

SLICE

Software **L**inked **I**nteractive **C**ompetitive **E**nvironment
Aceptance **T**est
Powerpoint **P**resentation

Requested by:

Dr. Darren Lim
Associate Professor
Siena College
Loudonville, NY

Prepared by:

P=NP*solutions*

Renee Solheim – Team Leader
Anthony Parente – Front-end Developer
Matthew Ferritto – Database/Webmaster
Zachary Fitzsimmons – Lead Developer

April 26th, 2012

Contents

- 1 SLICE Acceptance Test** **1**
- 1.1 Product Overview 1
- 1.2 User Case Narratives 1
 - 1.2.1 Participant User Case Narrative 1
 - 1.2.2 Team Advisor User Case Narrative 2
 - 1.2.3 Judges User Case Narrative 2
 - 1.2.4 Administrative User Case Narrative 2
- 1.3 Functional Requirements Inventory/Test Results 3
 - 1.3.1 Administrative 3
 - 1.3.2 Team Advisor 4
 - 1.3.3 Judge 4
 - 1.3.4 Scoreboard Judge 5
 - 1.3.5 Participant 5
 - 1.3.6 SLICE 6
- 1.4 Non-Functional Requirements/Test Results 6
- 1.5 Future Enhancements 6
- 1.6 SLICE ER Diagram 7
- 1.7 Logical Data Dictionary Description 9
- 1.8 Prototype Screens 10
 - 1.8.1 General View 10
 - 1.8.2 Administrative User Views 11
 - 1.8.3 Judge User Views 16
 - 1.8.4 Participant User Views 20
- 1.9 Test Plan 23
 - 1.9.1 Testing Approach 23
- 1.10 Development and Production Environment 24

- Appendix** **25**

- A Glossary of Terms and Gantt Chart** **26**

- B Data Dictionary** **29**

C	Test Plan	32
C.1	Testing Approach	32
C.2	Testing Plan Identifier	33
C.3	Test Results	33
C.3.1	Test Items and Functional Requirements Inventory	33
C.3.2	Administrative	34
C.3.3	Team Advisor	35
C.3.4	Judge	35
C.3.5	Scoreboard Judge	36
C.3.6	Participant	36
C.3.7	SLICE	37
C.3.8	Non-Functional Requirements	37
C.4	Integration Testing	38
C.5	Exception Handling To Test	38
C.6	Acceptance Test - Acceptance Criteria	38
D	Unit Test	40
D.1	Unit Tests	40
D.1.1	Unit Test Cases	41

Chapter 1

SLICE Acceptance Test

1.1 Product Overview

Competitive programming contests are held at all different levels and scales, with a few teams to up to many participants and from single network sessions to across the world through the Internet. Currently Dr. Darren Lim holds a programming contest at Siena College for local area high school students. In the Spring of 2013, Siena College will be holding the Consortium for Computer Science in Small Colleges Northeast (CCSCNE), which includes a programming contest. The current software is not user friendly and consists of several different parts. The goal of our software, SLICE, is to allow fast, both manual and automatic judging during a competitive programming contest and to integrate problem submission, judging, and scoreboard of a contest into a single application.

1.2 User Case Narratives

1.2.1 Participant User Case Narrative

Participants will each be able to log into SLICE with a provided username and password generated by the Administrative User and provided in a hardcopy sealed envelope. After a Participant logs in, the Participant will be able to view the contest scoreboard, submit problem-solutions or submit clarification-requests. For a problem-solution submission, a Participant will select one of the contest's problems as the associated problem and one of a list of programming languages for the language. For clarification-requests a Participant can make a general request or associate it with a problem. Participants receive a response from a Judge concerning the submission, stating one of the following: Correct Solution, Compiler Error, Running Time Error, Wrong Answer or Time-Limit Exceeded. Participants will also be able to receive both individual responses to clarification-requests from any Judge as well as publically broadcasted messages from any Judge or the Administrative User.

1.2.2 Team Advisor User Case Narrative

Team Advisors will be able to log into SLICE to view the problems and scoreboard. Team Advisors accounts will be created by the Administrative User in the same way that Participant Users are created. A Team Advisor's username and password to access SLICE and is provided to each Team Advisor in a hardcopy sealed envelope. The username and password for the Team Advisor account will be generated by the Administrative User. In addition to viewing the problems and scoreboard, Team Advisors will receive all publically broadcasted contest clarifications.

1.2.3 Judges User Case Narrative

Judges will each have the ability to log onto SLICE with a username and password generated by the Administrative User and provided to each Judge in a hardcopy sealed envelope. Once logged in, a Judge will be able to view Participant submitted solutions that the system will compile. If a submission compiles, a Judge will then run the submitted programs using test input provided in SLICE that corresponds to the associated question. A Judge then submits a response to the Participant stating one of the following: Correct Solution, Compiler Error, Running Time Error, Wrong Answer or Time-Limit Exceeded. If the submission was a correct solution to the corresponding problem, a Judge with extra permissions in SLICE to update the scoreboard, if there are errors present. A Judge will be able to either respond (in general or concerning a specific problem) to a specific participant or all Participants and Team Advisors by using a messaging prompt.

1.2.4 Administrative User Case Narrative

The Administrative User will be able to log into SLICE with a unique username and password. Once logged in, the Administrative User will be able to create and manage the Judge, Participant, and Team Advisor accounts. The permissions for one Judge to have Scoreboard Judge ability is also managed by the Administrative User. The Administrative user can view the scoreboard, but only the Scoreboard Judge can edit the scoreboard. The usernames and passwords for each individual account is put into a hardcopy sealed envelope and distributed to the associated Judges, Participant or Team advisors. The Administrative User will be able to send broadcast messages to all users, or any subset i.e. all Judges, the Scoring Judge, all Participants or all Team Advisors. The Administrative User will manage the set of problems, the test input, and correct output for each contest problem. The Administrative User will also be in charge of the length of the test contest, contest start and stop times and the allowed programming languages.

1.3 Functional Requirements Inventory/Test Results

For a general functional requirement, SLICE will be able to run on all popular browsers including, Internet Explorer, Mozilla Firefox, Google Chrome and Safari.

Below is a list of functional requirements for each human user that interacts with the system. One major functional requirement is that SLICE is scalable to up to 30 Participant users.

