

Acceptance Test

Requested by:
Dr. Darren Lim
Associate Professor of Computer Science
Siena College
Computer Science Department

Competitive Algorithm Calculation Testing in a Unified System (C.A.C.T.U.S.)

ExoNET Solutions

Prepared by: Paul Amodeo, Interface Expert
Tom Delaney, System Admin./ DBA
Stephanie Del Belso, Team Leader
David Purcell, Lead Programmer
Marco Samaritoni, Testing Manager

April 25th, 2012
C.A.C.T.U.S.
Acceptance Test

Contents

- Introduction 4
 - 1.1 Overview and Summary 4
- Requirements Inventory 5
 - 2.2 Functional Requirements Inventory 7
 - 2.3 Non-Functional Requirements Inventory 8
 - 2.4 Future Enhancements 8
 - 1.1.1 Log in Screen 9
 - 1.1.2 Log in Failed 9
 - 1.1.3 Contestant Screens 10
 - 1.1.3.1 View Clarifications 10
 - 1.1.3.2 Problems List 10
 - 1.1.3.3 Scoreboard View 11
 - 1.1.3.4 Submission Screen 11
 - 1.1.3.5 Submission Screen - Browse for source file 12
 - 1.1.3.6 Submission Screen – No file selected error 12
 - 1.1.4 Judge Screens 13
 - 1.1.4.1 Create Clarification 13
 - 1.1.4.2 View Clarifications 13
 - 1.1.4.3 Judge submissions 14
 - 1.1.4.4 Problems List 14
 - 1.1.4.5 Scoreboard View 15
 - 1.1.5 System Administrator Screens 15
 - 1.1.5.1 Change Admin password 15
 - 1.1.5.2 Change Admin password – Invalid old password error 16
 - 1.1.5.3 Change Admin password – New passwords do not match error 16
 - 1.1.5.4 Upload/ Edit problems 17
 - 1.1.5.5 Upload Problems - Browse for file 17
 - 1.1.5.6 Configure Contest Problems 18
 - 1.1.5.7 Configure Contest Settings 18
 - 1.1.5.8 Configure Contest Settings – Contest started 19
 - 1.1.5.9 Invalid Contest Length Error 19

1.1.5.10 Invalid Selection Error.....	20
1.1.5.11 Manage Accounts.....	20
1.1.5.12 Manage Accounts – Contestants	21
1.1.5.13 Manage Accounts – Judges.....	21
1.1.5.14 Manage Accounts – Spectator	22
1.1.6 Spectator Screens	22
1.1.6.1 Problems List.....	22
1.1.6.2 Scoreboard View	23
1.1.6.3 View Clarifications.....	23
1.2 Logical Data Dictionary	24
1.3 Logical Format of Data Files	25
Architectural Design Specification	26
1.2 Deliverables.....	27
1.3 Data Flow Diagrams (See Appendix A: Data Flow Diagrams)	28
1.4 Source Code	29
Project Timeline	34

Introduction

1.1 Overview and Summary

The size and complexity of the programming contests held at Siena College has brought about a need for a simple yet powerful solution. Dr. Lim has requested that a system be created to prepare and run programming contests simply and easily. The current contest system lacks the ability to keep a precise record of exactly when a team (consisting 1 to many contestants) submits a solution to a problem, the ability for teams to communicate electronically with judges (i.e. uploading problem solutions, checking ambiguities in the instructions, among any other form of legal correspondence dictated by the administrators of the contest). Also, in the current system, a scoreboard must be updated manually on a whiteboard, and judges lack the ability to run alternate arguments (parameters) on a submission, due to the lack of a way to submit contestants' code. Our product will solve these problems by:

- Keeping an electronic timestamp of all submissions made to judges for consideration
- Embedding a compiler into our system so that code may be submitted to judges, and also so the judges can run different data sets (arguments) against the contestants' code
- A digital scoreboard will be created so that all parties involved in the contest can see all teams' solved problems, and the timestamps for correct problem submissions
- All submissions will be stored in a database (MySQL); also, contestants' work will be saved periodically, so any system or software failure does not end in a complete loss of information for C.A.C.T.U.S..
- A chat window will be available within C.A.C.T.U.S., so that teams and judges can communicate with each other for ambiguities within contest problem sets, and any other pertinent information that can be shared within a given contest.

Requirements Inventory

2.1 User Case Narratives

The following User Case Narratives describe how users will interact with ExoNET Solutions' Competitive Algorithm Calculations Testing in a Unified System (C.A.C.T.U.S.) system. The narratives provide insight into each user's behavior within C.A.C.T.U.S. and allow ExoNET Solutions to discover user requirements. There are four types of users within C.A.C.T.U.S. The System Administrator configures the programming contest environment and has extensive privileges. Judges answer questions and monitor the scoreboard during the programming contest. Contestants participate in a programming contest and make Problem Submissions to contest problems. Spectators can view public information about a running programming contest.

System Administrator:

The System Administrator will be able to access C.A.C.T.U.S. by using a specific user name and password that will allow them into an administrative account. Within the administrative account, the System Administrator is prompted to input certain information in order to fully configure the contest environment. This information includes the uploading of contest-problems and problem solutions (including test cases), selection of languages (Java and possibly others), editing start and stop times of the competition, adding and dropping of Contestants, and designating the time for freezing the scoreboard during the competition. The System Administrator is also responsible for establishing user names and passwords for the Contestant accounts as well as Judge accounts and Spectator accounts.

Judge:

Each Judge is given a Judge account with a user name and password by the System Administrator. A Judge is allocated one or more Contestants to supervise for the duration of the contest.

When a Judge logs into C.A.C.T.U.S. using his or her Judge account, he or she will have access to a chat window and the contest scoreboard. A Judge can use the chat window to send text-based messages to other Judges and Contestants.

The chat window messages can be sent to all Judges and Contestants, or a subset of the Contestants that the Judge was assigned to supervise. A Judge may receive problem submissions from Contestants along with C.A.C.T.U.S.'s analysis of the submissions. The Judge may then review the problem submissions and can decide to approve or reject the submission. A Judge will be able to take over for another Judge that needs to take a leave of absence for any period of time. In this case, the Judge who is taking over will supervise all of the leaving Judge's Contestants.

Contestant:

Each Contestant is given a Contestant account with a unique user name and password assigned by the System Administrator. After logging into C.A.C.T.U.S. with the assigned Contestant account, the Contestant will be able to familiarize themselves with C.A.C.T.U.S. and get comfortable with the system before a contest is started. Once a contest has begun, a Contestant will have the ability to: submit problem submissions to C.A.C.T.U.S. in the form of source code, message their supervising Judge using a chat window, view the scoreboard and contest problems.

Spectator:

Each Spectator is given a Spectator account with a user name and password assigned by the System Administrator. A Spectator will be able to view the scoreboard at any time during a programming contest. A Spectator will be able to see every Contestant's name, score, and standing. A Spectator will be able to view the contest problems being used in a running programming contest. A Spectator will not be able to communicate with Contestants and Judges. A Spectator will be able to watch a programming contest unfold without interfering with the teams.

2.2 Functional Requirements Inventory

This is a list of functional requirements. A functional requirement defines a function of a software system or its components. This list is subject to change as ExoNET goes further into the project.

C.A.C.T.U.S (Competitive Algorithm Calculations Testing in a Unified System)-

- Responsiveness (specific ability of a functional unit to complete assigned tasks within a given time)
- Scalability (must be able to operate 30 teams)
- Efficiency (must get responses back within a reasonable time/if infinite loop-time limit exceeded)
- Time limit exceeded – specific time should be determined per contest (decided by system administrator).
- Robustness (if C.A.C.T.U.S. fails or crashes, users should be able to recover successfully)
- Saving option
- System must be secure (user's should not be able to access other user accounts)
- C.A.C.T.U.S. will be viewable on multiple web browsers
- Web browsers include: Mozilla FireFox, Internet Explorer, Google Chrome
- C.A.C.T.U.S will compile programs during the contest
- C.A.C.T.U.S will maintain a scoreboard during the contest.
- Scoreboard will list all Contestants in decreasing order of standing (determined by # of problems solved and total time)
- For every team, Scoreboard will show the time-stamp of every solved problem and list the total time for all solved problems
- Scoreboard will be able to be "frozen" after a set period of time

System Administrator-

- Securely log in and out of C.A.C.T.U.S.
- Upload problem solutions, contest problems and test cases
- Add/drop Contestants during contest creation
- Delete and create new Judge, Contestant, Spectator accounts
- Administrator will initialize user names and passwords for each user
- Change start and stop time for a contest
- Administrator will set freeze time for scoreboard
- Selecting Languages
- Java will always be a selection choice for the contest

