

# **Detailed Design**

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

**February 29<sup>th</sup>, 2011**

**C.A.C.T.U.S.**

**Detailed Design**

# Table of Contents

- Review..... 4
  - 1.1 Problem Review ..... 4
  - 1.2 User Case Narratives ..... 5
  - 1.3 Data Flow Diagrams..... 6
    - 1.3.1 Data Flow Diagram Key..... 6
    - 1.3.2 Context Diagram ..... 7
    - 1.3.3 Level 0 Diagram ..... 8
    - 1.3.4 Process 1 Diagrams ..... 9
    - 1.3.5 Process 2 Diagrams ..... 12
    - 1.3.6 Process 3 Diagrams ..... 15
    - 1.3.7 Process 4 Diagrams ..... 18
    - 1.3.8 Process 5 Diagrams ..... 21
  - 1.4 Structure Diagram..... 24
  - 1.5 Functional Requirements Inventory ..... 25
- Detailed Design ..... 27
  - 1.1 Logical Data Dictionary ..... 27
    - 1.1.1 Logical Data Dictionary Key..... 27
    - 1.1.2 Logical Data Dictionary Spreadsheet ..... 28
  - 2.1 ER Diagram ..... 36
  - 3.1 Prototype Screens ..... 37
    - 3.1.1 Log in Screen..... 37
    - 3.1.2 Log in Failed..... 37
    - 3.1.3 Contestant Screens..... 38
      - 3.1.3.1 View Clarifications..... 38
      - 3.1.3.2 Problems List..... 38
      - 3.1.3.3 Scoreboard View ..... 39
      - 3.1.3.4 Submission Screen..... 39
      - 3.1.3.5 Submission Screen - Browse for source file..... 40
      - 3.1.3.6 Submission Screen – No file selected error ..... 40
    - 3.1.4 Judge Screens ..... 41
      - 3.1.4.1 Create Clarification..... 41
      - 3.1.4.2 View Clarifications..... 41
      - 3.1.4.3 Judge submissions ..... 42
      - 3.1.4.4 Problems List..... 42
      - 3.1.4.5 Scoreboard View ..... 43
    - 3.1.5 System Administrator Screens..... 43
      - 3.1.5.1 Change Admin password..... 43
      - 3.1.5.2 Change Admin password – Invalid old password error..... 44
      - 3.1.5.3 Change Admin password – New passwords do not match error ..... 44
      - 3.1.5.4 Upload/ Edit problems..... 45
      - 3.1.5.5 Upload Problems - Browse for file..... 45
      - 3.1.5.6 Configure Contest Problems..... 46
      - 3.1.5.7 Configure Contest Settings ..... 46

3.1.5.8 Configure Contest Settings – Contest started .....	47
3.1.5.9 Invalid Contest Length Error .....	47
3.1.5.10 Invalid Selection Error.....	48
3.1.5.11 Manage Accounts .....	48
3.1.5.12 Manage Accounts – Contestants.....	49
3.1.5.13 Manage Accounts – Judges .....	49
3.1.5.14 Manage Accounts – Spectator .....	50
3.1.6 Spectator Screens.....	50
3.1.6.1 Problems List.....	50
3.1.6.2 Scoreboard View .....	51
3.1.6.3 View Clarifications.....	51
4.1 Testing Plan .....	52
5.1 Development and Production Environments .....	52
6.1 Appendices .....	54
6.1.1 Appendix A: Glossary of Terms.....	54
6.1.2 Appendix B: Project Timeline (Gantt Chart).....	56
Testing Plan.....	57
1.1 Overview and Strategy .....	57
1.2 Test Plan Description.....	57
2.1 Pass/Fail Criteria .....	58
2.1.1 Functional Requirements Inventory .....	58
2.1.1.1 System Administrator .....	58
2.1.1.2 Judge.....	58
2.1.1.3 Contestant .....	59
2.1.1.4 Spectator .....	59
2.1.2 Non-Functional Requirements Inventory .....	59
2.1.3 Exception Handling .....	59
2.1.4 Testing Approach .....	60
2.1.5 Acceptance Criteria .....	60
3.1 Unit Tests.....	61
3.1.1 Unit Tests and Test Cases.....	61
3.1.2 Performance Test.....	61

# Review

## 1.1 Problem Review

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.

## 1.2 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.

5

### *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.

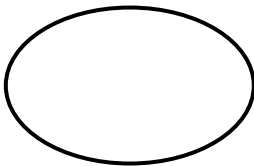
### 1.3 Data Flow Diagrams

The following legend should be used to interpret the Data Flow Diagrams in sections 5.1.2 to sections 5.1.8. ExoNET Solutions' Data Flow Diagrams have no system boundary and they do not follow some of the standard Data Flow Diagram rules. ExoNET Solutions' diagrams do not follow standard conventions for the sake of clarity and readability.

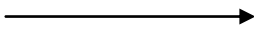
#### 1.3.1 Data Flow Diagram Key



**External Entity (Data Source/Sink)** – External Entities represent sources of data into the system or destinations of data sent by the system.



**Process** – Processes are system functions that transform or manipulate data. Processes receive and send out data.



**Data Flow** – Data flows represent the movement of data.



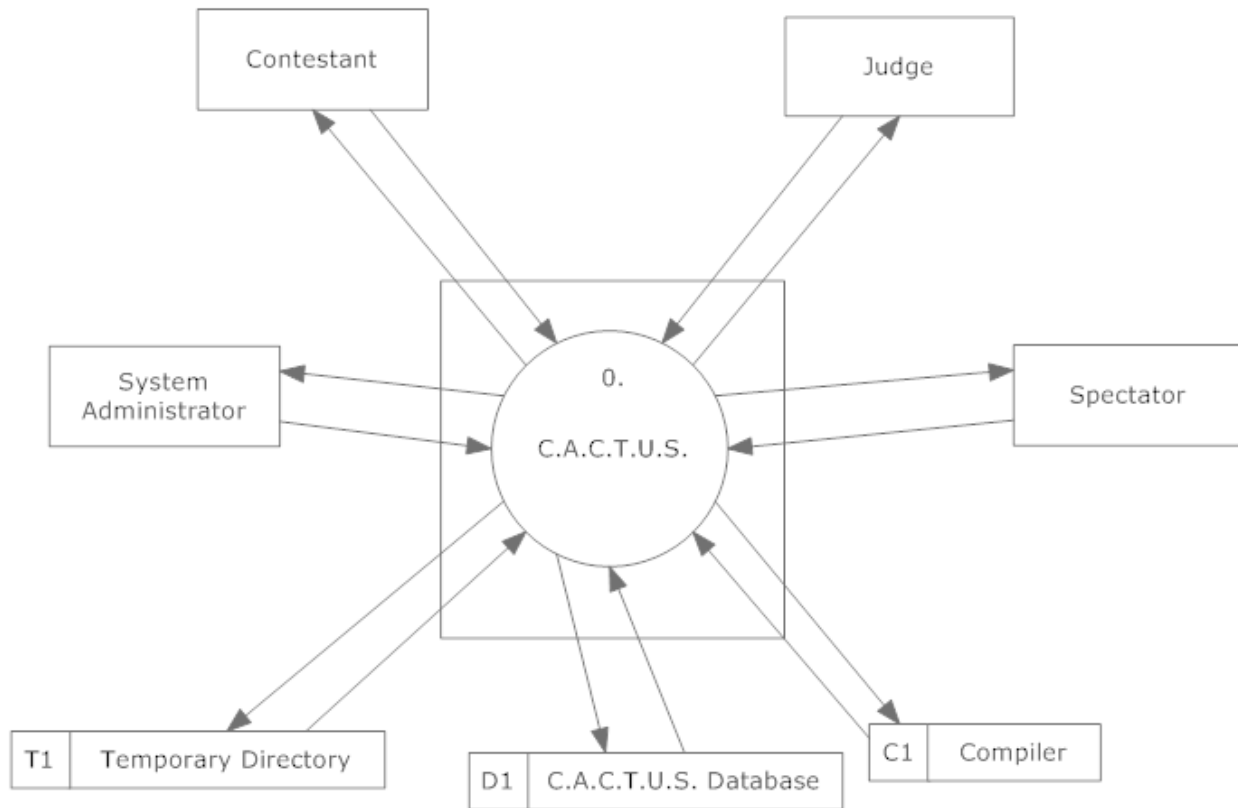
**Data Store** – Data stores contain persistent data that does not usually move. Data can be saved to and retrieved from Data Stores.

### 1.3.2 Context Diagram

A Context Diagram represents the actors outside a system that could interact with that system and shows the interaction between them.