1.3.1 Administrative

- Y/N Will be able to log into SLICE with a unique username and password
- Y/N Will be able to create the Judge accounts
- Y/N Will be able to create the Participant accounts
- Y/N Will be able to create the Team Advisor accounts
- Y/N Will be able to give Scoreboard Judge abilities to one judge
- Y/N Will be able to send broadcast messages to all users
- Y/N Will be able to send broadcast messages to any subset of users
- Y/N Will be able to manage the set of problems
- Y/N Will be able to manage test input
- Y/N Will be able to manage correct output for each contest problem
- Y/N Will be able to designate the length of the contest
- Y/N Will be able to set the contest start time
- Y/N Will be able to set the contest stop time
- Y/N Will be able to set the practice time for the contest
- Y/N Will be able to restrict the view of the scoreboard at a set time towards the end of the contest
- Y/N Will be able to choose the list of allowed programming languages
- Y/N Will be able to log out of SLICE at any point

1.3.2 Team Advisor

- Y/N Will be able to log into SLICE with the Advisor user name and password
- Y/N Will be able to view each individual problem
- Y/N Will be able to view the scoreboard while active
- Y/N Will be able to receive all publically broadcasted contest clarifications
- Y/N Will be able to log out of SLICE at any point

1.3.3 Judge

- Y/N Will be able to log onto SLICE with a given user name and password
- Y/N Will be able to view Participant submitted solutions
- Y/N Will be able to run the submitted solution using the test input provided in SLICE that corresponds to the associated question
- Y/N Will be able to submit a response to the team stating either "Correct Solution, Compiler Error, Running Time Error, Wrong Answer, OR Time-Limit Exceeded"
- Y/N Will be able to log out of SLICE at any point

1.3.4 Scoreboard Judge

Y/N Will be able to do everything the Judge can do

Y/N Will be able to edit the scoreboard

Y/N Will be able to log out of SLICE at any point

1.3.5 Participant

Y/N Will be able to log into SLICE with a given user name and password

Y/N Will be able to view the contest scoreboard while active

Y/N Will be able to submit problem-solutions

Y/N Will be able to submit clarification-requests

Y/N Will be able to make a general request

Y/N Will be able to associate a problem with request

Y/N Will be able to select on of the contest problems

Y/N Will be able to select one of a list of programming languages

Y/N Will be able to receive individual clarification responses from any Judge

Y/N Will be able to receive publically broadcasted messages from any Judge or Administrative User.

Y/N Will be able to log out of SLICE at any point

1.3.6 SLICE

- Y/N Will be able to compile submitted problems
- Y/N Will be able to provide an appropriate error message when a wrong username and password is enter
- Y/N Will be able to handle up to 30 teams

1.4 Non-Functional Requirements/Test Results

The non-functional requirement inventory is a list of non-functional system requirements. This list is composed of requirements that specify how the system should be. As more information is gathered about the project, this list may be subject to change:

- Y/N The system will be easily maintained
- Y/N The system will be stable.
- Y/N The system will be viewable on multiple browsers
- Y/N The system will run efficiently
- Y/N The system will be user friendly

1.5 Future Enhancements

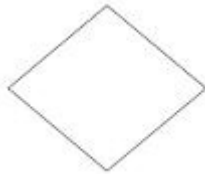
To continue our work on SLICE, we have several defined tasks to be implemented. Private and Public messaging, which work as a subset to the functionality of Problem Submission would be implemented and this would require the development of a client-side Javascript pop-up box. Problem Submission is almost completely functional. There are several issues with object references, but the sending, compilation, grading and response of participant submission has been working. The contest configuration in the Administrator interface is small, but integral task to the final completion of SLICE. The forms are already implemented in the user-interface and only the link to the database remains.

1.6 SLICE ER Diagram

An ER diagram or Entity-Relationship Diagram illustrates relations between entities in a database.

Legend

We adhere to the conventions of standard ER Diagrams, but to better explain our diagrams we describe each symbol below:



Relation: A relation between two entities



Entity: A data entity

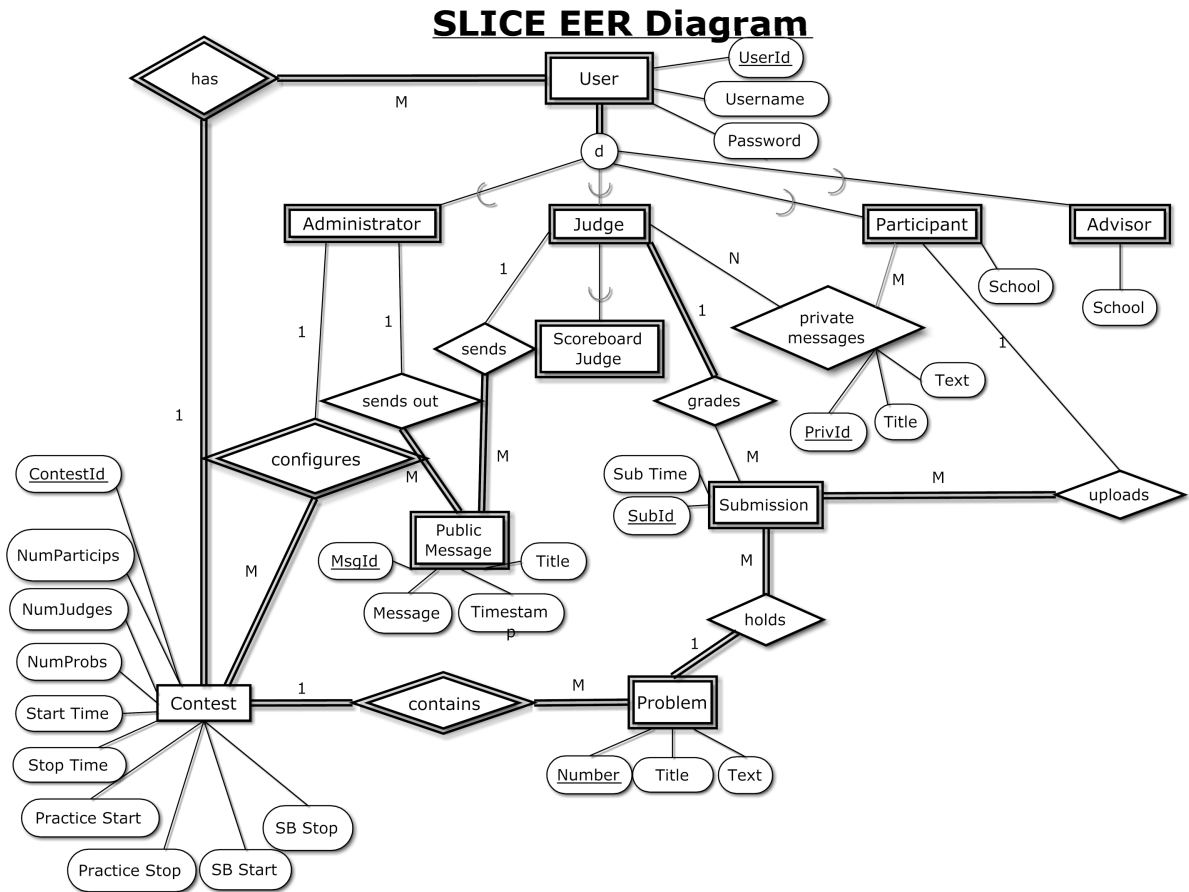


Attribute: An attribute associated with an entity or relation



Connection: The connection used between entities, attributes and relations

Figure 1.1: SLICE ER Diagram



1.7 Logical Data Dictionary Description

A Logical Data Dictionary is an inventory of all data entities and variables used within a system. All data entered, created, changed or stored by the system is included in the Logical Data Dictionary. Our data dictionary contains the following columns for each data entity described below:

- Data Name - The name of the data entity
- Applicable to - The views that the data is used or displayed in
- Data Type - How the data entity will be stored.
- Data Size - How large that data entity can be
- Description - What the data entity is used for
- Acceptable Input - A descriptor of allowed input
- Good Example of Input - An acceptable example input
- Notes - Additional comments on an entity The data dictionary is available on the $P = NP_{solutions}$ website as a Microsoft Excel File.