Judge-

- Securely log in and out of C.A.C.T.U.S.
- View contest problems
- Judge will be able to view test cases
- View Scoreboard
- View chat window
- Judge will be able to send broadcast messages to all Contestants
- Judge will be able to send specific messages to their assigned Contestants
- Process submissions
- Judge will be able to 'accept' or 'decline' submissions sent to them by their specified Contestants

Contestant-

- Securely log in and log out of C.A.C.T.U.S.
- View contest problems
- View Scoreboard
- Contestants will be able to see their time stamp for each problem submitted
- View broad casted messages and chat window
- Contestants will be able to send messages to their assigned Judge
- Submit submissions of source code
- Compile their code

Spectator-

- Securely log in and log out of C.A.C.T.U.S.
- View contest problems
- View Scoreboard
- View only broadcast messages

2.3 Non-Functional Requirements Inventory

- Must be easily used (user friendly)
- Stability

2.4 Future Enhancements

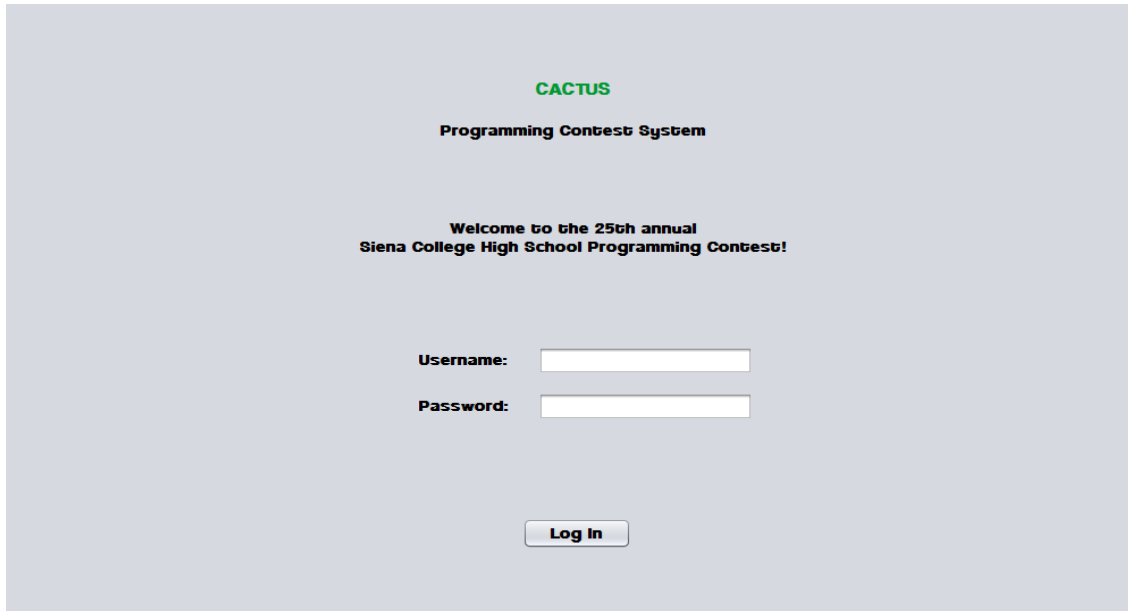
If ExoNET had more time to work on this software, we would make this project fully functional. This includes completing all pass/fail criteria that is specified by the Functional Requirements of this project.

External Design Specification

1.1 User Displays

1.1.1 Log in Screen

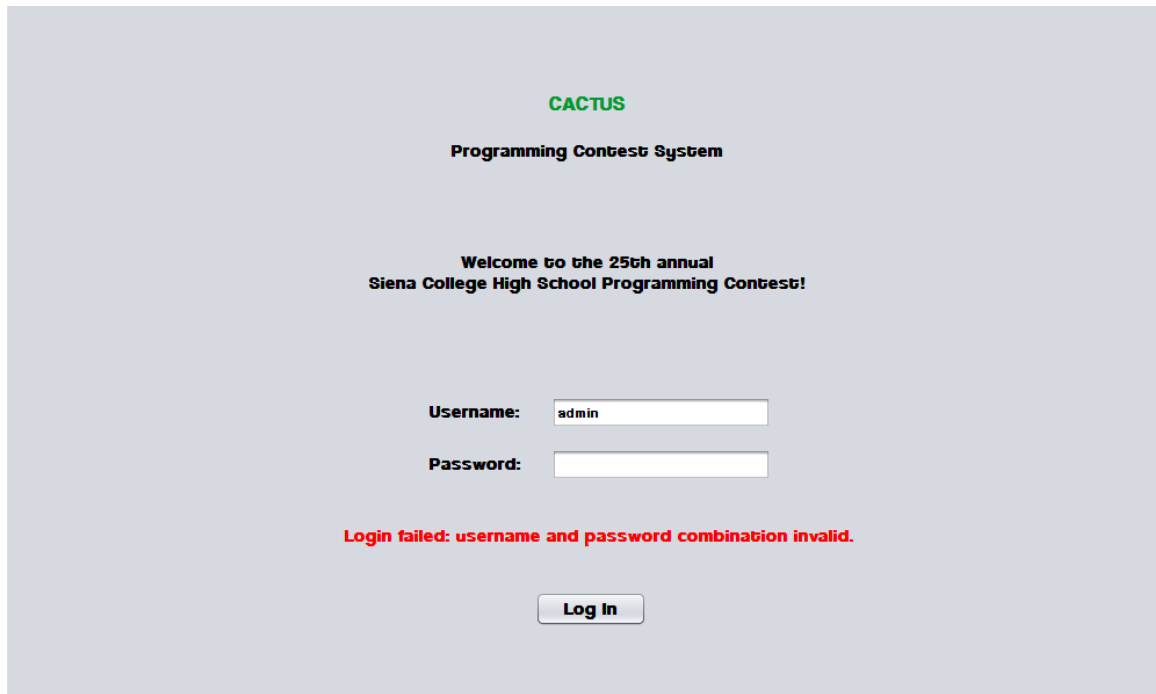
The login screen allows users to login to C.A.C.T.U.S. using a username and password.



The screenshot shows the login interface for the CACTUS Programming Contest System. At the top, the word "CACTUS" is displayed in green, followed by "Programming Contest System" in black. Below this, a welcome message reads: "Welcome to the 25th annual Siena College High School Programming Contest!". The login fields are labeled "Username:" and "Password:", each followed by a white input box. A "Log In" button is centered at the bottom of the form.

1.1.2 Log in Failed

An error message is displayed when a user enters an invalid username and password combination.

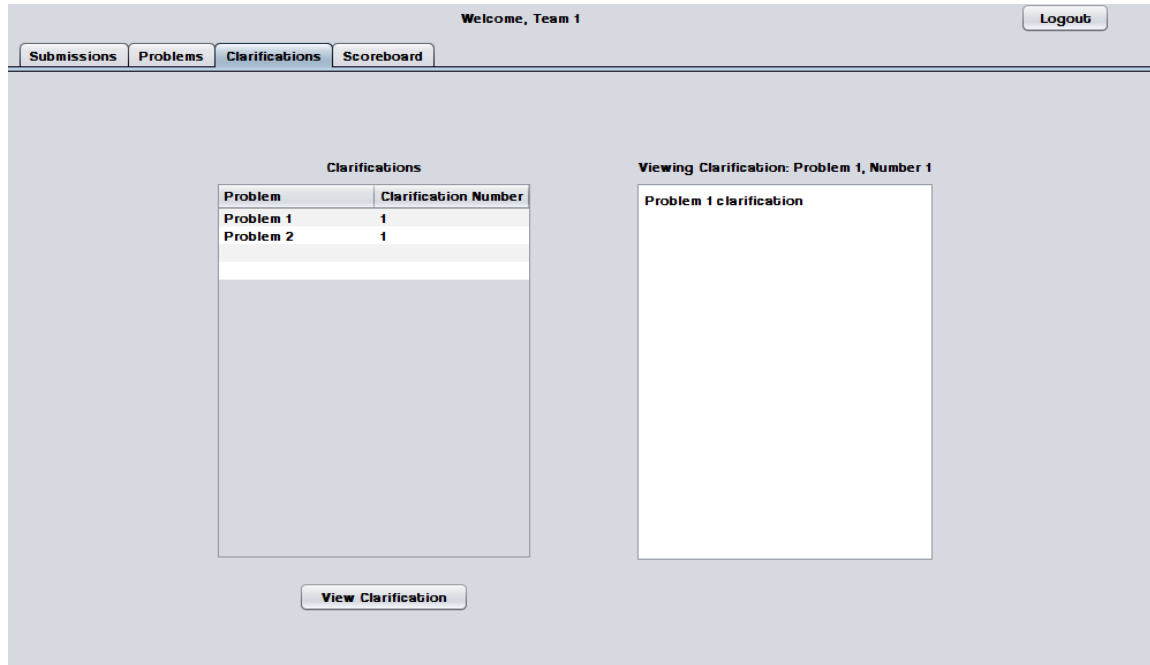


This screenshot shows the same login interface as the previous one, but with an error message. The "Username:" field now contains the text "admin". Below the password field, a red error message reads: "Login failed: username and password combination invalid." The "Log In" button remains at the bottom.