**ExoNET Solutions**  
**Project C.A.C.T.U.S.**

### Context Diagram



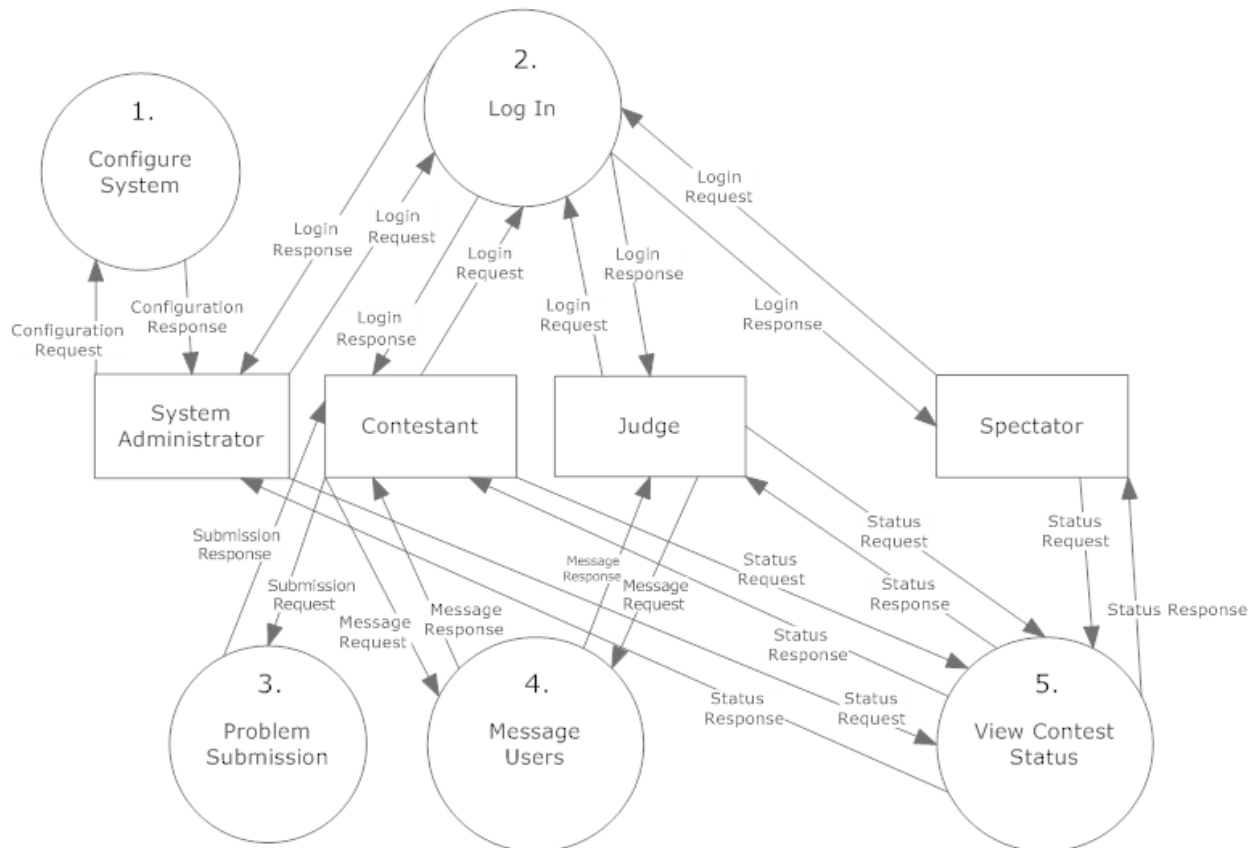
### 1.3.3 Level 0 Diagram

The Level 0 diagram goes into some detail of the system. It shows how the system is divided into sub-systems, or processes.

#### **ExoNET Solutions**

#### **Project C.A.C.T.U.S.**

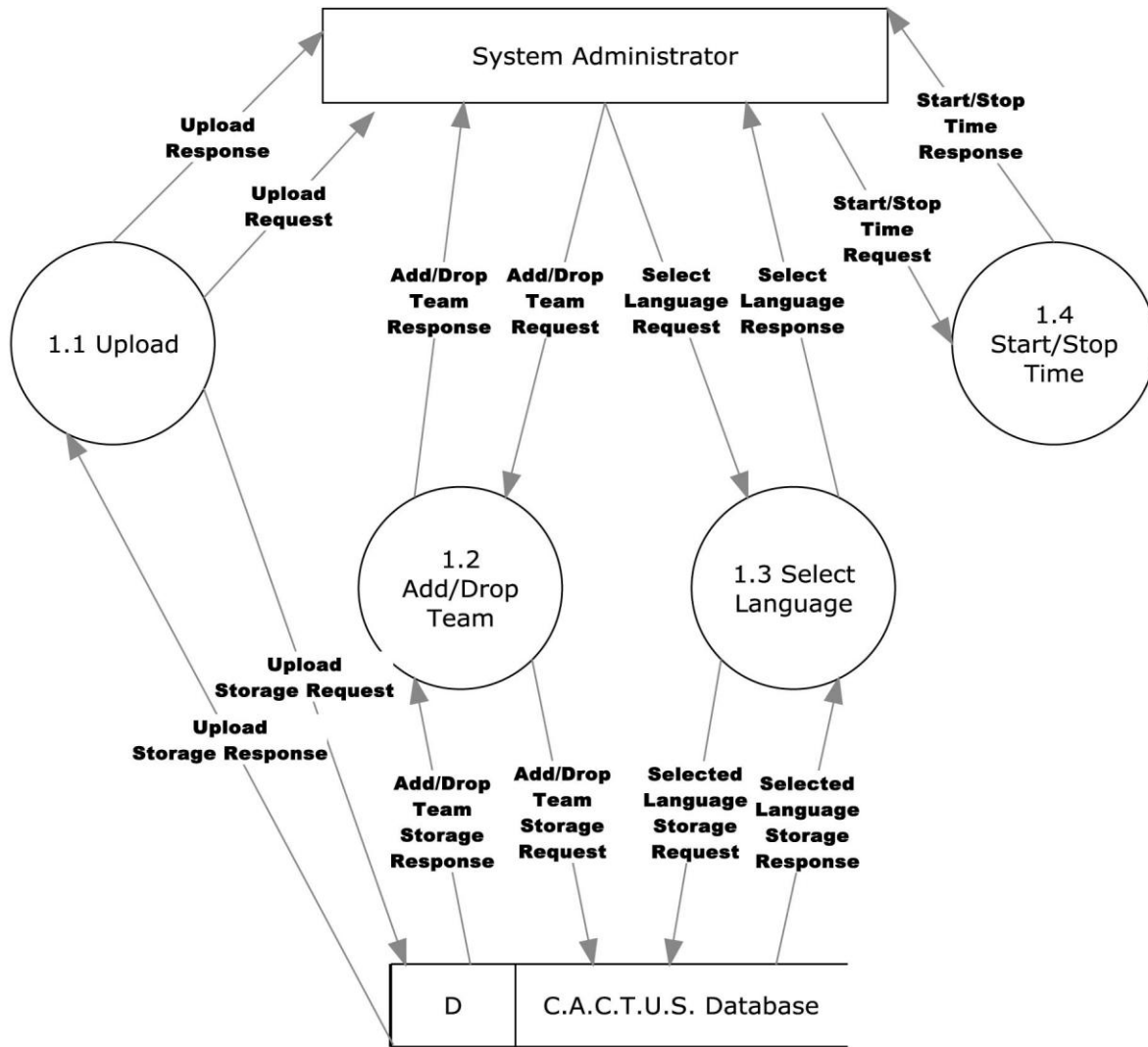
#### **Level 0 Data Flow Diagram**





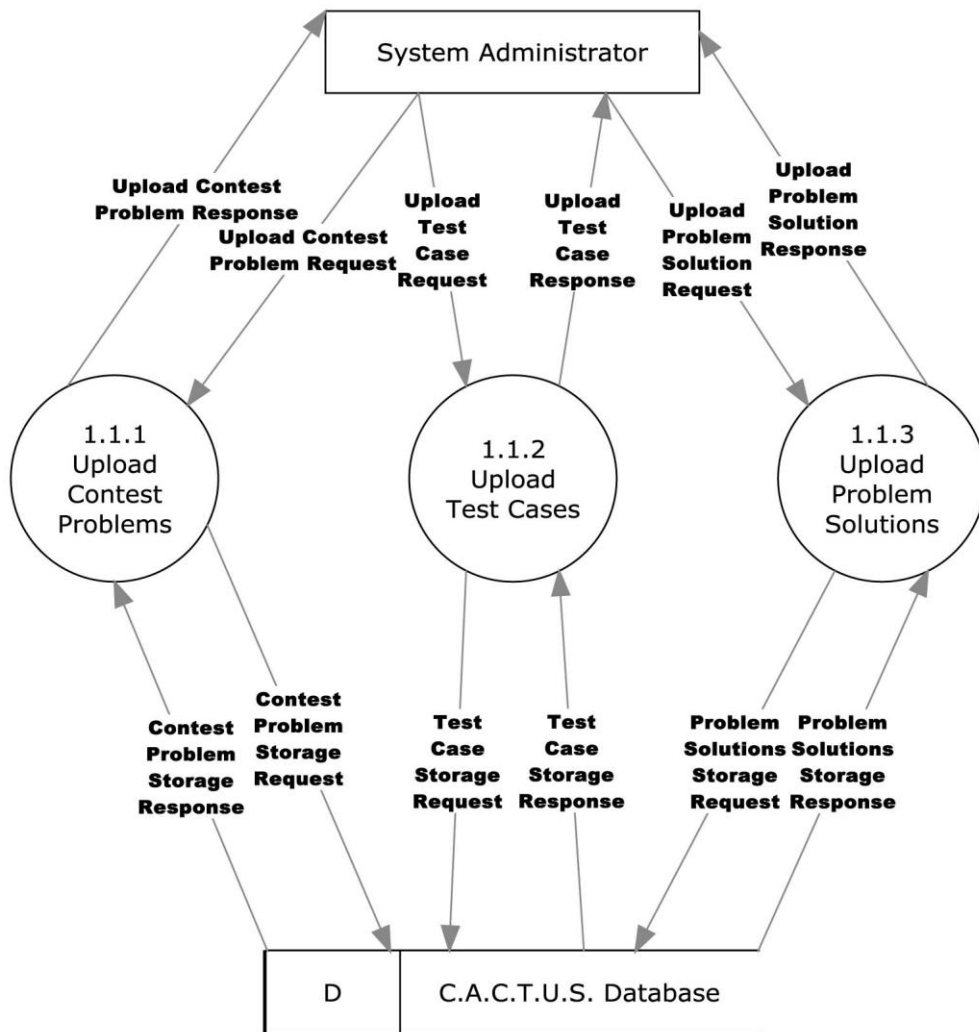
**Process 1, Level 1: Configure System**

This diagram demonstrates the processes that the System Administrator can perform when configuring the system.