The Data Dictionary is located in Appendix B.

1.8 Prototype Screens

The following images were used for feature discovery and outline what SLICE will look like aesthetically.

1.8.1 General View

Login Screen for SLICE



1.8.2 Administrative User Views

Contest Configuration Page

The screenshot shows the SLICE web interface for configuring a contest. The page has a green header with the SLICE logo on the left and the text "Siena College HS Programming Contest" in the center. A "Logout" link is in the top right. A left sidebar contains navigation buttons: "Configuration" (highlighted), "Manage Users", "Manage Problems", "Messaging", and "Scoreboard". The main content area is titled "Contest Configuration" and contains five input fields: "Contest title", "Start time", "End time", "Practice length", and "Scoreboard length". A "Submit" button is located below these fields. At the bottom of the page, there is a "Details" link with a right-pointing arrow.

SLICE [Logout](#)

Siena College HS Programming Contest

Configuration

- Configuration
- Manage Users
- Manage Problems
- Messaging
- Scoreboard

Contest Configuration

Contest title :

Start time :

End time :

Practice length :

Scoreboard length :

[▶ Details](#)

Manage Users Page

SLICE [Logout](#)

Siena College HS Programming Contest

Configuration | Manage Users | Manage Problems | Messaging | Scoreboard

Manage Users

Username :

Team name :

Password :

Confirm password :

Associated teams <input type="text"/>	Associated problems Problem 1 Problem 2
--	---

User type : Judge

Niskayuna
Saratoga
Troy HS
Dr. Lim
Dr. Briemer

▶ [Details](#)

Manage Problems

SLICE Siena College HS Programming Contest [Logout](#)

Manage Problems

Problem title :

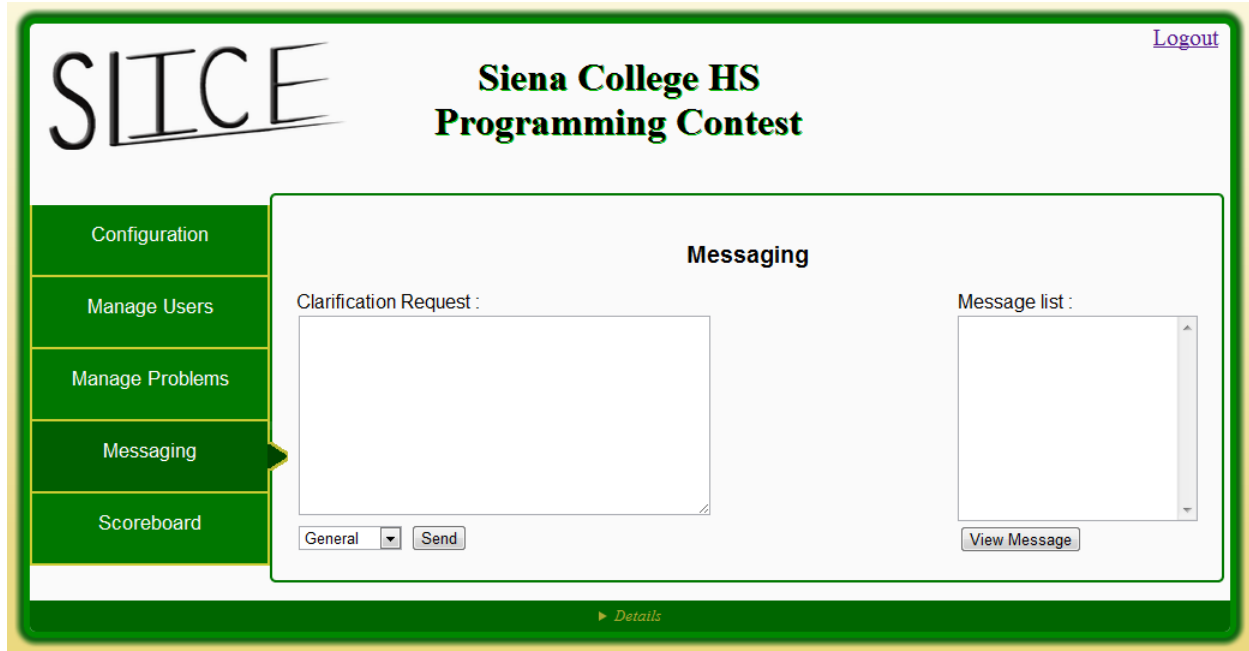
Problem input : No file chosen

Problem output : No file chosen

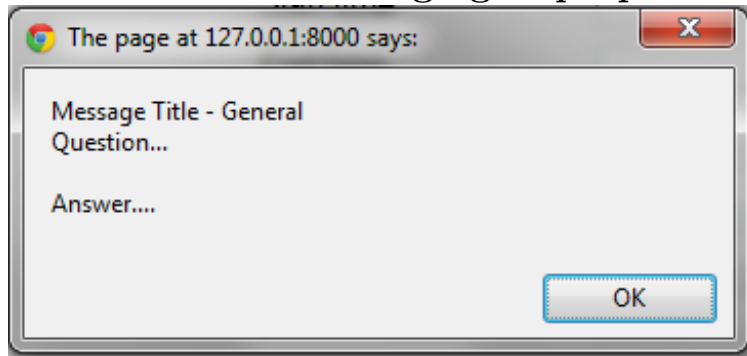
Associated judge :

[Details](#)

Messaging Page



Administrator Messaging Pop-up View



Scoreboard Page

The screenshot shows the SLICE Scoreboard Page for the Siena College HS Programming Contest. The page features a green sidebar with navigation options: Configuration, Manage Users, Manage Problems, Messaging, and Scoreboard (which is highlighted). The main content area displays a scoreboard table with columns for Team and seven Problems. The table shows scores for Team 1, Team 2, and Team 3 across the seven problems. A 'Logout' link is visible in the top right corner, and a 'Details' link is at the bottom center.