1.1.3 Contestant Screens

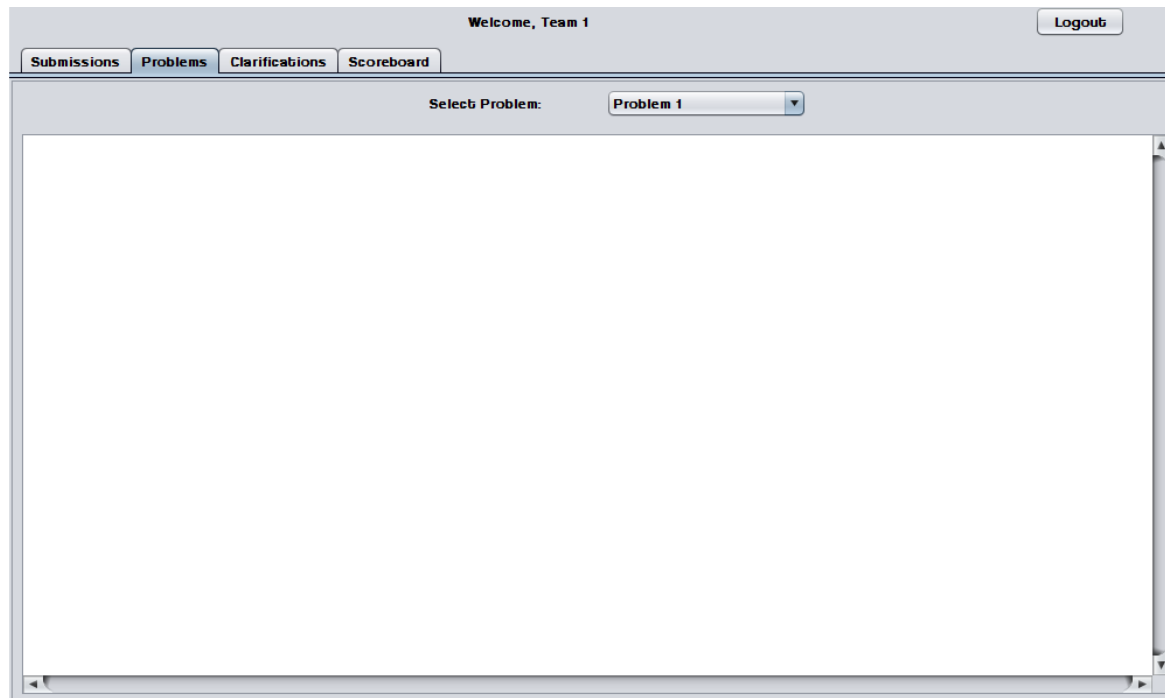
1.1.3.1 View Clarifications

Contestants can view all Clarifications sent out by the Judges.



1.1.3.2 Problems List

Contestants may select a contest problem to view in an embedded scroll area.



1.1.3.3 Scoreboard View

Contestants may view the contest scoreboard. The scoreboard must be manually refreshed and the scoreboard cannot be refreshed after it has been frozen.

Team	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7	Penalty Time	Total Time
Team 1	00:05	00:11							00:16
Team 2	00:15	00:10							00:25
Team 3			00:15						00:15
Team 4			00:30						00:30
Team 5								00:05	00:05
Team 6									
Team 7									
Team 8									
Team 9									
Team 10									
Team 11									
Team 12									
Team 13									
Team 14									
Team 15									
Team 16									
Team 17									
Team 18									
Team 19									
Team 20									
Team 21									
Team 22									
Team 23									
Team 24									
Team 25									
Team 26									
Team 27									
Team 28									
Team 29									
Team 30									

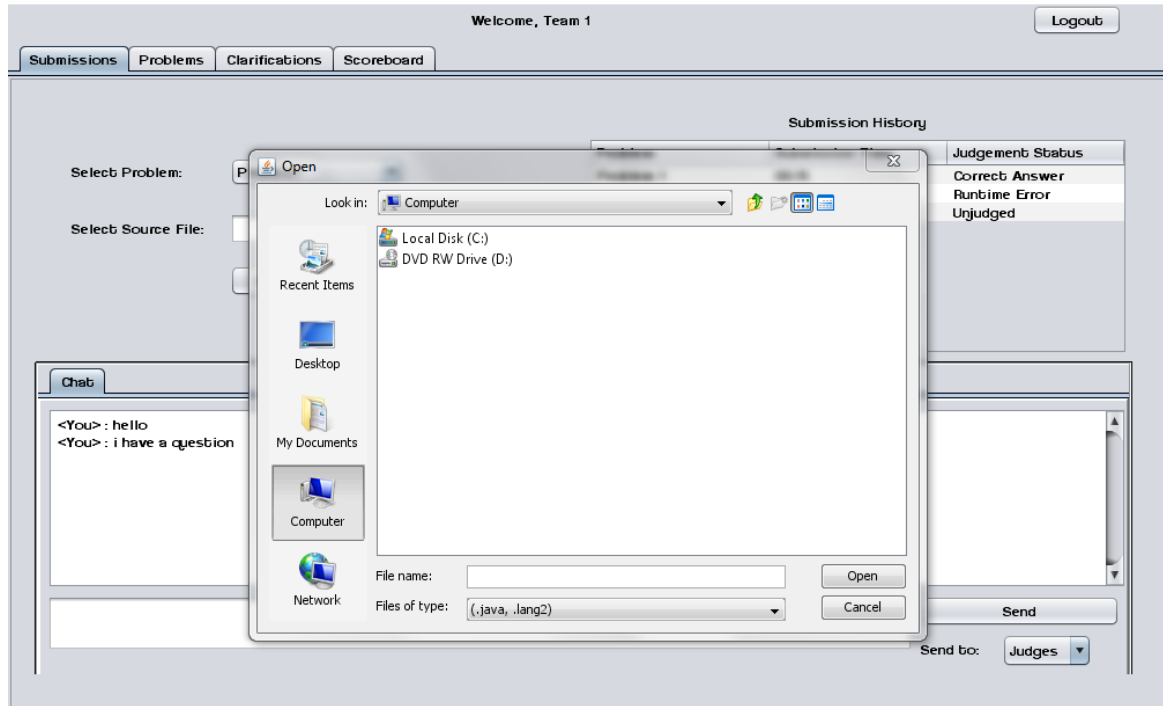
1.1.3.4 Submission Screen

The Submissions Screen allows a Contestant to chat with the Judges, submit Problem Submissions, and view submission history.

Submission History		
Problem	Submission Time	Judgement Status
Problem 1	00:15	Correct Answer
Problem 2	00:25	Runtime Error
Problem 3	00:26	Unjudged

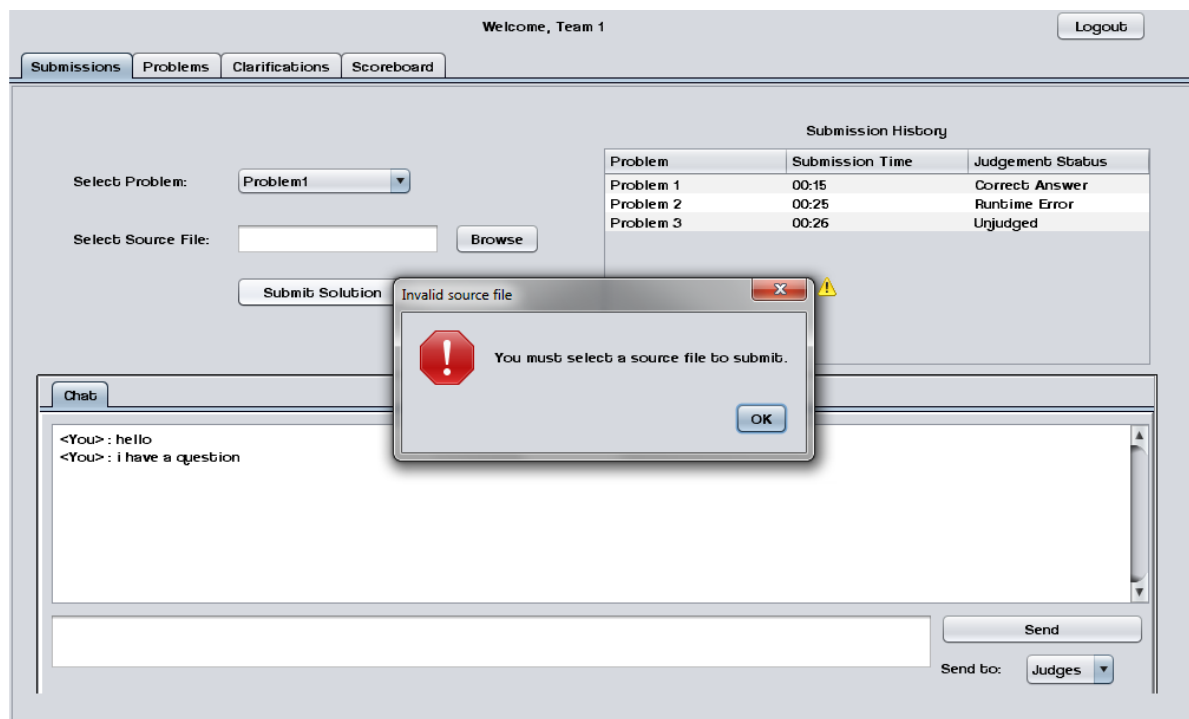
1.1.3.5 Submission Screen - Browse for source file

Contestants use a dialog window to browse for source files.