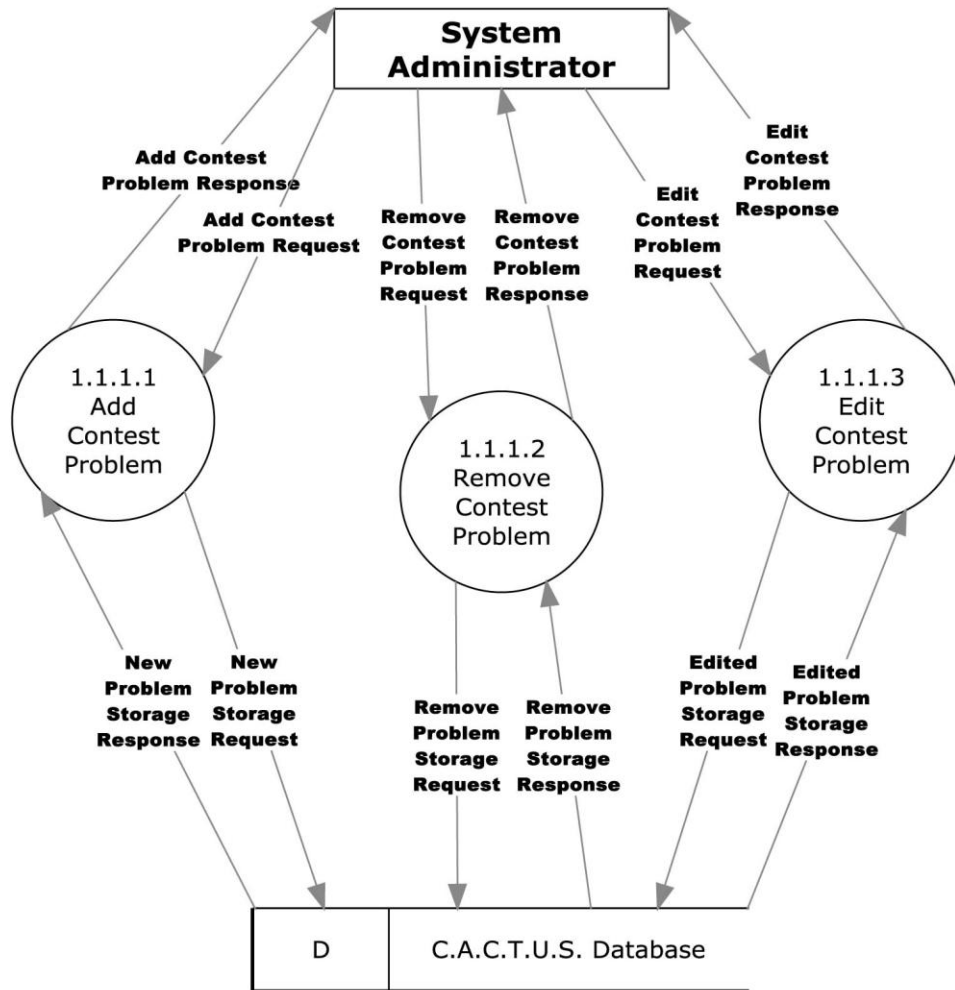
**Process 1.1, Level 2: Upload**

This diagram demonstrates how the different processes the System Administrator may perform when uploading.



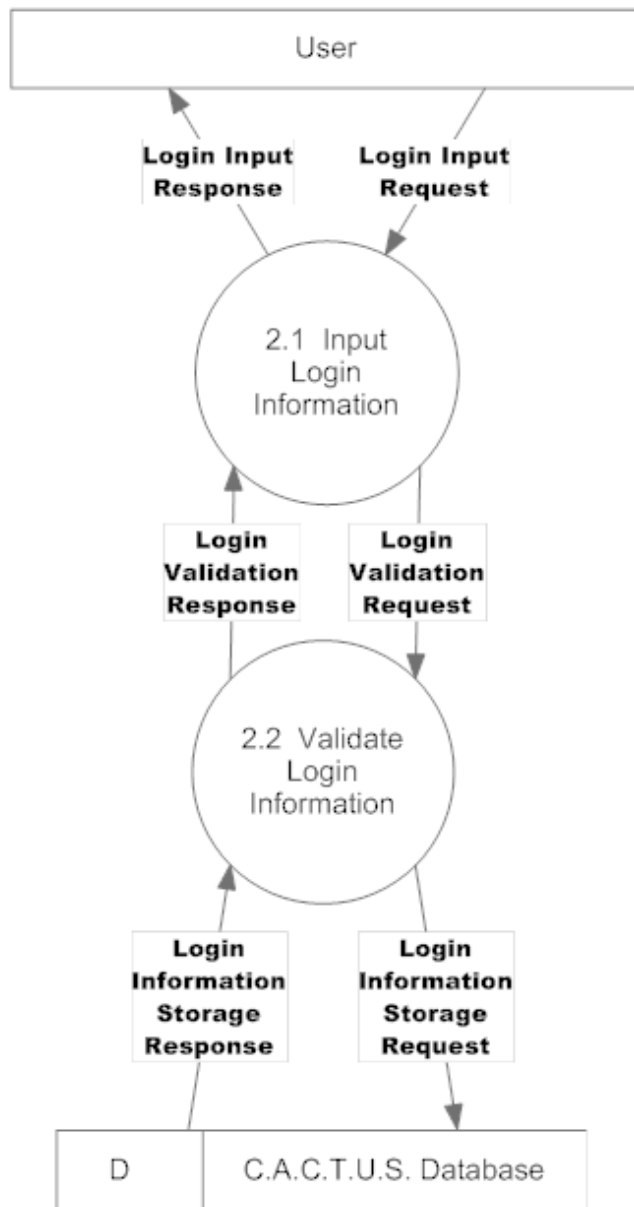
**Process 1.1.1, Level 3: Upload Contest Problems**

This diagram demonstrates the process the System Administrator can perform when uploading contest problems.



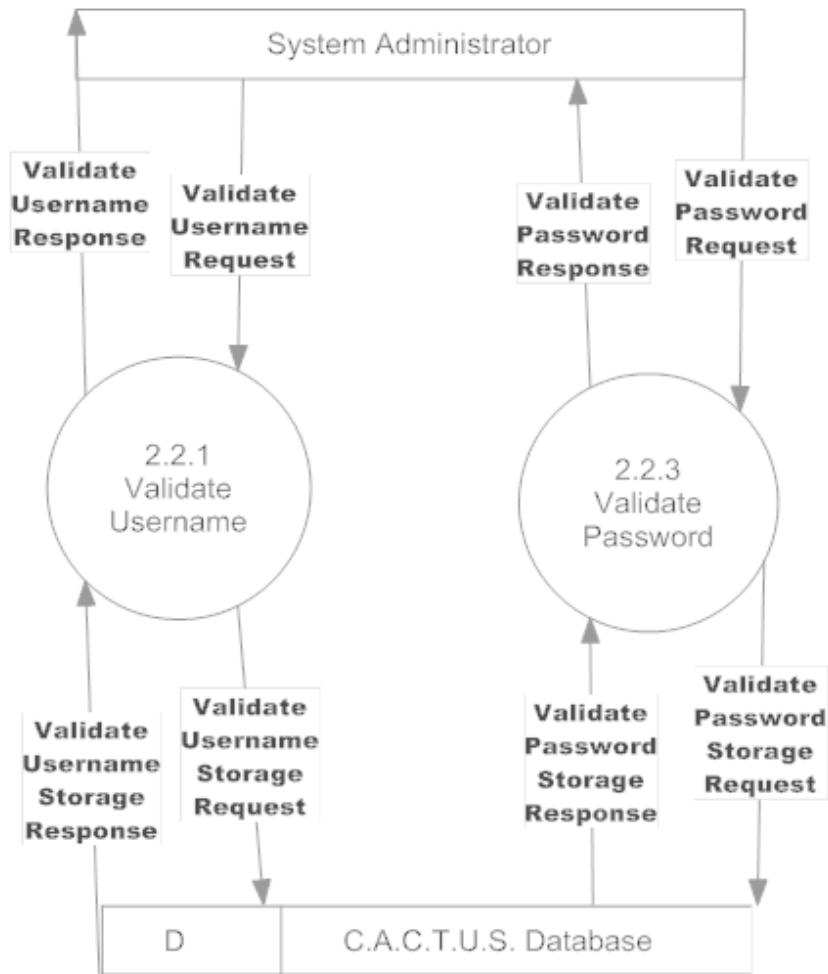
**Process 2, Level 1: Log In**

This diagram demonstrates the processes that any user can perform when logging into the system.