SLICE [Logout](#)

**Siena College HS
Programming Contest**

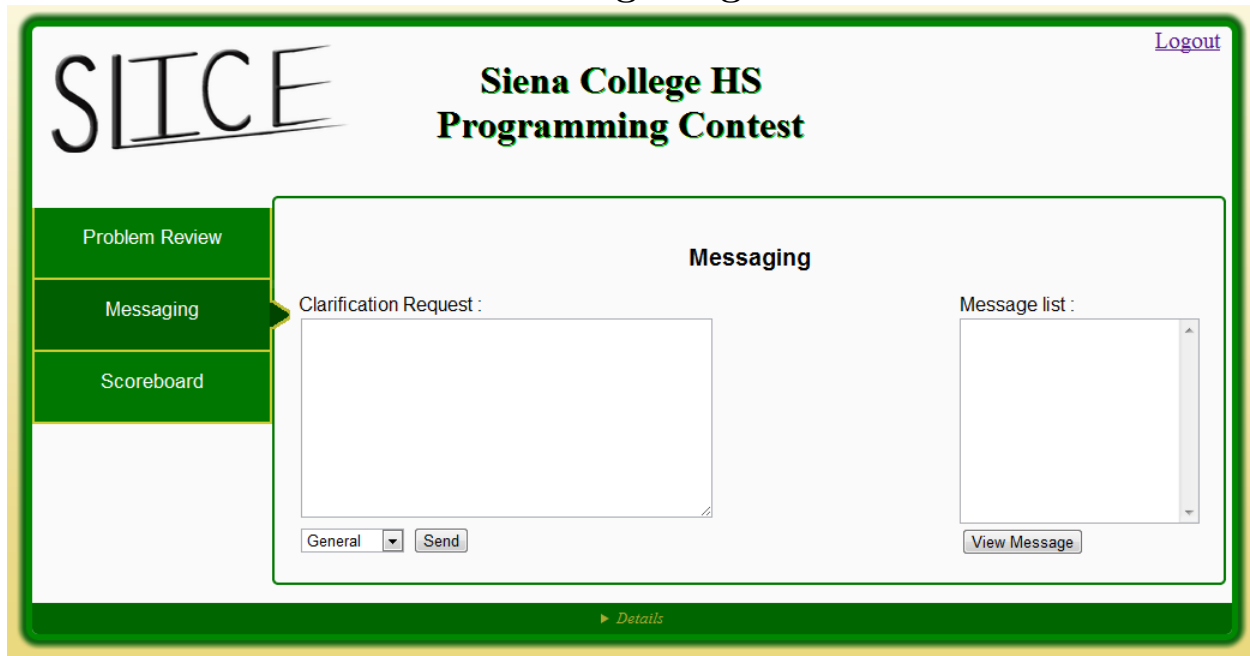
Scoreboard

	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Team 1	00:12	00:12	00:12	00:12	00:12		00:12
Team 2			00:12		00:12		
Team 3	00:12						

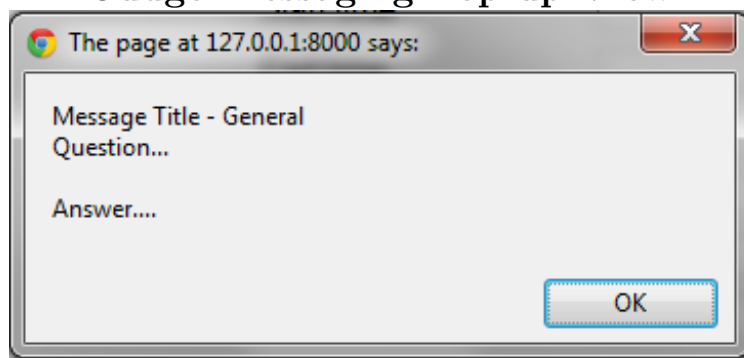
► Details

1.8.3 Judge User Views

Message Page



Judge Messaging Pop-up View



Problem Review Page

The screenshot shows a web interface for a programming contest. At the top left is the 'SLICE' logo. To its right is the text 'Siena College HS Programming Contest'. In the top right corner, there is a 'Logout' link. On the left side, there is a vertical navigation menu with three items: 'Problem Review' (highlighted with a green arrow), 'Messaging', and 'Scoreboard'. The main content area is titled 'Problem Submission' and contains a text box with the text 'It Works!'. Below the text box is a dropdown menu currently showing 'Correct', with a list of other options: 'Wrong Answer', 'Formatting Error', 'Time Limit Exceeded', 'Compiler Error', and 'Run-Time Error'. To the right of the dropdown is a 'Submit Grade' button. In the top right of the main content area, there is a section titled 'Problems to Review' which includes a dropdown menu showing 'Team 0 Problem 2 16:02:50' and a 'Review Submission' button. At the bottom center of the page, there is a 'Details' link with a right-pointing arrow.

Scoreboard Page

The screenshot shows a web interface for a programming contest scoreboard. On the left is a green sidebar with three menu items: "Problem Review", "Messaging", and "Scoreboard". The "Scoreboard" item is highlighted with a white arrow pointing to the right. The main content area has a white background with a green border. At the top left of this area is the "SLICE" logo. To its right is the text "Siena College HS Programming Contest". In the top right corner of the main area is a "Logout" link. Below the title is a table titled "Scoreboard". The table has 8 columns: a blank header, "Problem 1", "Problem 2", "Problem 3", "Problem 4", "Problem 5", "Problem 6", and "Problem 7". There are three rows of data for "Team 1", "Team 2", and "Team 3". The "00:12" time is present in the "Problem 1" column for Team 1, the "Problem 3" column for Team 2, and the "Problem 1" column for Team 3. At the bottom center of the main area is a "Details" link with a right-pointing arrow.

SLICE

**Siena College HS
Programming Contest**

[Logout](#)

Scoreboard

	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Team 1	00:12	00:12	00:12	00:12	00:12		00:12
Team 2			00:12		00:12		
Team 3	00:12						

[▶ Details](#)

Scoreboard Judge Scoreboard View

The screenshot shows a web interface for the SLICE Judge Scoreboard. At the top left is the SLICE logo. To its right is the text "Siena College HS Programming Contest" and a "Logout" link. On the left side, there is a vertical menu with three items: "Problem Review", "Messaging", and "Scoreboard", with "Scoreboard" being the active item. The main content area is titled "Scoreboard" and contains a table with 8 columns: "Team", "Problem 1", "Problem 2", "Problem 3", "Problem 4", "Problem 5", "Problem 6", and "Problem 7". The table has three rows: "Team 1", "Team 2", and "Team 3". Below the table, there is a form with a dropdown menu showing "Niskayuna", another dropdown menu showing "12:12 - Compiler Error", and a "Change Score" button. At the bottom center of the interface, there is a "Details" link with a right-pointing arrow.

SLICE [Logout](#)