1.1.3.6 Submission Screen - No file selected error

An error message appears when a Contestant attempts to send a Problem Submission without selecting a source file.



1.1.4 Judge Screens

1.1.4.1 Create Clarification

Judges can create and send out Clarifications to all Contestants, Judges, and Spectators.

The screenshot shows the 'Create Clarification' screen. At the top, there is a navigation bar with buttons for 'Judge Submissions', 'Scoreboard', 'Problems', 'Create Clarification', and 'View Clarifications'. The 'Create Clarification' button is active. Below the navigation bar, there is a 'Welcome, Judge1' message and a 'Logout' button. The main content area features a 'Selected Problem:' dropdown menu currently set to 'Problem 1'. Below this is a large text input field labeled 'Problem 1 clarification'. At the bottom center, there is a 'Create Clarification' button.

1.1.4.2 View Clarifications

Judges can view all Clarifications that have been sent out by Judges.

The screenshot shows the 'View Clarifications' screen. At the top, there is a navigation bar with buttons for 'Judge Submissions', 'Scoreboard', 'Problems', 'Create Clarification', and 'View Clarifications'. The 'View Clarifications' button is active. Below the navigation bar, there is a 'Welcome, Judge1' message and a 'Logout' button. The main content area is divided into two sections. On the left, there is a table titled 'Clarifications' with two columns: 'Problem' and 'Clarification Number'. The table contains three rows: 'Problem 1' with '1', 'Problem 2' with '1', and 'Problem 1' with '2'. The last row is highlighted. Below the table is a 'View Clarification' button. On the right, there is a section titled 'Viewing Clarification: Problem 1, Number 2' with a large text input field labeled 'Problem 1 clarification'.

Problem	Clarification Number
Problem 1	1
Problem 2	1
Problem 1	2

1.1.4.3 Judge submissions

The Judge Submissions screen allows a Judge to chat with Contestants and other Judges. This screen also allows a Judge to view and judge Problem Submissions sent by Contestants.

The screenshot shows the 'Judge Submissions' interface. At the top, it says 'Welcome, Judge1' and has a 'Logout' button. Below this are navigation tabs: 'Judge Submissions', 'Scoreboard', 'Problems', 'Create Clarification', and 'View Clarifications'. The 'Judge Submissions' tab is active. It displays 'Selected Submission: Problem 1, team1, Number 1'. There are buttons for 'View Submission Output', 'View Test Input', and 'View Test Output'. Below these is a 'Selected Judgement:' dropdown menu set to 'Correct Answer' and a 'Judge Submission' button. To the right is a table with the following data:

Problem	Team	Number	CACTUS Response	Judgement
Problem 1	team1	1	Correct Answer	
Problem 1	team2	1	Wrong Answer	

At the bottom, there is a 'Chat' section with a large text area, a 'Send' button, and a 'Send to:' dropdown menu set to 'Judges'.

1.1.4.4 Problems List

Judges may select a contest problem to view in an embedded scroll area.

The screenshot shows the 'Problems' tab selected in the navigation menu. It displays 'Welcome, Judge1' and a 'Logout' button. Below the navigation tabs, there is a 'Select Problem:' dropdown menu set to 'Problem 3'. Below this is a large scrollable area containing the text 'Problem 3 document.'.

1.1.4.5 Scoreboard View

Judges may view the contest scoreboard. The scoreboard must be manually refreshed.

Team	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7	Penalty Time	Total Time
Team 1	00:05	00:11							00:16
Team 2	00:15	00:10							00:25
Team 3			00:15						00:15
Team 4			00:30						00:30
Team 5							00:05		00:05
Team 6									
Team 7									
Team 8									
Team 9									
Team 10									
Team 11									
Team 12									
Team 13									
Team 14									
Team 15									
Team 16									
Team 17									
Team 18									
Team 19									
Team 20									
Team 21									
Team 22									
Team 23									
Team 24									
Team 25									
Team 26									
Team 27									
Team 28									
Team 29									
Team 30									

1.1.5 System Administrator Screens

1.1.5.1 Change Admin password

The System Administrator may change the System Administrator Account password.

Old Password:

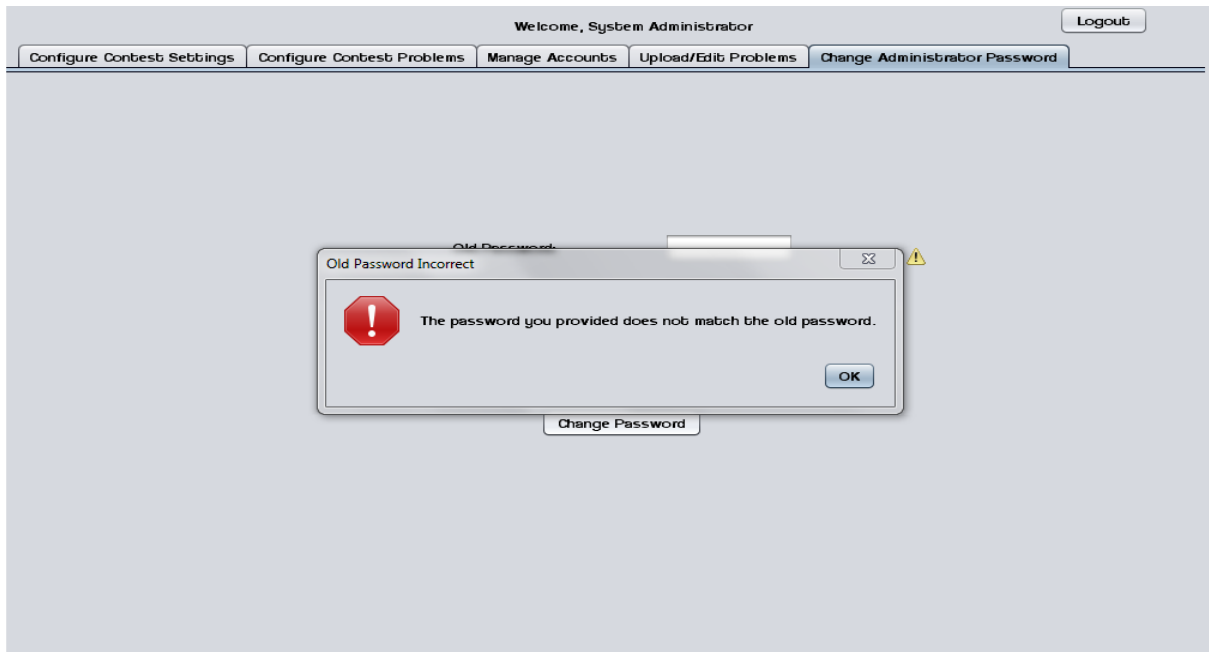
New Password:

Confirm New Password:

Change Password

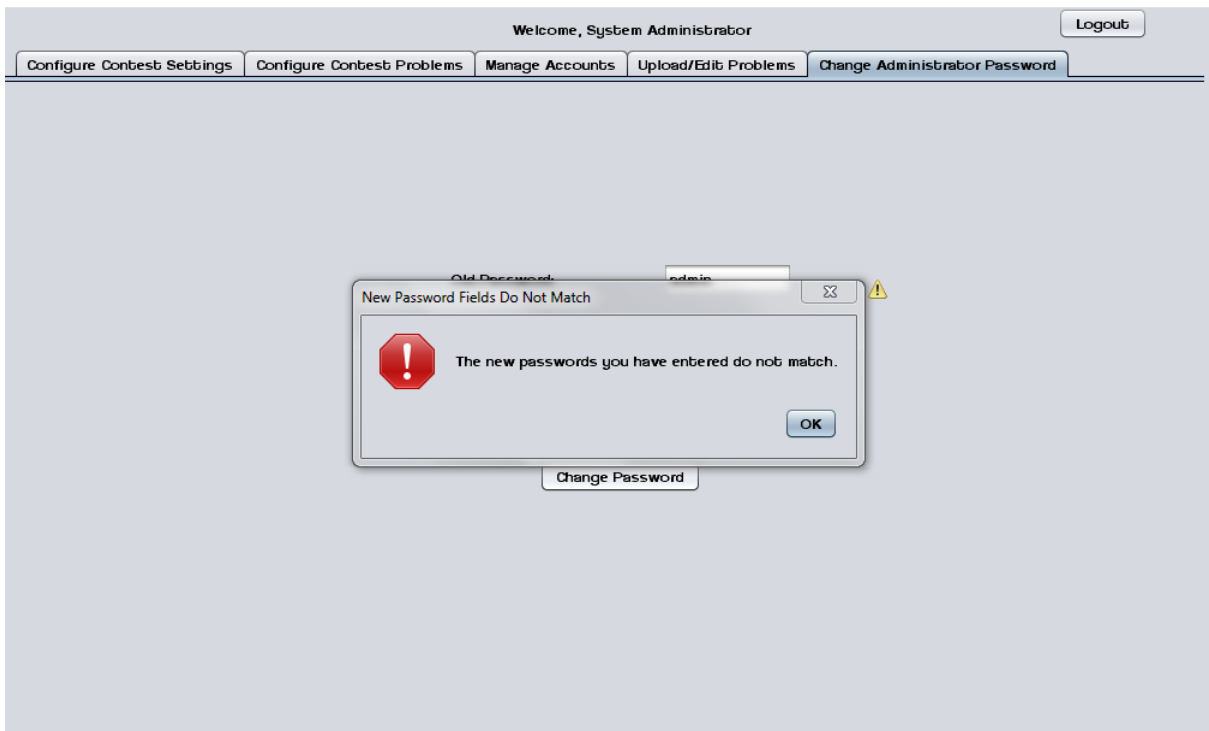
1.1.5.2 Change Admin password – Invalid old password error

This error message appears when the System Administrator enters a password that does not match the old one.



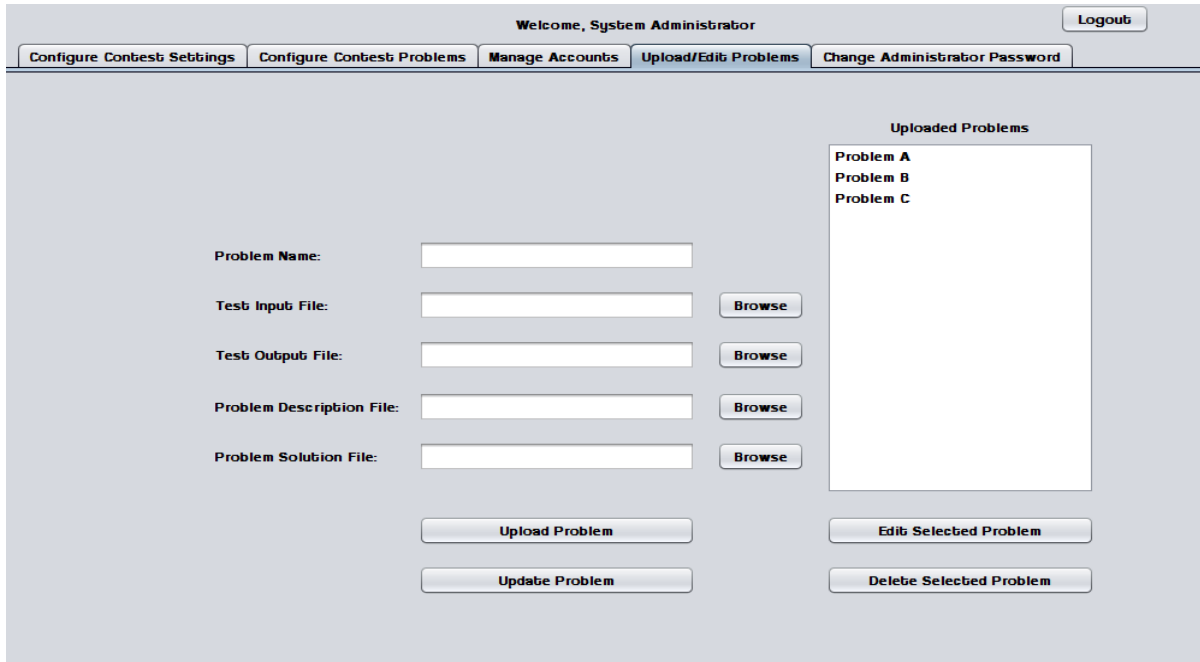
1.1.5.3 Change Admin password – New passwords do not match error

This error message appears when the System Administrator enters two different passwords in the "new password" fields.



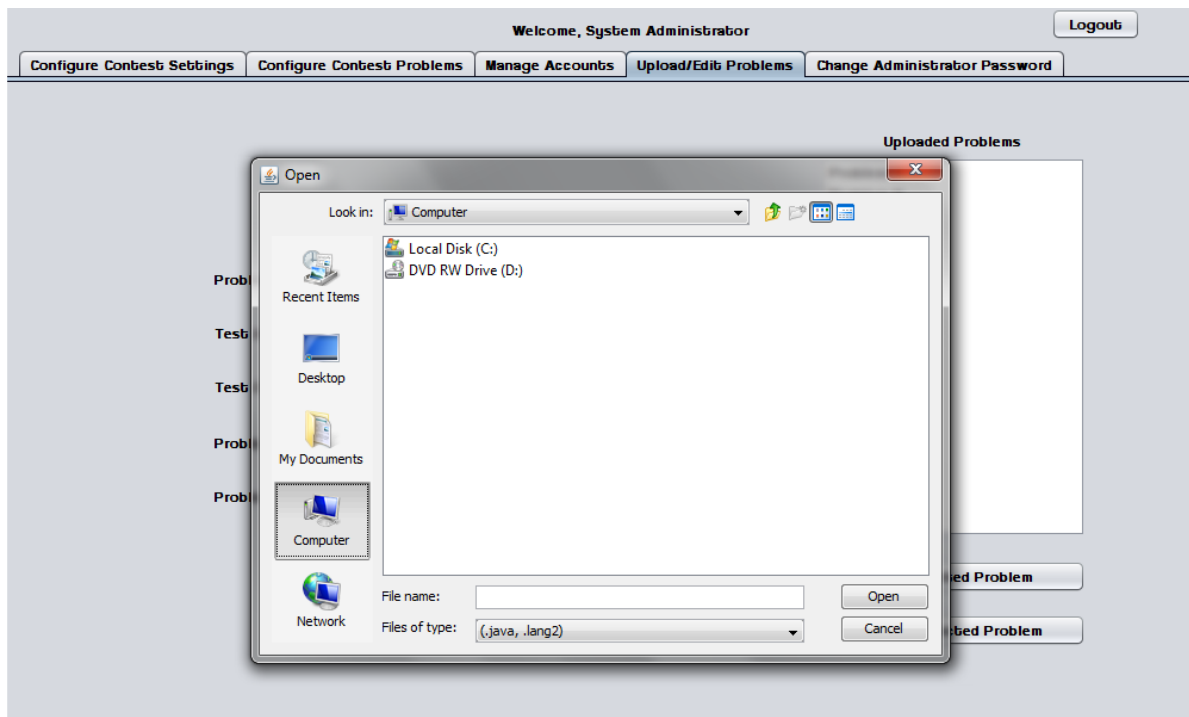
1.1.5.4 Upload/ Edit problems

The System Administrator can upload and edit contest problems into a persistent database.



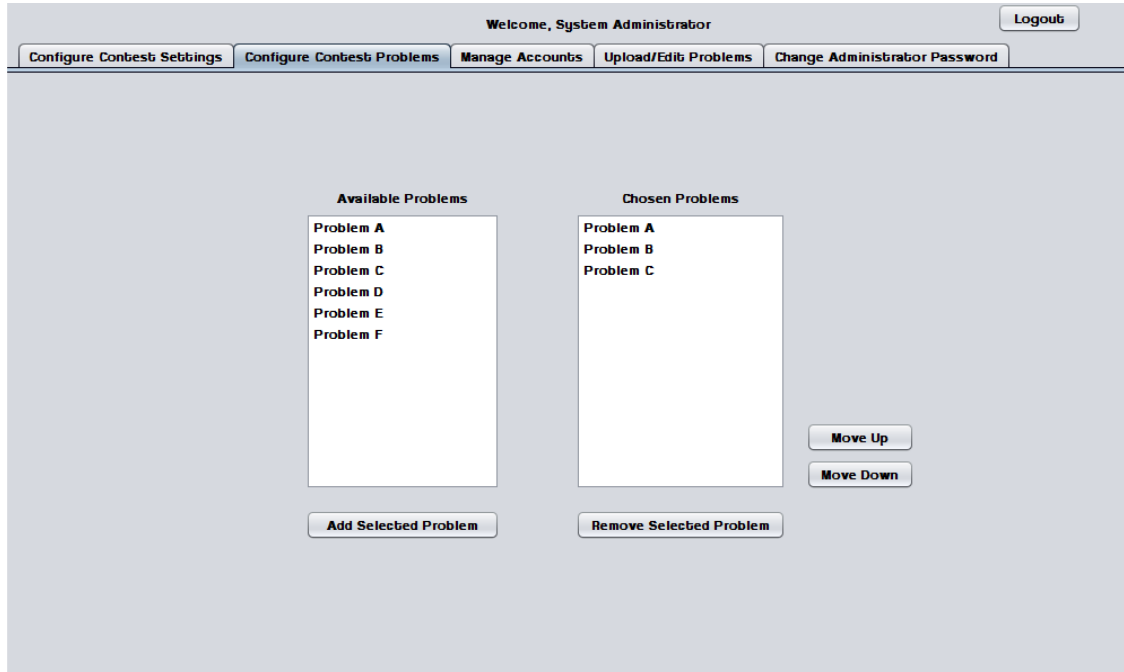
1.1.5.5 Upload Problems - Browse for file

The System Administrator uses a dialog window to browse for test input, test output, problem description, and problem solution files.