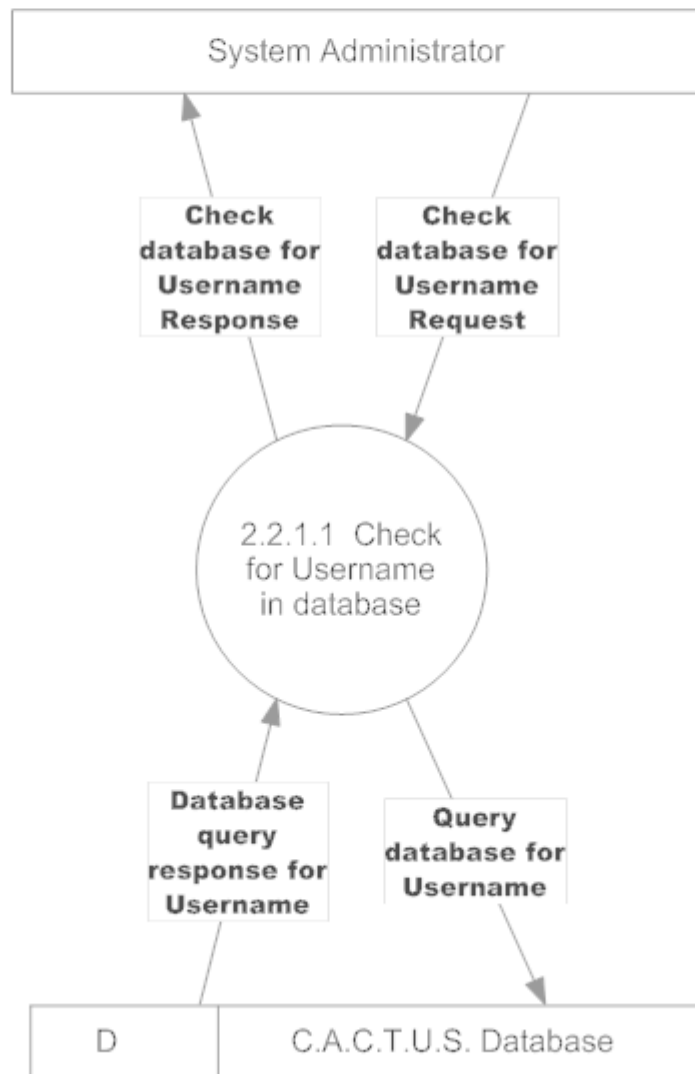
**Process 2.2, Level 2: Validate Login Information**

This diagram demonstrates the processes that the system performs when logging a user into the system.



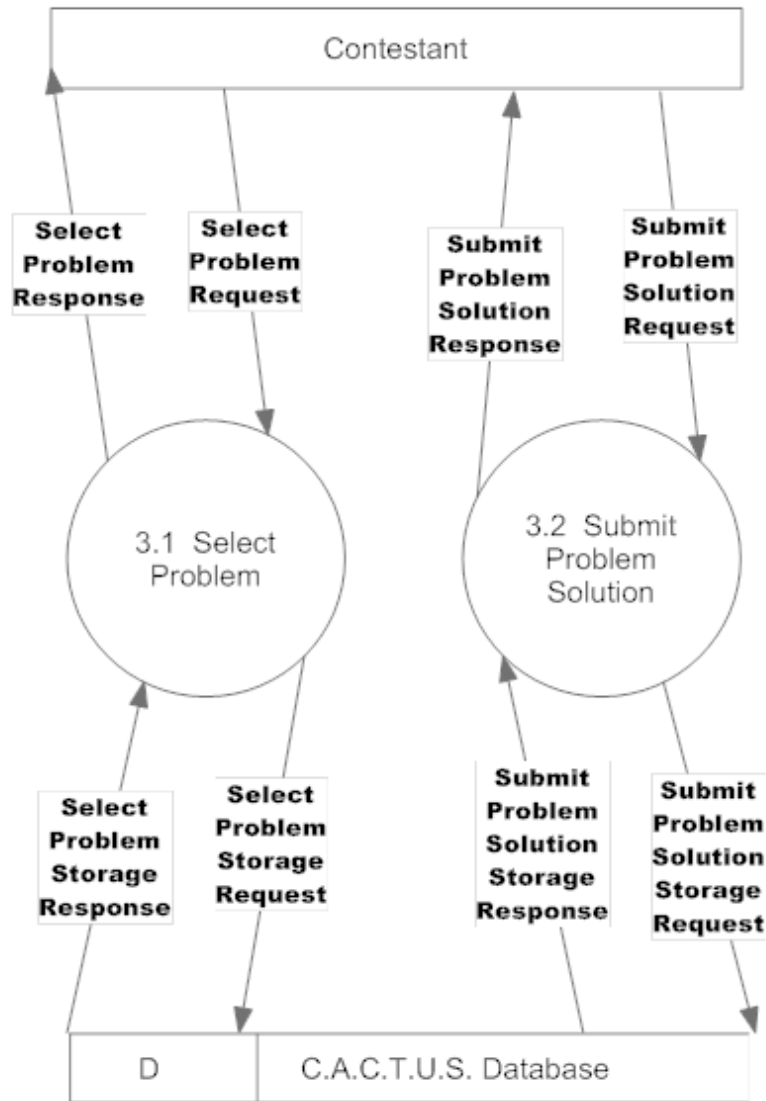
**Process 2.2.1, Level 3: Validate Username**

This diagram demonstrates the processes that the system performs when validating a user's Username.



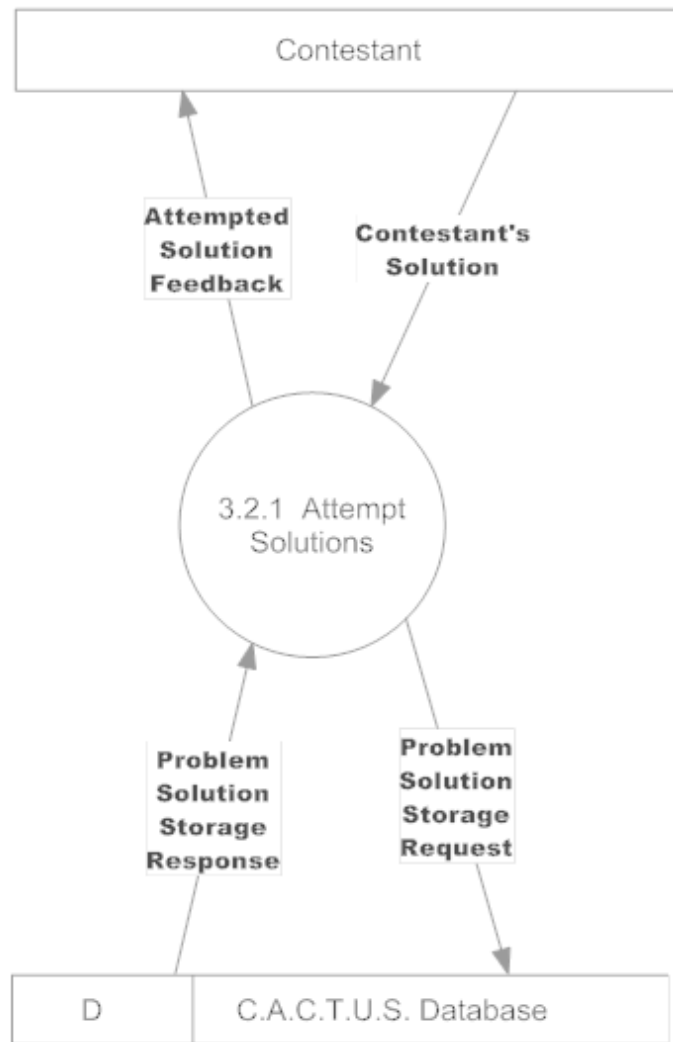
**Process 3, Level 1: Submit Problem Solution**

This diagram demonstrates the processes that the contestant performs when submitting a solution to a contest problem.



**Process 3.2, Level 2: Submit Solution**

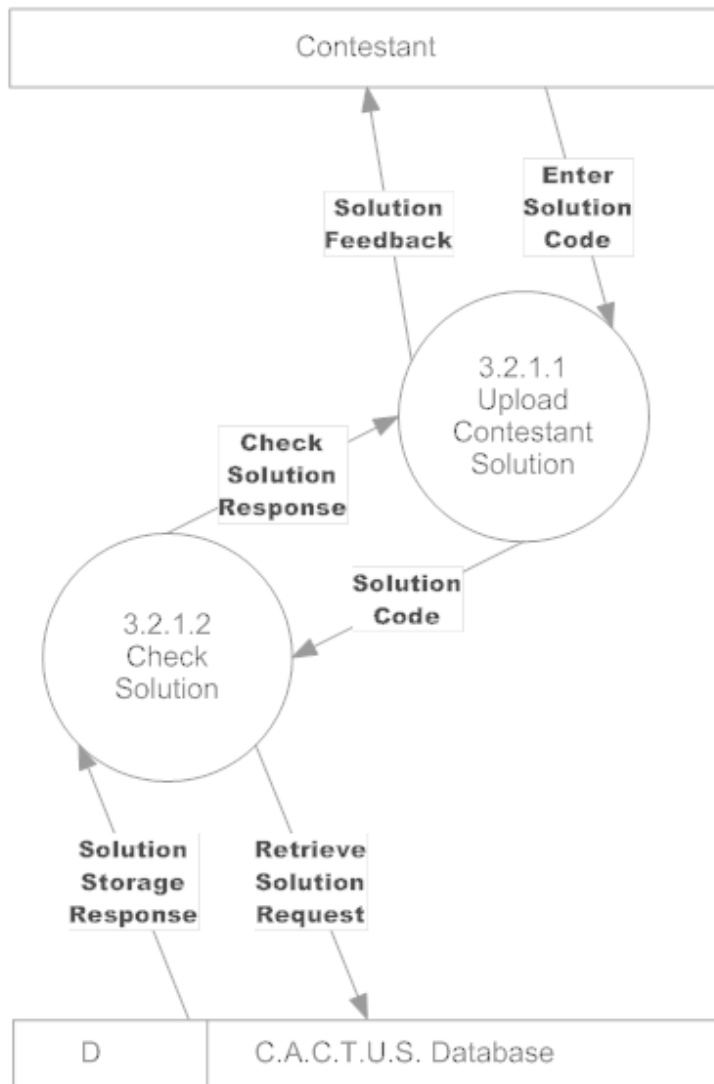
This diagram demonstrates the processes that the system performs when a user submits a solution to a contest problem.