**Siena College HS
Programming Contest**

Problem Review
Messaging
Scoreboard

Scoreboard

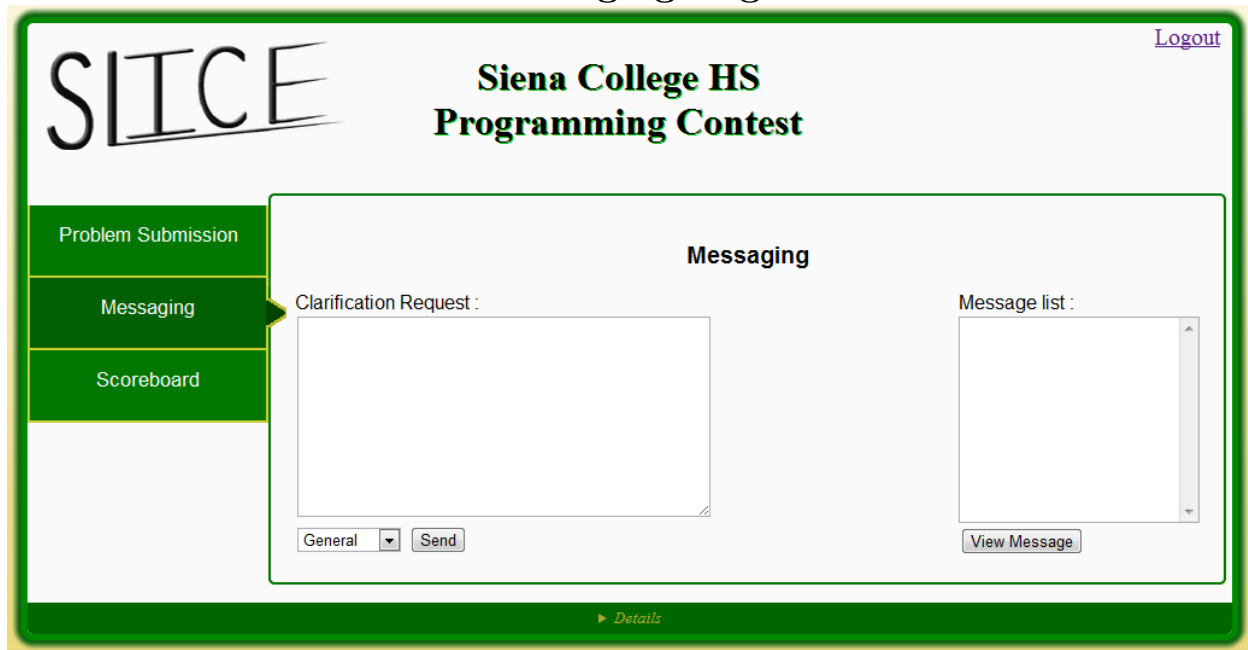
	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Team 1	00:12	00:12	00:12	00:12	00:12		00:12
Team 2			00:12		00:12		
Team 3	00:12						

Niskayuna | 12:12 - Compiler Error | Change Score

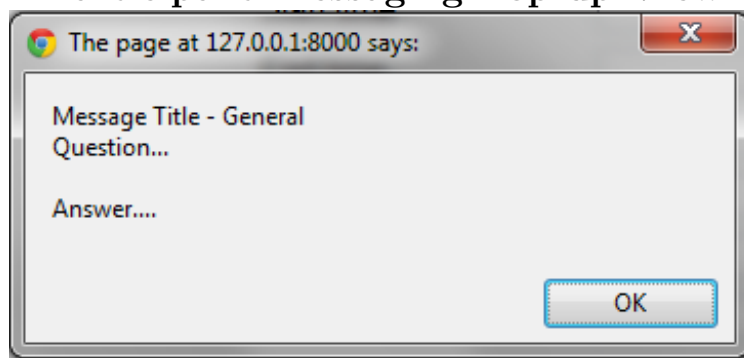
[▶ Details](#)

1.8.4 Participant User Views

Messaging Page



Participant Messaging Pop-up View



Scoreboard Page

The screenshot shows a web interface for a programming contest. At the top left is the 'SLICE' logo. To its right is the title 'Siena College HS Programming Contest' and a 'Logout' link. On the left side, there is a vertical menu with three items: 'Problem Submission', 'Messaging', and 'Scoreboard'. The 'Scoreboard' item is highlighted with a green arrow. The main content area is titled 'Scoreboard' and contains a table with 8 columns: 'Team', 'Problem 1', 'Problem 2', 'Problem 3', 'Problem 4', 'Problem 5', 'Problem 6', and 'Problem 7'. The table has 4 rows: a header row, and three data rows for 'Team 1', 'Team 2', and 'Team 3'. At the bottom center of the page, there is a 'Details' link with a right-pointing arrow.

SLICE [Logout](#)

**Siena College HS
Programming Contest**

Problem Submission

Messaging

Scoreboard

Scoreboard

	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7
Team 1	00:12	00:12	00:12	00:12	00:12		00:12
Team 2			00:12		00:12		
Team 3	00:12						

► Details

Submission Page

The screenshot shows a web interface for a programming contest. At the top left is the 'SLICE' logo. To its right is the text 'Siena College HS Programming Contest' in green. In the top right corner, there is a 'Logout' link. On the left side, there is a vertical navigation menu with three items: 'Problem Submission' (highlighted with a green arrow), 'Messaging', and 'Scoreboard'. The main content area is titled 'Problem Submission' and contains a file upload section with a 'Choose File' button, the filename 'problem1.java', a dropdown menu showing 'Problem 1', and a 'Submit' button. To the right of this section, the text 'Correct Problems To Be Graded' is displayed. At the bottom center of the page, there is a 'Details' link with a right-pointing arrow.

Participant Graded Page

The screenshot shows a web interface for the SLICE Siena College HS Programming Contest. The page has a green header with the SLICE logo and the contest name. A sidebar on the left contains navigation links for Problem Submission, Messaging, and Scoreboard. The main content area is titled 'Problem Submission' and features a file upload section with a 'Choose File' button, a 'No file chosen' message, a dropdown menu, and a 'Submit' button. To the right, there are two sections: 'Correct Problems' showing 'Problem 1 - 4:02 p.m.' and 'To Be Graded' showing 'Problem 2 - 4:02 p.m.'. A 'Logout' link is in the top right corner, and a 'Details' link is at the bottom center.

1.9 Test Plan

1.9.1 Testing Approach

The method our team, P=NP, will be using to test SLICE is to follow the guidelines outlined within this document. This method is to make certain that all Functional and Non-Functional requirements are met. The process will be a thorough, multi-step process validating SLICE. Smaller individual concepts all the way up to the completed system will all be tested.

Each step of this process consists of a Unit Test script. The script will be run repetitively until that test case is considered passed. Every test case within the Unit Test must pass. This process is done with each Unit Test. When each Unit Test is passed the Team will then test to see if the Units can work properly together.

Finally the system must be tested as a whole to guarantee that all requirements are met. On March 29th, 2012 P=NP will meet up with ExoNet to test each others projects on a small scale. We will then come back together on April 12, 2012 along with the rest of our software engineering class and others in the computer science department. This is so we can continue to test SLICE as a whole and to test its capability of managing 30 teams at once. We will perform a mock programming contest.

If at any point an error is found, P=NP will develop a solution and must repeat the testing procedure. This is to make sure that the problem is now fixed but to also test that the solution did not create a new problem.

Testing will be finished when everything is passed and P=NP has, as a team, determined all Functional and Non-Functional requirements have been met.

- The rest of the Test Plan can be found in the Appendix C.