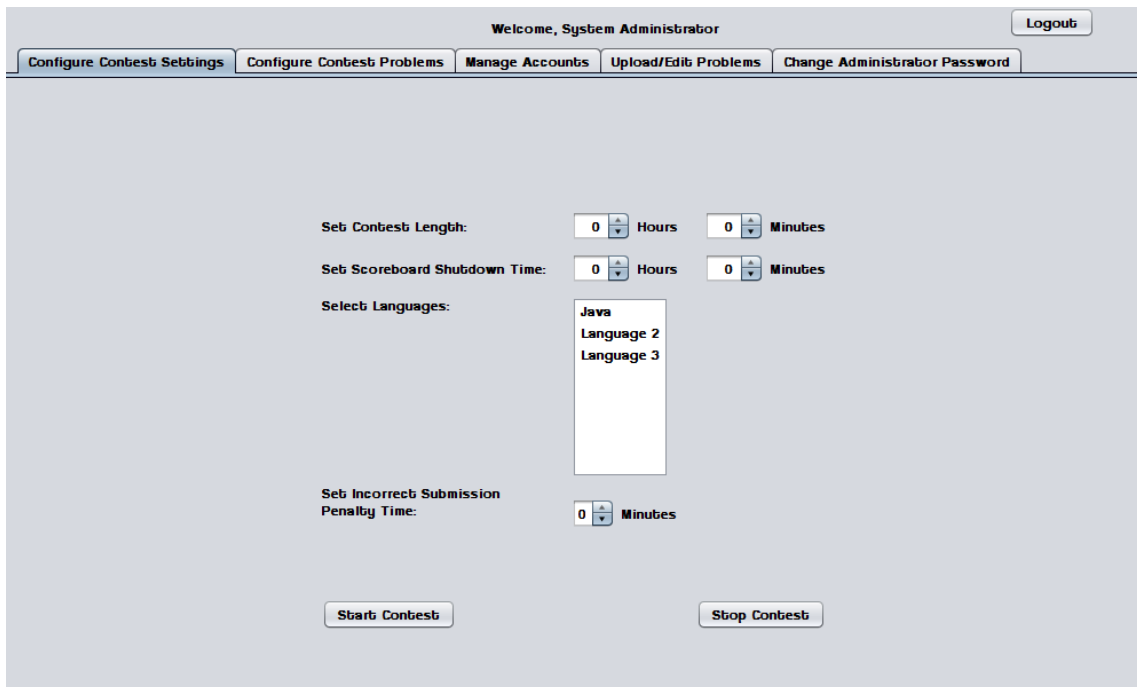
1.1.5.6 Configure Contest Problems

The System Administrator may select a number of problems to use in a contest from a list of available/uploaded problems.



1.1.5.7 Configure Contest Settings

The System Administrator may configure contest settings such as duration, scoreboard freeze time, and acceptable contest languages. The contest may be started and stopped from this screen.



1.1.5.8 Configure Contest Settings – Contest started

When a contest is started many settings cannot be changed until the contest ends or it is stopped by the System Administrator.



1.1.5.9 Invalid Contest Length Error

This error message appears when the System Administrator tries to start a contest with a duration of less than one minute.



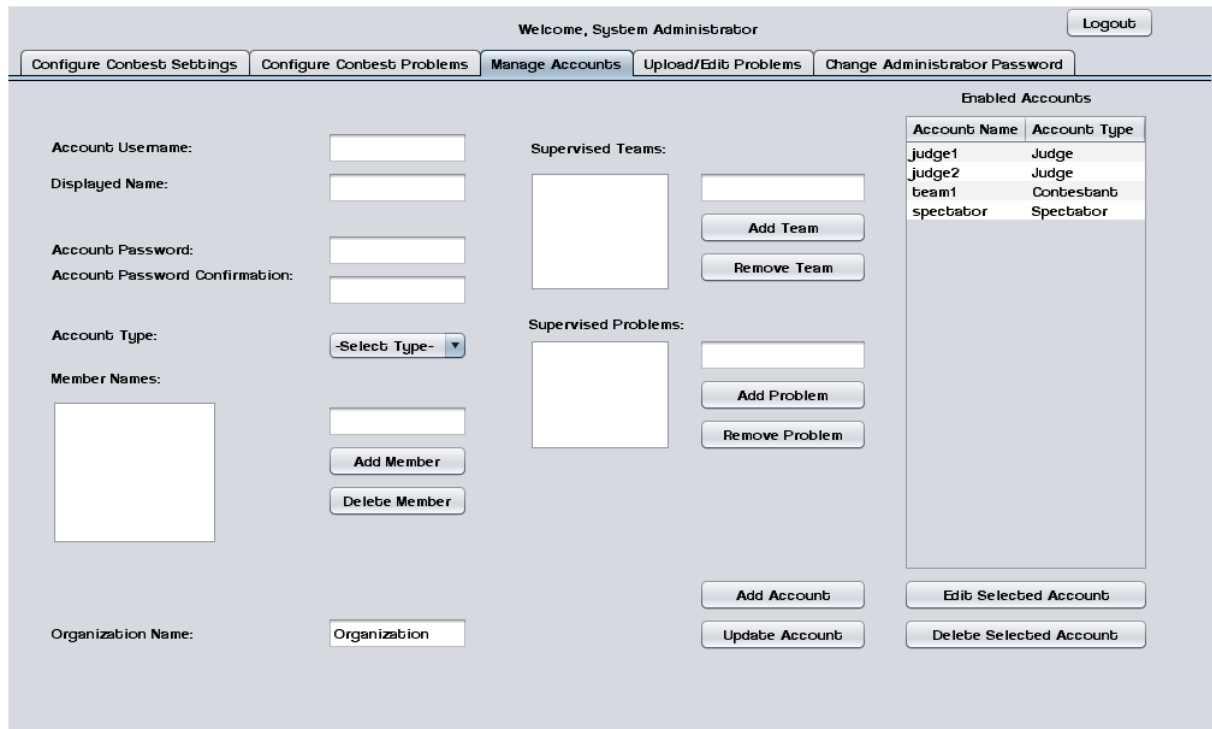
1.1.5.10 Invalid Selection Error

This error message appears when the System Administrator tries to start a contest without any languages selected.



1.1.5.11 Manage Accounts

This screen shows the general interface that the System Administrator uses to manage user accounts.



1.1.5.12 Manage Accounts – Contestants

This screen shows the System Administrator managing a Contestant account.

Contestants may have multiple human member names and an organization/school name.

The screenshot shows the 'Manage Accounts' interface for a Contestant account. At the top, there is a navigation bar with 'Welcome, System Administrator' and a 'Logout' button. Below the navigation bar are tabs for 'Configure Contest Settings', 'Configure Contest Problems', 'Manage Accounts', 'Upload/Edit Problems', and 'Change Administrator Password'. The main content area is divided into several sections:

- Account Information:** Fields for 'Account Username' (beam2), 'Displayed Name' (Team 2), 'Account Password' (b2pass), and 'Account Password Confirmation' (b2pass). The 'Account Type' is set to 'Contestant'.
- Member Names:** A list of names (David, Stephanie, Marco, Thomas, Paul) and buttons for 'Add Member' and 'Delete Member'.
- Supervised Teams:** An empty box with 'Add Team' and 'Remove Team' buttons.
- Supervised Problems:** An empty box with 'Add Problem' and 'Remove Problem' buttons.
- Organization Name:** A field containing 'ExoNET Solutions'.
- Enabled Accounts:** A table listing existing accounts:

Account Name	Account Type
judge1	Judge
judge2	Judge
beam1	Contestant
spectator	Spectator

At the bottom right, there are buttons for 'Add Account', 'Update Account', 'Edit Selected Account', and 'Delete Selected Account'.

