment

This table holds information about a user registered with a specific course.



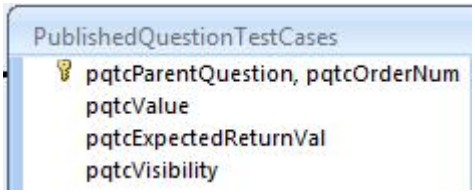
PublishedQuestionSets

This table holds information about a specific published question set.



PublishedQuestionTestCases

This table holds information about a specific published question's solution test cases.



4.1 Testing Overview

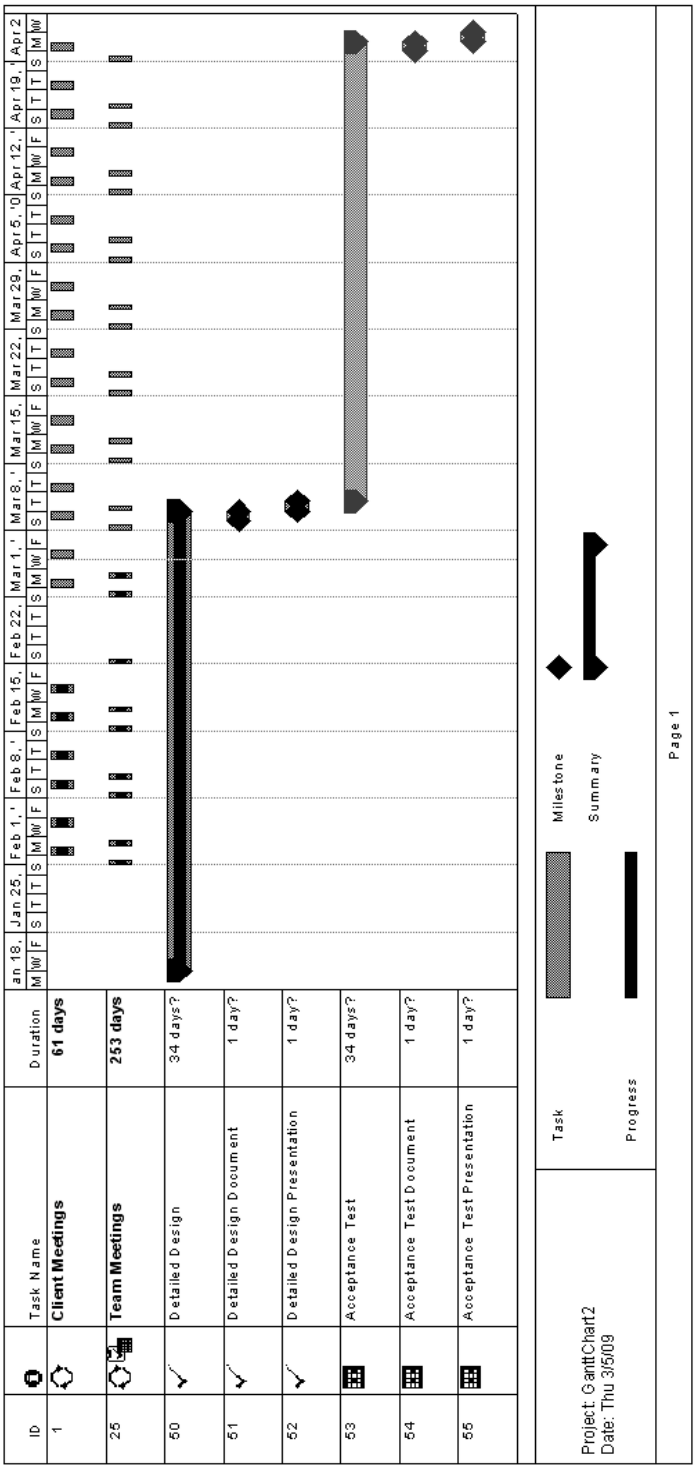
Javanet will undergo extensive unit testing during the Acceptance Test phase of our project. This testing will ensure Javanet meets all functional and non-functional requirements as stated in the functional requirements inventory section of this document, Section 2.3. The acceptance test will be presented to demonstrate our software's compliance with the needs of the clients. The Unit Test Catalog and Unit Tests are provided in the accompanying Test Plan Document.

4.2 Test Plan

Please refer to the separate Test Plan document.

Appendix

5.1 Enlarged Timeline (Gantt Chart)



5.2 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. The most common application of CSS is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document, include 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 Hypertext Preprocessor 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 that they have created/uploaded to their private pool, which Students and other Instructors do not have access to.

Shneiderman's "Eight Golden Rules of Interface Design"

These rules were obtained from the text Designing the User Interface by Ben Shneiderman. Shneiderman proposed this collection of principles that are derived heuristically from experience and applicable in most interactive systems after being properly refined, extended, and interpreted.
To improve the usability of an application it is important to have a well designed interface.

Shneiderman's "Eight Golden Rules of Interface Design" are a guide to good interaction design:

1 Strive for consistency.
Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.

2 Enable frequent users to use shortcuts.
As the frequency of use increases, so do the user's desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.

3 Offer informative feedback.
For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.

4 Design dialog to yield closure.
Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans and options from their minds, and an indication that the way is clear to prepare for the next group of actions.

5 Offer simple error handling.
As much as possible, design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.

6 Permit easy reversal of actions.
This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.

7 Support internal locus of control.
Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.

8 Reduce short-term memory load.
The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

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

The Universal Pool is 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 they Universal Pool Administrator has granted permission.

Workstation

A workstation, or *engineering workstation*, is a high-end microcomputer designed for technical or scientific applications. Workstations are intended to be used by primarily 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 *Xtensible Hypertext Markup Language*, or XHTML, is a markup language that has the same depth of expression as HTML, but also conforms to XML syntax.