1.10 Development and Production Environment

Our PC is an Optiplex 760 (name: seb2)

- Windows Vista Enterprise (32 bit) with service pack 1 installed
- An Intel Core2 Duo CPU processor (E7500 @ 2.93GHz)
- 4 GB of Memory

Our Mac is an iMac (model identifier: iMac5,1)

- Mac OS X, version 10.6.4
- An Intel Core 2 Duo Processor (667 MHz)
- 1 GB of Memory

Our server is an x86_64 PC

- Hostname: oraserv.cs.siena.edu
- CentOS 5.2 (final)
- Kernal: 2.6.18-92.el5
- Intel Xeon 2.66 GHz CPU
- 8 GB of Memory
- Java SE Runtime Environment (build 1.6.0_10-rc-b28)
- GCC Version 4.1.2 20071124 (Red Hat 4.1.2-42)
- Python 2.7.2
- Django 1.4 pre-alpha
- Python setuptools 0.6
- mod_wsgi Apache module

Our operating environment is a web based application that is only to be used within the SoS (Siena School of Science) network. The maintenance environment includes all of the hardware and software that is used to modify SLICE both in programming code and in visual appearance. This includes the computers in the Software Engineering Lab (stated above) and the software Adobe Dreamweaver, Adobe Fireworks, Adobe Photoshop, Internet Explorer, Mozilla Firefox, Google Chrome, Safari, etc

Appendix A

Glossary of Terms and Gantt Chart

- Actor: An entity in UML Use Case Diagrams and UML Activity Diagrams. It represents the human and non-human external entities that interact with the system
- Activity Diagram: A diagram based on the Unified Model Language (UML). This represents the processes that comprise a certain activity within the system. These diagrams are generally created with the perspective of an actor in mind.
- Compiler: A program that reads in source code and generates an executable.
- Data Flow Diagrams: Used to show how data is moved and processed within a system. There are various levels, each providing more detail than the next.
- Data Flows: A component of a Data Flow Diagram that represents the movement of data.
- Data Stores: A component of a Data Flow Diagram that represents a location in which information or data is stored.
- Django: A Python Web framework
- External Entities: A component of a Data Flow Diagram that represents any human or non-human user of a Software System.

- Functional Requirements Inventory: Define what the system will be able to do and what is testable about the system.
- Hardware: The physical parts of a computer, such as the hard drive and the CPU.
- HTML: Hypertext Markup Language is the scripting language used to describe the information contained on a website. HTML utilizes Cascading Style Sheets (CSS) to generate the style of the page. HTML and CSS are parsed by web browsers, such as Internet Explorer and Firefox, to render the websites for users Software: The intangible components of a computer and server. It is a set of machine-level instructions that is run from within the memory, and is used to perform a specific set of functions. Examples include Microsoft Word, Adobe Photoshop, and Mozilla Firefox.
- Microsoft Internet Explorer a web-browser developed by Microsoft
- Non-Functional Requirements: Specifies how a product is supposed to be in relationship to the functional requirements.
- Process: A component of a Data Flow Diagram that represents an activity that transforms or manipulates data.
- UML: Unified Modeling Language is the industry-standard language for the specification. Visualization, construction, and documentation of the components of software systems.
- Unit Testing: a method by which individual units of source code are tested to determine if they are fit for use. A unit is the smallest testable part of an application
- Use Case Diagram: Represents the high-level functions of the system. Also depicts how actors interact with each of those functions.
- User Case Narrative: an explanation of the functions and abilities users have for a specific Software System.

Appendix B

Data Dictionary

Below are the screen shots of P=NP solutions Data Dictionary excel file.

	A	B	C	D	E	F	G	H
1	Team Name	P=NP Solutions						
2	Project Name	S.L.I.C.E.						
3	Client Name	Dr. Darren Lim						
4								
5	Data Name	Applicable to	Data Type	Data Size	Description	Acceptable Input	Good Example of Input	Notes
11	currTime	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	5 Chars	Current Time in Contest	0-9,:	01:00	
12	resButton	Authenticate	boolean	4-5 Chars	Login Reset	TRUE, FALSE	TRUE	
13	resButton2	Configuration	boolean	4-5 Chars	Configuration Reset	TRUE, FALSE	TRUE	
14	pracStart	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	5 chars	Practice start-time	0-9,:	12:50	
15	pracEnd	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	5 chars	Practice end-time	0-9,:	13:00	
16	pracDur	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	2 Chars	Practice duration in minutes	0-9	10	
17	contStart	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	5 chars	Contest start-time	0-9,:	13:00	

18	contEnd	Authenticate, Submission, Scoring, Messaging, Scoreboard, Configuration	String	5 chars	Contest end-time	0-9, ;	17:00	
19	contDur	Configuration	String	2 Chars	Practice duration in hours	0-9	5	
20	submissionState	Submission, Scoring, Scoreboard	boolean	4-5 Chars	A problem submission's graded state	TRUE, FALSE	TRUE	
21	submissionTime	Submission, Scoring, Scoreboard	String	8 Chars	A problem submission's time-stamp	0-9, ;	13:33:33	Automatically generated by SLICE
22	submissionFeed	Submission, Scoring, Scoreboard	Enum	Enumerated Type	Submission Feedback	Correct Answer, Wrong Answer, Run-Time Error, Compiler Error, Formatting Error	Correct Answer	
23	scoreFreeze	Configuration, Scoreboard	String	5 chars	Time from contest end time the scoreboard is inactive to participants	0-9, ;	02:00	
24	judgeList	Configuration Judge Pop-Up	List of Strings	Up to 15 Strings	The list of all Judge users	A-Z, a-z, 0-9	{Judge1, Judge 2}	
25	participantList	Configuration Participant/Advisor Pop-Up	List of Strings	Up to 60 Strings	The list of all participants and team advisors	A-Z, a-z, 0-9	{sienaGold1, sienaGold1Adv}	
26	selectedUser	Configuration Participant/Advisor Pop-Up and Judge Pop-Up	String	8-15 Chars	The selected user in the configuration Pop-Up	A-Z, a-z, 0-9	seinaGold1	
27	editUser	Configuration Participant/Advisor Pop-Up and Judge Pop-Up	boolean	4-5 Chars	Edit the currently selected user	TRUE, FALSE	TRUE	
28	isTeamAdvisor	Configuration Participant/Advisor Pop-Up	boolean	4-5 Chars	Flag the user as a Team Advisor	TRUE, FALSE	TRUE	

5	Data Name	Applicable to	Data Type	Data Size	Description	Acceptable Input	Good Example of Input	Notes
29	currPass	Administrator Change Password Pop-Up	String	8-15 Chars	Password	ASCII 48-126	r0ll1ngthun3;!	
30	newPass	Administrator Change Password Pop-Up	String	8-15 Chars	Password	ASCII 48-126	r0ll1ngthun3;!	
31	confirmPass	Administrator Change Password Pop-Up	String	8-15 Chars	Password	ASCII 48-126	r0ll1ngthun3;!	
32	personalMessList	Administrator Messaging, Participant Messaging, Judge Messaging	List of (String,String) Tuples	Up to 100 Messages	Personally sent messages	(ASCII 48-126, ASCII 48-126)	{{(Does the output for problem 1 needs a new line?, Problem 1), (What is the example output for problem 2?, Problem 2)}}	
33	publicMessList	Administrator Messaging, Participant Messaging, Judge Messaging, Team Advisor Messaging	List of (String,String) Tuples	Up to 100 Messages	Publically send messages	(ASCII 48-126, ASCII 48-126)	{{(Please send only source files, not .class files., General)}}	
34	selectedMess	Administrator Messaging, Participant Messaging, Judge Messaging, Team Advisor Messaging	(String, String) Tuple	(Up to 500 Chars, Up to 50 Chars)	The user selected message	ASCII 48-126	(Please send only source files, not .class files., General)	
35	userMess	Administrator Messaging, Participant Messaging, Judge Messaging, Team Advisor Messaging	(String, String) Tuple	(Up to 500 Chars, Up to 50 Chars)	The user created message	ASCII 48-126	(Does the output for problem 1 needs a new line?, Problem 1)	
36	scoresTable	Scoreboard	Two Dimensional Array of Strings	Number of Participants x Number of Problems	The Scoreboard information	A-Z, a-z, 0-9	{{Team1,01:00,02:00}, {Team2,00:30,4:00}}	
37	isScoringJudge	Configuration, Scoreboard	boolean	4-5 Chars	Flag for the Judge with scoreboard permissions	TRUE, FALSE	TRUE	
38								

Appendix C

Test Plan

C.1 Testing Approach

The method our team, P=NP, will be using to test SLICE is to follow the guidelines outlined within this document. This method is to make certain that all Functional and Non-Functional requirements are met. The process will be a thorough, multi-step process validating SLICE. Smaller individual concepts all the way up to the completed system will all be tested.