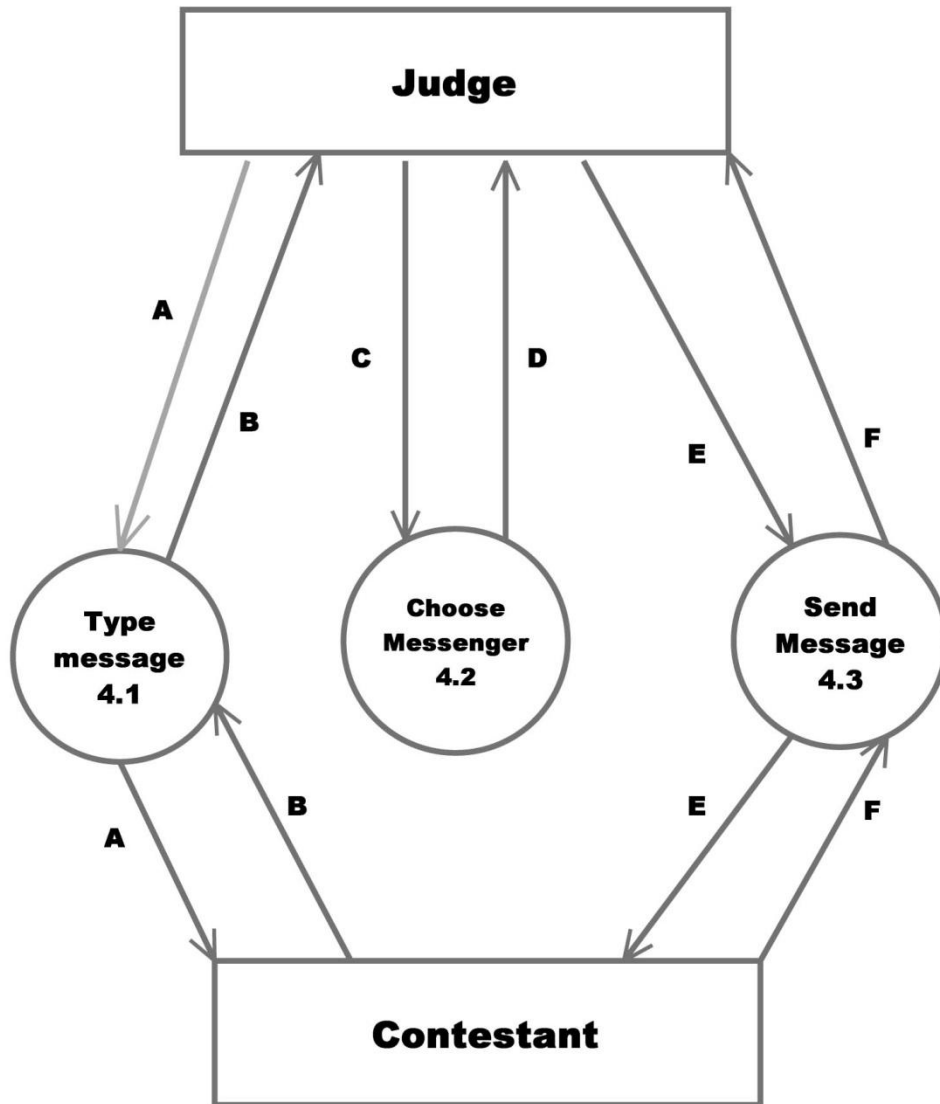


**Process 3.2.1, Level 3: Attempt Solutions**

This diagram demonstrates the processes that the system performs when a user attempts a solution to a contest problem.



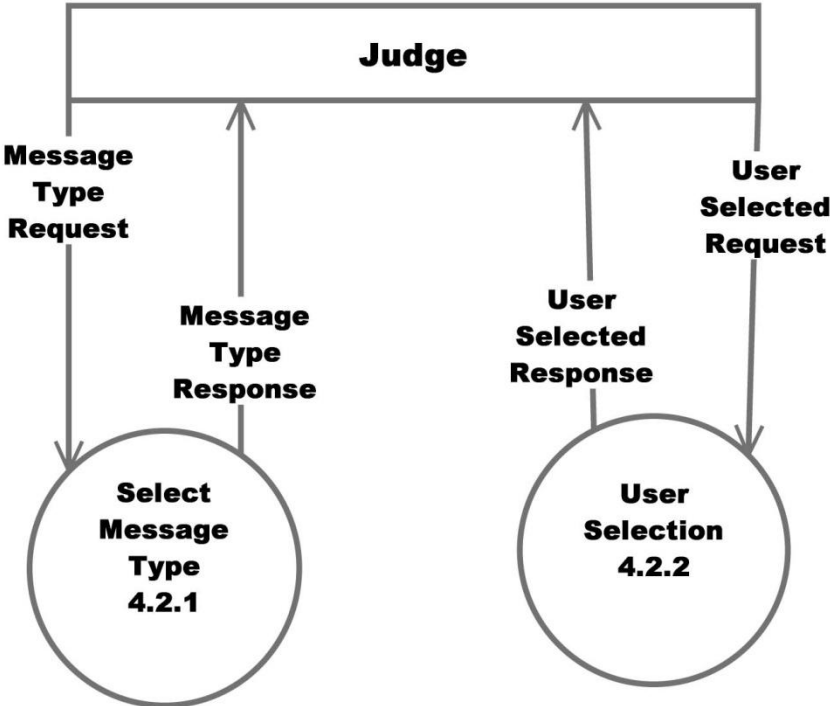
**Process 4, level 1: Message Users**



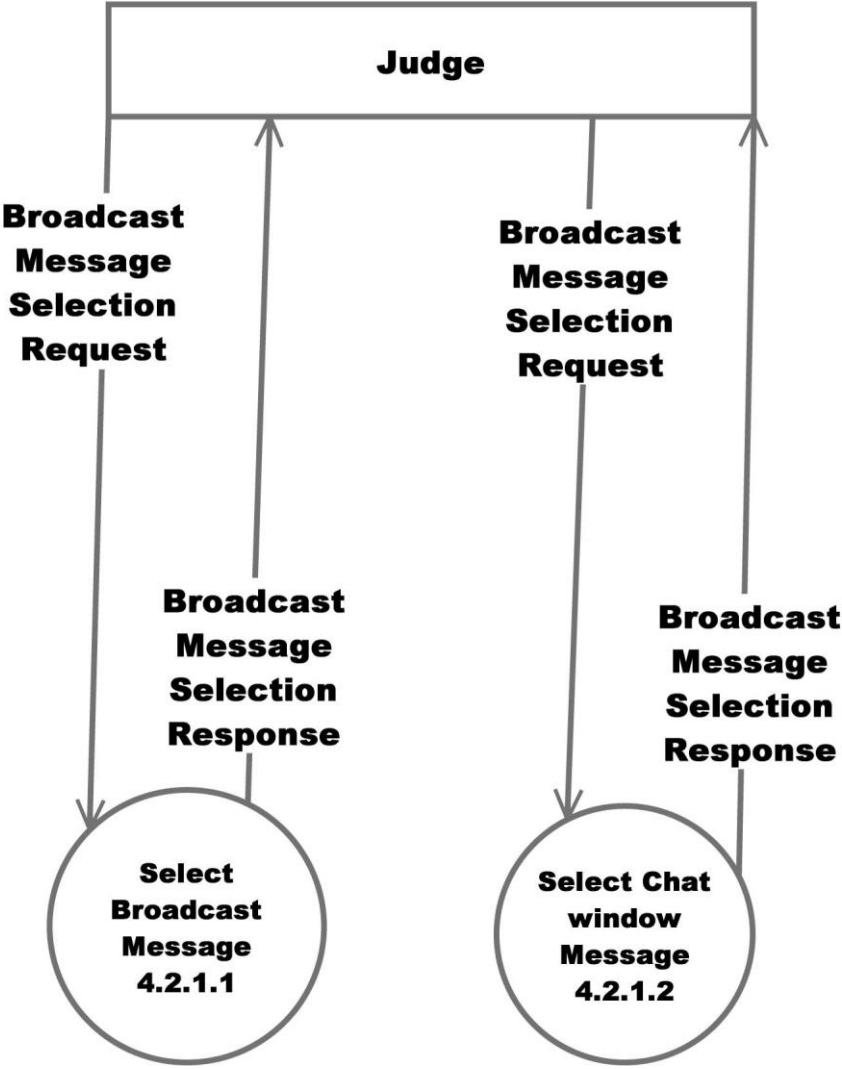
**Legend**

- |                                 |                               |                                 |
|---------------------------------|-------------------------------|---------------------------------|
| <b>A. Message Typed Request</b> | <b>C. Messenger Request</b>   | <b>E. Message Sent Request</b>  |
| <b>B. Message Response</b>      | <b>D. Messenger Activated</b> | <b>F. Message Sent Response</b> |