1.1.5.13 Manage Accounts – Judges

This screen shows the System Administrator managing a Judge account. Judges may supervise submissions from specific teams and/or supervise specific problems.

The screenshot shows the 'Manage Accounts' interface for a Judge account. The layout is similar to the Contestant account page, but with the following differences:

- Account Information:** 'Account Username' is judge3, 'Displayed Name' is Judge 3, and 'Account Type' is Judge.
- Member Names:** The list is empty.
- Supervised Teams:** The box contains 'beam1'.
- Supervised Problems:** The box contains 'Problem 7'.
- Enabled Accounts:** The table lists the same accounts as the Contestant page:

Account Name	Account Type
judge1	Judge
judge2	Judge
beam1	Contestant
spectator	Spectator

The bottom right buttons are the same as in the Contestant page.

1.1.5.14 Manage Accounts – Spectator

This screen shows the System Administrator managing a Spectator account.

Welcome, System Administrator Logout

Configure Contest Settings | Configure Contest Problems | **Manage Accounts** | Upload/Edit Problems | Change Administrator Password

Account Username: Supervised Teams:

Displayed Name:

Account Password: Supervised Problems:

Account Password Confirmation:

Account Type:

Member Names:

Account Name	Account Type
judge1	Judge
judge2	Judge
team1	Contestant
spectator	Spectator

Organization Name:

1.1.6 Spectator Screens

1.1.6.1 Problems List

Spectators may select a contest problem to view in an embedded scroll area.

Welcome, Spectator Logout

Scoreboard | **Problems** | Clarifications

Select Problem:

Problem 1 document.

1.1.6.2 Scoreboard View

Spectators may view the contest scoreboard. The scoreboard must be manually refreshed.

Team	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5	Problem 6	Problem 7	Penalty Time	Total Time
Team 1	00:05	00:11							00:16
Team 2	00:15	00:10							00:25
Team 3			00:15						00:15
Team 4			00:30						00:30
Team 5								00:05	00:05
Team 6									
Team 7									
Team 8									
Team 9									
Team 10									
Team 11									
Team 12									
Team 13									
Team 14									
Team 15									
Team 16									
Team 17									
Team 18									
Team 19									
Team 20									
Team 21									
Team 22									
Team 23									
Team 24									
Team 25									
Team 26									
Team 27									
Team 28									
Team 29									
Team 30									

1.1.6.3 View Clarifications

Spectators can view all Clarifications that have been sent out by the Judges.

Clarifications	
Problem	Clarification Number
Problem 1	1
Problem 2	1
Problem 1	2

Viewing Clarification: Problem 1, Number 1

Problem 1 clarification

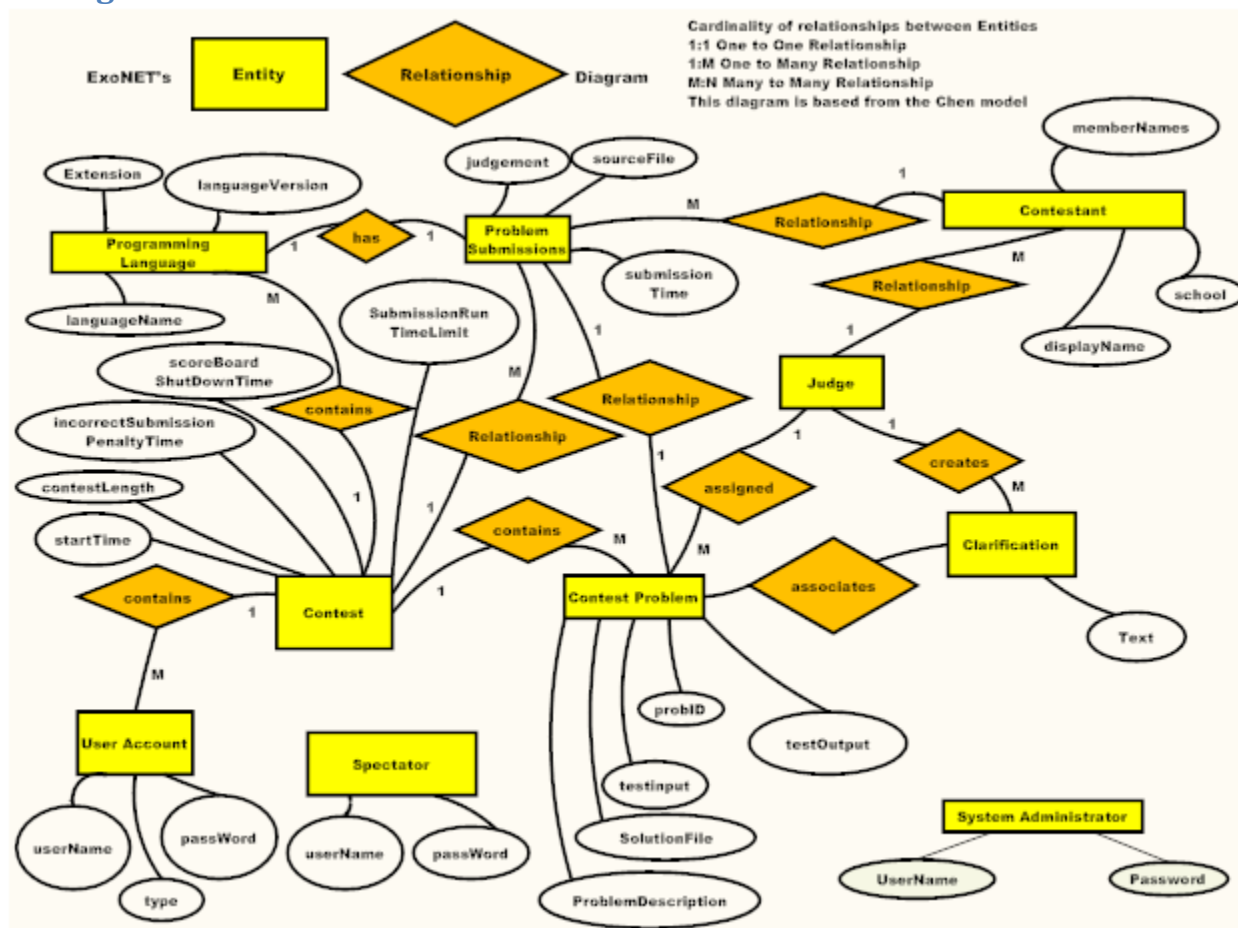
1.2 Logical Data Dictionary

As defined by IBM, a data dictionary is “as centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format.”

ExoNET’s data dictionary contains: What user this data type is applicable to, the data entity’s name, what the data entity is applicable to in C.A.C.T.U.S., the data type of the entity (such as String, int, etc.), the size of the data type, the data type’s description, the acceptable input format for this data type, examples of good input for this data type, examples of bad input for this data type, and notes regarding this data type (if any are necessary).

Please see Appendix C: Logical Data Dictionary

1.3 Logical Format of Data Files



Architectural Design Specification

1.1 Development and Production Environments

Server:

- Operating System: CentOS (Linux) Release 5.2 (Final)
- Server Name: oraserv.cs.siena.edu
- CPU Type: x86_64
- Web Server: Apache Version 2.2.9
- PHP Version: 5.2.6
- Database: MySQL Version 5.0.45; Oracle Version 9.0.1

SE Lab hardware/software specs:

- Dell ACP x86-based PC
- Intel® Core™ 2 Duo CPU E7500 @ 2.93 Ghz
- Operating System: Windows Vista Enterprise
- Memory: 305.1 GB of total space 258.6 GB free space
- Ram: 4.00 GB
- Network Adapters: Intel(r) 82567LM-3 Gigabit Network Connection
- Display Chip: Intel (R) 4 series Internal Chipset 2.93 GHz
- Browsers: Mozilla FireFox 4.0.1; Internet Explorer 9; Google Chrome; Macromedia Flash, Macromedia Dreamweaver

Marco's hardware/software specs:

- HDD: 600GB
- Display Adapter: NVIDIA GeForce GTX 285
- DVD/CD ROM:
TSSTcorp CDDVDW SH-S22A SCSI CdRom Device (DVD/CD burner)
ZGFKPUJ 12JKXIR SCSI CdRom Device (DVD/CD burner)
- Logitech HID-Compliant Keyboard
- Logitech HID-Compliant G5 Laser Mouse
- HP 2159 Series Wide LCD Monitor
- Processor: Intel Core i7 CPU 920 @ 2.67 GHz
- Audio: SoundMAX Integrated Digital HD Audio
- Software: Bluej, Netbeans, Microsoft Office, Google Chrome, KompoZer

David's hardware/software specs:

- Operating System: Windows 7 Home Premium 64-bit (6.1, Build 7601)
System Model: H55M-S2V
- Processor: Intel(R) Core(TM) i3 CPU
- Memory: 4096MB RAM
- Speed: 4 CPUs @ 3.2GHz

- Gimp 2.6
- Paint.NET 3.5.8
- Audacity 1.3
- Netbeans 7.0.1
- Notepad++ 5.9.3
- Google Chrome 14.0.835.163
- Mozilla Firefox 6.0.2
- WinSCP 4.2.9
- PuTTY 0.60

Programming languages used:

- Java
- SQL
- XHTML
- CSS

1.2 Deliverables

ExoNET Solutions will have all the necessary files and features on a CD-ROM/DVD. C.A.C.T.U.S. will be accessible to use locally and remotely on oraserv.cs.siena.edu/~perm_exonet/launch.html

A CD-ROM will be delivered containing:

- All ExoNET Team Files - Documents, Test Plan, Appendices
- Files associating with C.A.C.T.U.S.
- ExoNET's Team Song with audio/video file, music/sound file, and lyrics
- A README.txt file explaining what files are located where

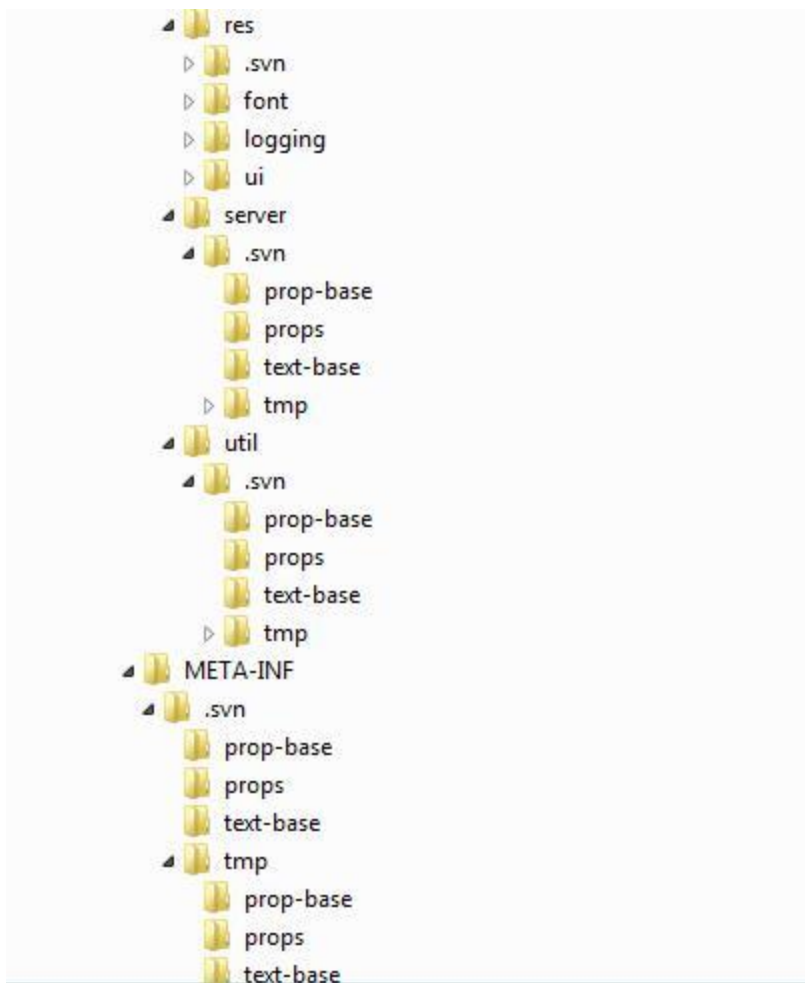
1.3 Data Flow Diagrams (See Appendix A: Data Flow Diagrams)

The purpose of the Data Flow Diagram is view the flow of data and transformations of data throughout the system. This helps viewers see what is happening within the system as it is running. The Data Flow diagrams are organized into specific levels of the system. The context diagram is a general overview of the entire system. The level 0 diagram shows in more detail how each entity interacts with the system through a process. The level 1 diagrams show an in-depth look at each process within the system. In all diagrams arrows are drawn between entities, databases and processes to show the movement of data.

1.4 Source Code

The following is a complete directory listing of all files created in the development process of the C.A.C.T.U.S. project.





Test Requirements and Results

1.1 Explanation of Test Plan

A Test Plan documents the strategy that will be used to verify and ensure that a product meets its design specifications, as well as Functional and Non-functional requirements. The testing will be a multi-step process that consists of activities for validating the software product, from the most primitive elements up to the fully integrated system. This area includes activities such as unit testing, integration testing, system testing, performance testing, and acceptance testing.

Our strategy consists of dividing the project up into separate modules, each with a distinct functionality. Each module has a Unit Test script that is followed, moving test case by test case to ensure that all parts of the module are thoroughly tested. In order for each unit to be considered 'passed' each test case in the unit must pass.

1.2 Test Results (See Appendix B: Test Requirements and Results)

Glossary of Terms

Actor: An entity in UML Use Case Diagrams and UML Activity Diagrams. It represents the human and non-human external entities (outside the system boundary) that interact with the system.

Apache HTTP Server (Web Server) - Referred to as Apache, it is web server software notable for playing a key role in the initial growth of the World Wide Web.

C++ - (pronounced "see plus plus") is a [statically typed](#), [free-form](#), [multi-paradigm](#), [compiled](#), general-purpose [programming language](#). It is regarded as an intermediate-level language, as it comprises a combination of both [high-level](#) and [low-level](#) language features.

CACTUS - Java Open Language Toolkit definition project -

A project aimed at providing integrated system for computer programming contests hosted at Siena College.

Cascading Style Sheets (CSS) - A style sheet language used to describe the presentation semantics (the look and formatting) of a document written in a markup language.

Chrome – An Internet browser designed by Google.

Conflict – When an activity can't be scheduled due to room use, weekend, and one resource being currently in use.

Constraint – When the client specifies that a certain resource must be used, or the project has to be done in a certain way, using certain specifications.

Data Flow Diagram - A graphical representation of the "flow" of data through an [information system](#)

Data Flows: A component of a Data Flow Diagram (DFD) that represents the movement of data from an External Entity to a Process or Data Store, and vice versa.

Data Stores: A component of a Data Flow Diagram(DFD) that represents any location in which information or data is stored.

Database - An organized collection of data for one or more uses, typically in digital form.

Dreamweaver – A web development application.

Dropbox - A Web-based file hosting service operated by Dropbox, Inc. which uses cloud computing to enable users to store and share files and folders with others across the Internet using file synchronization.

External Entities: A component of a Data Flow Diagram that represents any human or non-human user of a Software System.

Entity-Relational (ER) Diagram - an abstract and conceptual representation of [data](#); used to represent the logical format of a database

Firefox – An Internet browser designed by Mozilla.

Functional Requirements Inventory: Define what the system will be able to do and what is testable about the system.

Gantt Chart - 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.

HTML (HyperText Markup Language) - The predominant markup language for web pages. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content. It is the building blocks of all basic websites.

HTTP (Hypertext Transfer Protocol) - a protocol used to transfer hypertext requests and information between servers and browsers.

Internet - A global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a *network of networks* that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic and optical networking technologies.

Internet Explorer (IE) – An Internet browser designed by Microsoft.

Java - a high-level, object-oriented computer programming language used especially to create interactive applications running over the Internet.

JavaScript - An implementation of the ECMAScript language standard and is typically used to enable programmatic access to computational objects within a host environment.

MySQL - A relational database management system that runs as a server providing multi-user access to a number of databases.

PHP (PHP Hypertext Preprocessor) - A widely used, general-purpose “server side” scripting language that was originally designed for web development to produce dynamic web pages.

SQL- structured query language: a computer programming language used for database management

Software: The programs installed on the computer, such as Microsoft Office and Adobe Fireworks.

System: A component of UML Use Case Diagram which represents the Software System.

UML (Unified Modeling Language) Use Case Diagram - a type of behavioral diagram to present a graphical overview of the functionality provided by a system.

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: A component of a UML Use Case Diagram which represents any process located within the System that is performable by an Actor.

User Case Narrative: an explanation of the functions and abilities users have for a specific Software System.

Waterfall Model (Classic) - The Classic Waterfall Model is a sequential software development model in which development is seen as flowing steadily downwards (similar to a waterfall) through the phases of requirements analysis, design, implementation, testing, integration, and maintenance.

XHTML (eXtensible Hypertext Markup Language) - A family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML), the language in which web pages are written.

XML (Extensible Markup Language) - A set of rules for encoding documents in machine-readable form. To create a tagging scheme that allows elements of a document to be marked according to their content rather than their format.

Project Timeline