Each step of this process consists of a Unit Test script. The script will be run repetitively until that test case is considered passed. Every test case within the Unit Test must pass. This process is done with each Unit Test. When each Unit Test is passed the Team will then test to see if the Units can work properly together.

Finally the system must be tested as a whole to guarantee that all requirements are met. On March 29th, 2012 P=NP will meet up with ExoNet to test each others projects on a small scale. We will then come back together on April 12, 2012 along with the rest of our software engineering class and others in the computer science department. This is so we can continue to test SLICE as a whole and to test its capability of managing 30 teams at once. We will perform a mock programming contest.

If at any point an error is found, P=NP will develop a solution and must repeat the testing procedure. This is to make sure that the problem is now fixed but to also test that the solution did not create a new problem.

Testing will be finished when everything is passed and P=NP has, as a team, determined all Functional and Non-Functional requirements have been met.

C.2 Testing Plan Identifier

The reason for the Test Plan for S.L.I.C.E is it is necessary to test all the requirements to make sure the program is functional. The Test Plan will describe how the project is supposed to work. Throughout the testing process, the Test Plan will change when a requirement is altered or even removed. A detailed document will be provided showing changes to any requirements and if one had failed or passed during any point of the Test Plan. An updated version of this document will be provided at the end of the Detailed Design phase. The final version will be provided at the end of the Acceptance Test phase.

C.3 Test Results

C.3.1 Test Items and Functional Requirements Inventory

The Functional Requirements will be tested using a detailed checklist. The checklist is used to guide the members of P=NP to assure that each requirement is tested and tested properly. The purpose of this is to guarantee that all of the clients, Dr. Lim, functional requirements are met and are met for every user.

We have five unique users of SLICE. They are the Team Advisor, Administrator, Judge, Scoreboard Judge, and Participant. We also have functional requirements for SLICE itself.

C.3.2 Administrative

- Y/N Will be able to log into SLICE with a unique username and password
- Y/N Will be able to create the Judge accounts
- Y/N Will be able to create the Participant accounts
- Y/N Will be able to create the Team Advisor accounts
- Y/N Will be able to give Scoreboard Judge abilities to one judge
- Y/N Will be able to send broadcast messages to all users
- Y/N Will be able to send broadcast messages to any subset of users
- Y/N Will be able to manage the set of problems
- Y/N Will be able to manage test input
- Y/N Will be able to manage correct output for each contest problem
- Y/N Will be able to designate the length of the contest
- Y/N Will be able to set the contest start time
- Y/N Will be able to set the contest stop time
- Y/N Will be able to set the practice time for the contest
- Y/N Will be able to restrict the view of the scoreboard at a set time towards the end of the contest
- Y/N Will be able to choose the list of allowed programming languages
- Y/N Will be able to log out of SLICE at any point

C.3.3 Team Advisor

- Y/N Will be able to log into SLICE with the Advisor user name and password
- Y/N Will be able to view each individual problem
- Y/N Will be able to view the scoreboard while active
- Y/N Will be able to receive all publically broadcasted contest clarifications
- Y/N Will be able to log out of SLICE at any point

C.3.4 Judge

- Y/N Will be able to log onto SLICE with a given user name and password
- Y/N Will be able to view Participant submitted solutions
- Y/N Will be able to run the submitted solution using the test input provided in SLICE that corresponds to the associated question
- Y/N Will be able to submit a response to the team stating either "Correct Solution, Compiler Error, Running Time Error, Wrong Answer, OR Time-Limit Exceeded"
- Y/N Will be able to log out of SLICE at any point

C.3.5 Scoreboard Judge

Y/N Will be able to do everything the Judge can do

Y/N Will be able to edit the scoreboard

Y/N Will be able to log out of SLICE at any point

C.3.6 Participant

Y/N Will be able to log into SLICE with a given user name and password

Y/N Will be able to view the contest scoreboard while active

Y/N Will be able to submit problem-solutions

Y/N Will be able to submit clarification-requests

Y/N Will be able to make a general request

Y/N Will be able to associate a problem with request

Y/N Will be able to select on of the contest problems

Y/N Will be able to select one of a list of programming languages

Y/N Will be able to receive individual clarification responses from any Judge

Y/N Will be able to receive publically broadcasted messages from any Judge or Administrative User.

Y/N Will be able to log out of SLICE at any point

C.3.7 SLICE

Y/N Will be able to compile submitted problems

Y/N Will be able to provide an appropriate error message when a wrong username and password is enter

Y/N Will be able to handle up to 30 teams

C.3.8 Non-Functional Requirements

Non-Functional Requirements are what SLICE is and looks like rather than what it does. These are harder to test but just as important as the Functional Requirements.

Y/N The system will be easily maintained

Y/N The system will be stable.

Y/N The system will be viewable on multiple browsers

Y/N The system will run efficiently

Y/N The system will be user friendly

C.4 Integration Testing

Integration testing involves the combination and testing of individual software modules as a group. The individual modules themselves will have already been unit tested. This ensures that any problems that arise within integration testing are a result of integration, not the individual software modules. This enables the system as a whole to perform higher order functions.

C.5 Exception Handling To Test

SLICE will be designed as a secure system. All users must authenticate to have access to SLICE. Participants may only submit source code of problem solutions. The solutions are compiled by a Judge on a central server under a restricted account environment separate from the contest problems test input and output. There are also time limits set on both compile time and execution time of Participant problem solutions to avoid denial of service attacks on SLICE.

C.6 Acceptance Test - Acceptance Criteria

The acceptance test is concerned with validating the functional and non-functional requirements of a system. Utilizing this testing process will validate the acceptability of a system. Once a system is declared complete the test plan document and the Unit test logs (described and shown below) are helpful tools to explain to those who were never involved in the project why and how the project is validated. The document validates the integrity of the steps a team took to show that the system meets the requirements and specifications.

The acceptance criteria for the system will be defined by the functional requirements inventory that was explained in further detail earlier in the document. The part of the system that is testable by users is considered the functional requirements. Non-functional requirements are not testable items, such as look or maintainability. By the end of the Acceptance Test some of the requirements may have altered or changed due to complications. Nonetheless, P=NP Solutions will state which of those requirements were met or not met.