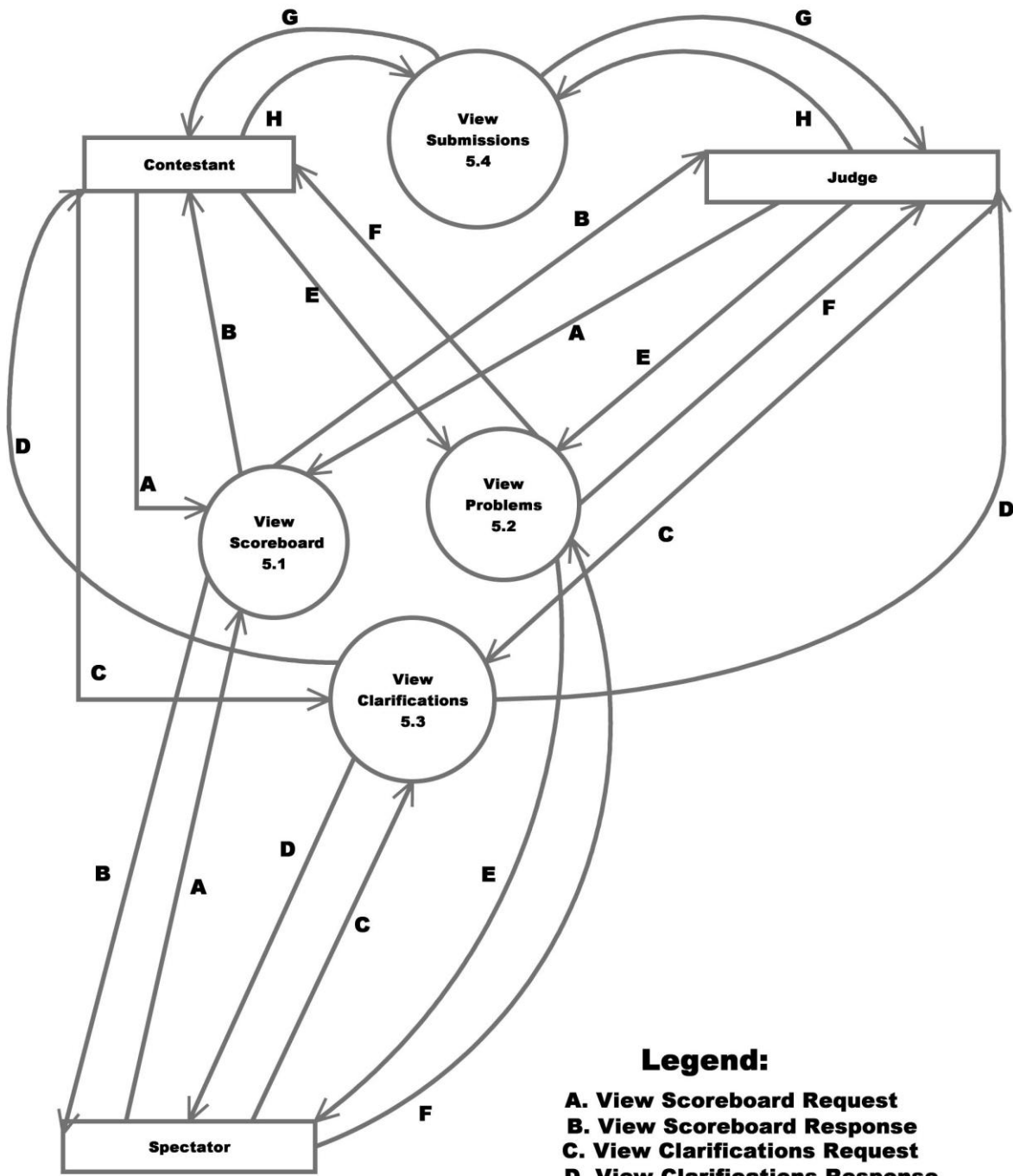
**Process 4.2, Level 2: Messenger Selection**



# Process 4.2.1, Level 3 Message Type Selection



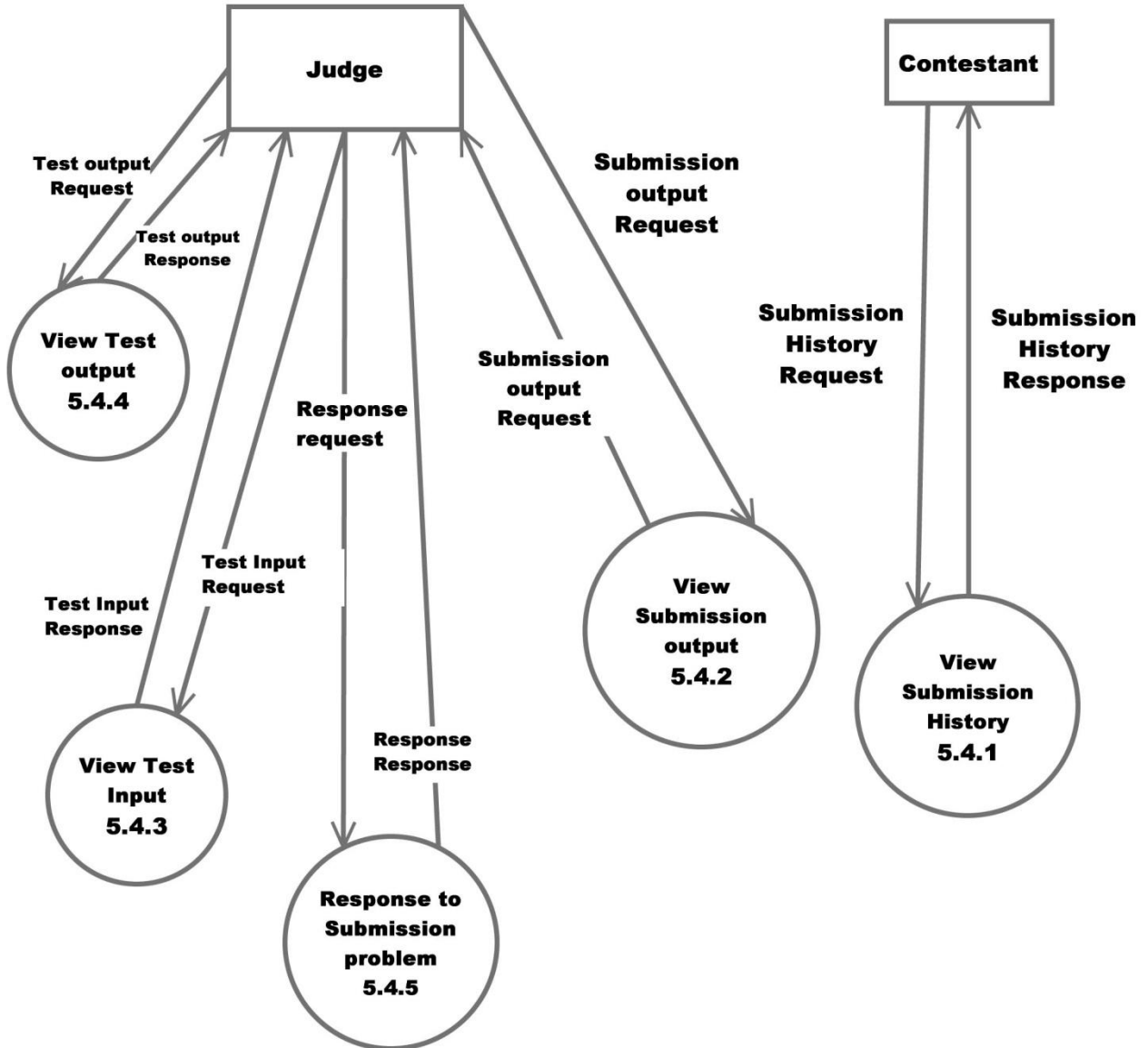
**Process 5, Level 1: View Contest Status**



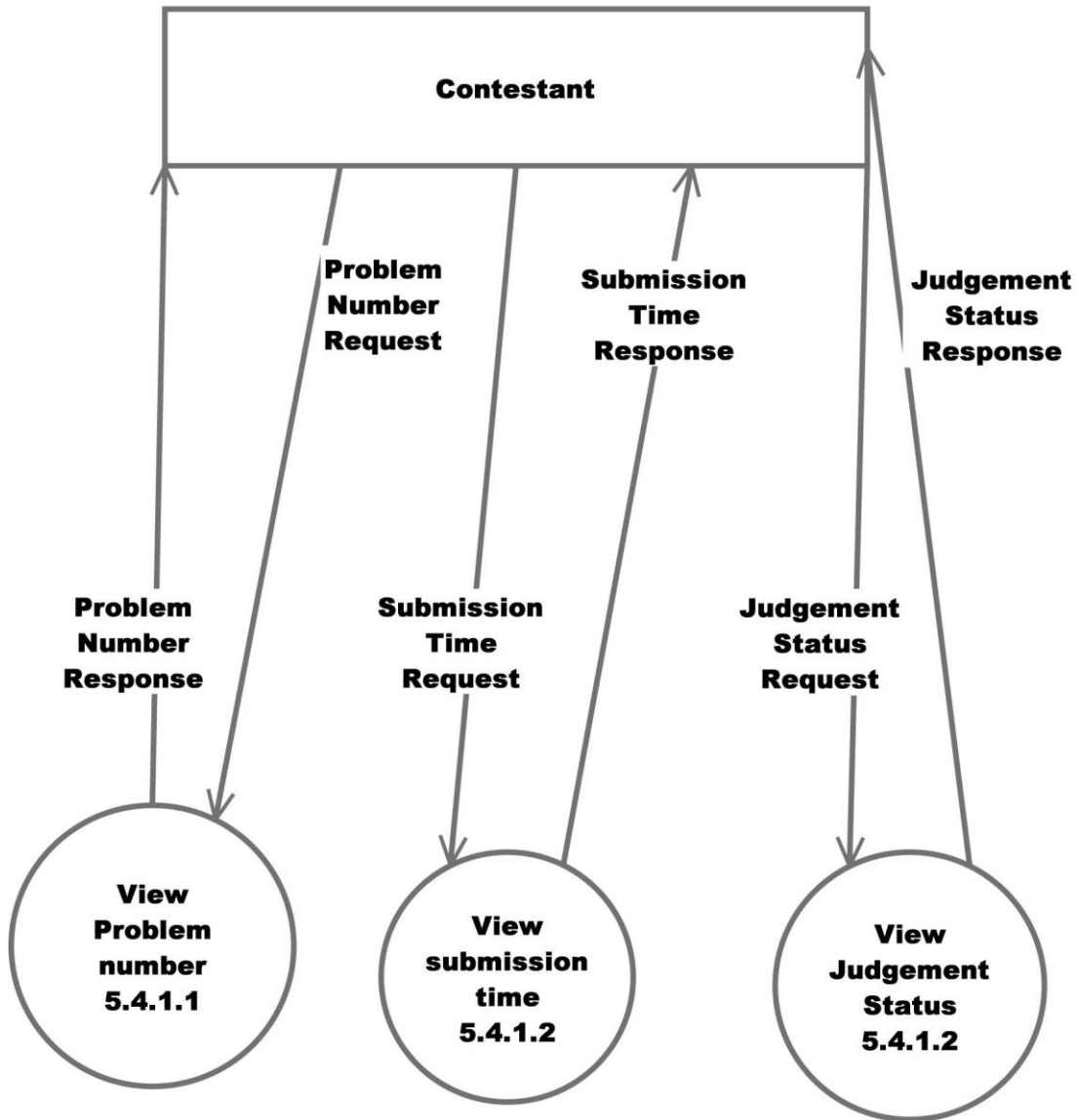
**Legend:**

- A. View Scoreboard Request**
- B. View Scoreboard Response**
- C. View Clarifications Request**
- D. View Clarifications Response**
- E. View Problems Request**
- F. View Problems Response**
- G. View Submission History Response**
- H. View Submission History Request**

## Process 5, Level 2: View Submissions

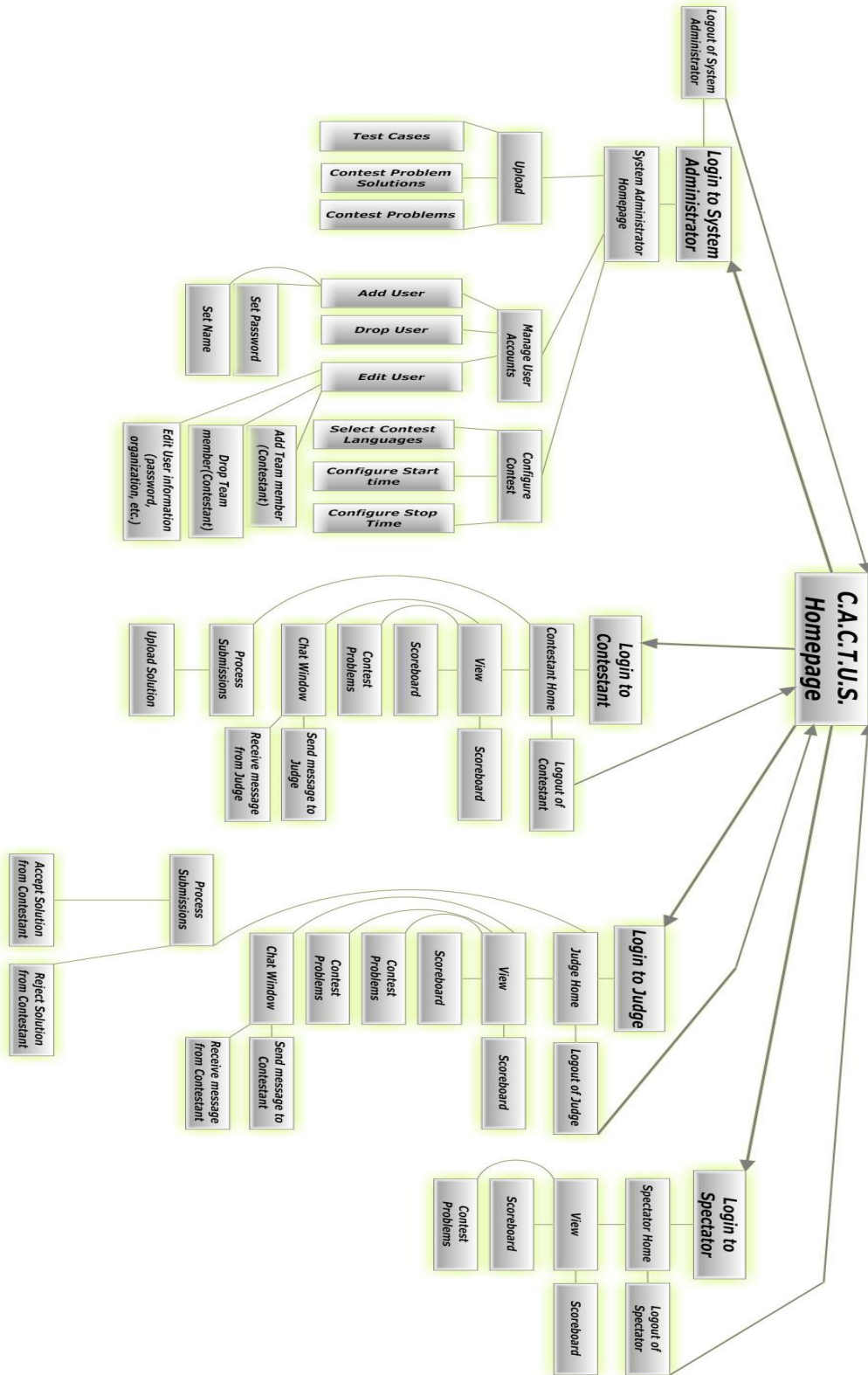


### Process 5, Level 3: View Submission History



### 1.4 Structure Diagram

The data structure diagram is a graphical alternative to the composition specifications within such data dictionary entries.





## 1.5 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

# Detailed Design

## 1.1 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).

### 1.1.1 Logical Data Dictionary Key

The Key for ExoNET’s Logical Data Dictionary is as follows:

- User Type: What user(s) in our system this data type applies to
- Data Name: This particular data entity’s name
- Applicable to: What Use Case in C.A.C.T.U.S. this data entity pertains to
- Data Type: The type of data this entity represents (such as ‘String’ or ‘int’)
- Data Size: The size that this data entity is going to take up in C.A.C.T.U.S.
- Description: A brief description of what this data entity is going to be used
- Acceptable Input Format: A listing of what input is allowable in C.A.C.T.U.S. for this data entity
- Examples of Good Input: Listing(s) of a properly constructed or executed entry for this data entity
- Examples of Bad Input: Listing(s) of unacceptable input for this data entity
- Notes: Additional information regarding this data entity (if any are necessary)

1.1.2 Logical Data Dictionary Spreadsheet

User Type	Data Name	Applicable to	Data Type	Data Size	Description
All	broadcastMessage	All Users	String	0 < Characters =<420	A message broadcast to all users, such as the scoreboard freezing an hour before the contest ends
All	contestProblem	All Users	File	<1MB	A File containing the description for a problem in the contest
	logoutButtonText	All Users	String	Up to 15 Characters	The text on the logout button
	scoreboard	All Users	String Array	0 <= Characters < 420 per cell	An Array containing information regarding each Contestant's progress in the contest
	Start Time/Stop time	All Users	String	XX:XX:XX (3 sets of 2 numerical Characters separated by a colon)	A time representation of when the contest starts/stops. Ordered by Hour:Minutes:Seconds
	Tab	All Users	Graphical User Interface Tab	Not measurable at this time	A tab in a graphical user interface that, when clicked, changes the system's view of the GUI
All	password	Authentication	String	5-20 Characters	A key to authenticate a given userName
	userName	Authentication	String	5-20 Characters	A System Administrator, Judge, Contestant, or Spectator userName for entry into C.A.C.T.U.S., for user authentication purposes
	sendMessageButton	Chat Window	Button	Not measurable at this time	A Button that sends the input from the text box of the Chat Window to the text area of the Chat Window, where previous messages are viewed from Contestant and Judge
	Text Area	Chat Window, Contestant, Judge	String	Up to 420 Characters per Message	View the previous messages from Judge and Contestant
	Text Box	Chat Window, Contestant, Judge	String	Up to 420 Characters per Message	A text entry into a chat window that will be used between Judge and Contestant
	contestantWelcomeLabel	Contestant	String	Up to 50 Characters	A welcome message that is displayed for a contestant upon logging into C.A.C.T.U.S.