SLICE will be tested on Internet Explorer, Mozilla Firefox, Google Chrome and Apple Safari. P=NP Solutions will be doing weekly tests on SLICE whenever a new feature is added. We will then be conducting a smaller test of about ten teams. Upon completing those two rounds of testing, we will then have a larger test of thirty different teams. P=NP will be using these methods of testing to determine if our stated requirements are met.

Appendix D

Unit Test

D.1 Unit Tests

The following is a list of the separate Unit Tests that will be tested from now until the end of our project. Each Unit will either pass or fail. A full system test will be performed once all Unit Tests are passed.

P=NP Solutions 8 Unit Test

- User Login
- Problem Submission
- Contest Setup
- Private Messaging
- Public Messaging
- View Scoreboard
- Edit Scoreboard
- Log out

D.1.1 Unit Test Cases

Each Unit Test is divided into test cases. Test Cases include a test number for tracking, along with a description of the purpose for the test case. The excel table also includes the action to be performed or the input, steps to be executed, the state before the test, and the expected resulted. There are also columns that are to be filled in when one is testing SLICE. There is a column to edit to say what happened when testing that specific test case.

Below are screen shots of the excel file of our 8 different Unit Tests with their Test Cases.

Figure D.1: Directory of the Unit Tests

	A	B	C	D	E	F	G
1	System Test - Test Results						
2							
3	Team Name	P=NP Solutions					
4	Project Name	S.L.I.C.E.					
5	Client Name	Dr. Darren Lim					
6							
7	Unit Test Directory						
8							
9	Pass/Fail Status	Unit Number	Unit Test Name	Date Last Tested	Comments or description	Integrated with these units	
10							
11	F	83%	1	User Login	3/26/12		All
12	P	100%	2	Log Out	3/26/12		All
13	F	0%	3	Contest Setup	4/24/12		4, 5, 6, 7, 8
14	F	29%	4	Problem Submission	4/24/12		None
15	F	0%	5	Public Messaging	4/24/12		None
16	F	0%	6	Private Messaging	4/24/12		None
17	F	33%	7	View Scoreboard	4/24/12		8
18	F	0%	8	Edit Scoreboard	4/24/12		None
19							
20							
21							

Figure D.2: User Login

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	User Login										
3	Allows users to log on using their accounts.										
4											
5											
	Test Case										
6	Pass/Fail Status	Test Number	Description	Action to Perform Test (Input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
7											
8	P	1.001	Null Username Field	Leave Username field blank	Fill out remainder of form and press submit	Empty Form	Failed Login	Failed Login	Small Scale Test	RS, MF	3/26/12
9	P	1.002	Null Password Field	Leave Password field blank	Fill out remainder of form and press submit	Empty Form	Failed Login	Failed Login	Small Scale Test	RS, MF	3/26/12
10	P	1.003	Incorrect Password for Given Username	Input incorrect Username and Password combination	Fill out remainder of form and press submit	Empty Form	Failed Login	Failed Login	Small Scale Test	RS, MF	3/26/12
11	P	1.004	Nonexisting Username	Input a Username that does not exist	Fill out remainder of form and press submit	Empty Form	Failed Login	Failed Login	Small Scale Test	RS, MF	3/26/12
12	F	1.005	Test for Lockout	Incorrectly attempt to log on 3 times with the same Username	Fill out remainder of form and press submit	Empty Form	Failed Login, Error Message: "Too many incorrect login attempts have been made. This Username has been temporarily locked. Please try again later."	Failed Login, No Error Message or Lockout	Small Scale Test	RS, MF	3/26/12
13	P	1.006	Correct Username and Password	Enter valid Username and Password	Fill out remainder of form and press submit	Empty Form	Successful Login	Successful Login	Small Scale Test	RS, MF	3/26/12
14	F	Unit Summary		83%	passing	5	passed				
15		6	tests		1	1	failed				
16											
17											
18											

Figure D.3: Problem Submission

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Problem Submission										
3	Allows participant user to submit problem submissions										
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Problem Submission										
3	Allows participant user to submit problem submissions										
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											

Figure D.4: Contest Setup

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Contest Setup										
3	Allows administrator to complete the initial contest setup										
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18	F	Unit Summary	tests	0%	passing	0	passed			Date of last test =	4/24/12
19		9			9	9	failed				
20											
21											
22											

Figure D.6: Public Messaging

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Public Messaging										
3	Allows user to send and receive public messages										
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											

Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
F	5.001	Failed to Send Public Message	Public message text	Attempt to send public message	No public message sent	Error message: "Delivery of message failed"	No message sent		RS, MF	4/24/12
F	5.002	Send Public Message (Success)	Public message text	Send public message	No public message sent	Message: "Message sent"	No message sent		RS, MF	4/24/12
F	5.003	Receive Public Message	Public message text	Send message text	No public message received	Reception of public message	No reception		RS, MF	4/24/12
F	Unit Summary		0%	passing	0	passed				
	3	tests		failed	3	failed				

15	Directory Page
----	--------------------------------

Figure D.7: View Scoreboard

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	View										
3	Scoreboard										
4	Allows user to view										
5	Scoreboard										
6	Test Case										
7	Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
8	F	6.001	View Active Scoreboard	Automatically Inputted	Go to Scoreboard Tab	None	Scoreboard of all Team with scores	Scoreboard and not dynamic		RS, MF	4/24/12
9	P	6.002	View Inactive Scoreboard	Automatically Inputted	Go to Scoreboard Tab	None	Blank Scoreboard	Blank scoreboard		RS, MF	4/24/12
10	F	6.003	Final Scoreboard	Activate Scoreboard	Go to Scoreboard Tab	None	Order of Winners with scores	Scoreboard and not dynamic		RS, MF	4/24/12
11											
12	F	Unit Summary		33%	passing	1	passed			Date of last test =	4/24/12
13		3	tests		failed	2	failed				
14											
15											

Figure D.8: Edit Scoreboard

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Edit Scoreboard										
3	Allows Scoreboard Judge to edit scoreboard										
4											
5	Test Case										
6	Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
7											
8	F	7.001	Edit Scoreboard	Correct Data text	Edit Scoreboard data	None	Correct data is in the Scoreboard index	No editing ability		RS, MF	4/24/12
9											
10	F	Unit Summary		0%	passing	0	passed			Date of last test =	4/24/12
11		1	tests			1	failed				
12											
13										Directory Page	

Figure D.9: Log Out

	A	B	C	D	E	F	G	H	I	J	K
1	SLICE										
2	Log Out										
3	Allows Users to Log out										
4											
5		Test Case									
6	Pass/Fail Status	Test Number	Description	Action to perform test (input)	Steps to be Executed	State Before Test	Expected result	Observed result	Comments	Tested By	Test Date
7											
8	P	8.001	Log Out of Slice	Log Out button	Click Log Out Button	None	To have no access to Slice but only the ability to log back in	Logged Out		RS, MF	3/26/12
9											
10	P	nit Summary	100%	passing	1	passed				Date of last test =	3/26/12
11		1	tests	0	failed						
12											
13		Directory Page									