# NET Solutions

## Data Dictionary

Data Name	Acceptable Input Format	Examples of Good Input	Examples of Bad Input	Notes
broadcastMessage	Character Array with less than 420 Characters, and greater than or equal to 0 Characters	The Contest is ending in five minutes! "Wow, that team didn't solve any problems! What a bunch of failures!"	that'sFunny.JPG, youLose.PNG, whatAmIDoingHere.XLXS	This message is broadcast for all users logged into C.A.C.T.U.S.
contestProblem	.txt	contestProblem1.txt	haha.JPG, youHaveNoClueHowToDoThis.png	The Contest Problem is submitted via a .txt file for all Users to View.
logoutButtonText	All ASCII Characters	Logout	>15 character submission	It's a message on the logout button.
scoreboard	A Character Array with greater than or equal to 0 Characters and less than 420 Characters	"---", "12:38:01, Problem 1	someTeamsSolution.xlsx, myResearchPaper.docx	An array of Strings that represent each team's progress
Start Time/Stop time	(00:00:00 to 23:59:59), Military Time	12:59:07, 23:59:59	99:99:99, -00:00:00,asdf	The start/stop time for a contest, determined by the System Administrator
Tab	A Left Click, or (Left Click + Hold + Drag)	A Left Click.	Anything that has nothing to do with Left Clicking	
password	A-Z, a-z, 0-9	goodVives29, howYABEEN99	aaap, !><, QQQ	Checks the Database to allocate privileges to a user; will deny access for incorrect entry
userName	A-Z, a-z, 0-9	abc23teamOne, 123school5pirit	453v, 1!, >><<, asdf	Checks the Database to allocate privileges to a user; will deny access for incorrect entry
sendMessageButton	A Left Click	(Mouse Button) Left Click	(Mouse Button) Right Click, (Mouse Button) Middle Click, any other clicking not associated with Left Click.	Left Click the Button to send a message.
TextArea	Previous entries from a Text Box	String	>420 Character String submission	Binary Color coordinated between recipient and sender
TextBox	All ASCII Characters	How are you doing?, Why can't I submit this solution? Is there an error in the test cases? Do you smell something burning?	>420 Character String submission	This dialogue box will be used for communication between Judges and Teams
contestantWelcomeLabel	All ASCII Characters	asdf,qwerty, welcomeTeamX	>50 character submission	It's a welcome message.

User Type	Data Name	Applicable to	Data Type	Data Size	Description
	sendMessageButtonText	Judge, Contestant	String	Up to 15 Characters	The text on the Send Message Button
	problemSolution	Judge, System Administrator	File	<1MB	A file containing the solution for a problem in the contest
	Test Case	Judges, System Administrator	File	> 0 Characters	A test case is obtained by parsing text from a file to produce Integers, Strings, or any other input format for entry as parameters into a Contestant Submission
	Submission	Process Submissions	File	< 1MB	A Team's code submission for judgement
	Time Stamp	Scoreboard	String	XX:XX:XX (3 sets of 2 numerical Characters separated by a colon)	A time representation of when a team submits a correct solution to a contest problem. Ordered by Hour:Minutes:Seconds
	accountPasswordConfirmLabel	System Administrator	String	Up to 15 characters	The text on the accountPasswordConfirmLabel
	accountTypeLabel	System Administrator	String	Up to 15 Characters	A label next to options to select the User Account
	accountUserNameLabel	System Administrator	String	Up to 15 Characters	The text on an Account User Name Label
	addAccountButton	System Administrator	Button	Not measurable at this time	The button used to confirm the addition of a user account in C.A.C.T.U.S.
	addMemberButton	System Administrator	Button	Not measurable at this time	This adds a member's name to a Contestant
	addSelectedProblemButton	System Administrator	Button	Not measurable at this time	This adds a specific problem that has already been uploaded to the system to the set of contest problems
	addSupervisedProblemButton	System Administrator	Button	Not measurable at this time	This adds a problem to the set of supervised problems for a Judge Account.
	addSupervisedTeamButton	System Administrator	Button	Not measurable at this time	This adds a Contestant to the list of supervised teams for a Judge
	availableProblemsLabel	System Administrator	String	Up to 30 Characters	Label for the list that displays currently available problems for the contest
	browseForDescriptionFileButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a file browser to select the problem description file
	browseForInputFileButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a file browser to select the problem input for a test case



Data Name	Acceptable Input Format	Examples of Good Input	Examples of Bad Input	Notes
sendMessageButtonText	All ASCII Characters	Send -->, Sendj,	>15 character submission	It's a message on the Send Message button.
problemSolution	.txt	contestProblem1Solution.txt	troilSolution.JPG, troilFace.PNG, badIdeabear.XLS, scopeCreep.circ	The Contest Problem Solution is submitted via a .txt file for Judge and System Administrator to View
Test Case	.txt	sampleSubmission.txt, cactus.txt	yourFaceWhen.JPG, 410mldermis.XLS	A System Administrator uploads a test case to check the validity of a solution submitted by a Contestant
Submission	java file	submission.java, problem1.java	!!%&%', toTheJudges.jpg, asdf	Note that other file formats, such as .c, may be added to the acceptable input format in the future
Time Stamp	(00:00:00 to 23:59:59), Military Time	12:59:07, 23:59:59	99:99:99, -00:00:00,asdf	A Military time representation of when a Contestant submitted a correct solution to a contest problem
accountPasswordConfirmLabel	All ASCII Characters	Confirm	>15 character submission	It's a message on the accountPasswordConfirmLabel for the System Administrator
accountInputLabel	All ASCII Characters	Judge, Contestant, Spectator	> 15 character submission	It's a label next to the account type choices for creating user accounts in the System Administrator section of C.A.C.T.U.S.
accountUserNameLabel	All ASCII Characters	team1,judge23,spectator69	>15 character submission	
addAccountButton	A Left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
addMemberButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
addSelectedProblemButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
addSupervisedProblemButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
addSupervisedTeamButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
availableProblemsLabel	All ASCII Characters	AvailableProblems	>30 character submission	
browseForDescriptionFileButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
browseForInputFileButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.



User Type	Data Name	Applicable to	Data Type	Data Size	Description
	browseForOutputFileButton	System Administrator	Button	Not measurable at this time	this is the button that triggers a file browser to select the problem output for a test case
	browseForSolutionFileButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a file browser to select a problem solution (source code) for a test case
	changePasswordButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a text field to alter the System Administrator Password
	chosenProblemLabel	System Administrator	String	Up to 30 Characters	Label for the list that displays currently selected problems
	configureContestProblemsPanel	System Administrator	String	Up to 30 Characters	Label for the tab that contains contest problem configuration settings
	configureContestSettingsPanel	System Administrator	String	Up to 30 Characters	Label for the tab that contains contest settings configuration
	confirmChangePasswordLabel	System Administrator	String	Up to 30 Characters	Label for the button that confirms a password alteration for System Administrator
	contestLengthHourSpinnerLabel	System Administrator	String	Up to 30 Characters	Label for the spinner that determines hour duration for the contest
	contestLengthLabel	System Administrator	String	Up to 30 Characters	Label for the message before the hour and minute spinners
	contestLengthMinutesSpinnerLabel	System Administrator	String	Up to 30 Characters	Label for the spinner that determines minutes duration for the contest
	deleteAccountButton	System Administrator	Button	Up to 30 Characters	Label for the button that deletes a user account
	deleteMemberButton	System Administrator	Button	Not measurable at this time	This is the button that triggers the deletion of a member from a Contestant
	deleteProblemButton	System Administrator	Button	Not measurable at this time	This is the button that triggers the delete of a problem from the contest
	displayNameLabel	System Administrator	String	Up to 30 Characters	This is the label that is in front of the displayNameTextField in Manage Accounts
	editAccountButton	System Administrator	Button	Not measurable at this time	This is the button that triggers an account edit
	editProblemButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a contest problem edit
	enabledAccountsLabel	System Administrator	String	Up to 30 characters	This is the label that is above the accounts that are currently enabled in Manage Accounts

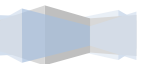




Data Name	Acceptable Input Format	Examples of Good Input	Examples of Bad Input	Notes
browserForOutputFileButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
browserForSolutionFileButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
changePasswordButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
chosenProblemsLabel	All ASCII Characters	Chosen Problems	>30 character submission	
configureContestProblemsPanel	All ASCII Characters	Configure Contest Problems	>30 character submission	
configureContestSettingsPanel	All ASCII Characters	Configure Contest Settings	>30 character submission	
confirmChangePasswordLabel	All ASCII Characters	Change Password	>30 character submission	
contestLengthHoursSpinnerLabel	All ASCII Characters	Hours	>30 character submission	
contestLengthLabel	All ASCII Characters	Set Contest Length	>30 character submission	
contestLengthMinutesSpinnerLabel	All ASCII Characters	Minutes	>30 character submission	
deleteAccountButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
deleteMemberButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
deleteProblemButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
displayedNameLabel	All ASCII Characters	Displayed Name:	> 30 characters	
editAccountButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
editProblemButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
enabledAccountsLabel	All ASCII Characters	Enabled Accounts	>30 characters	



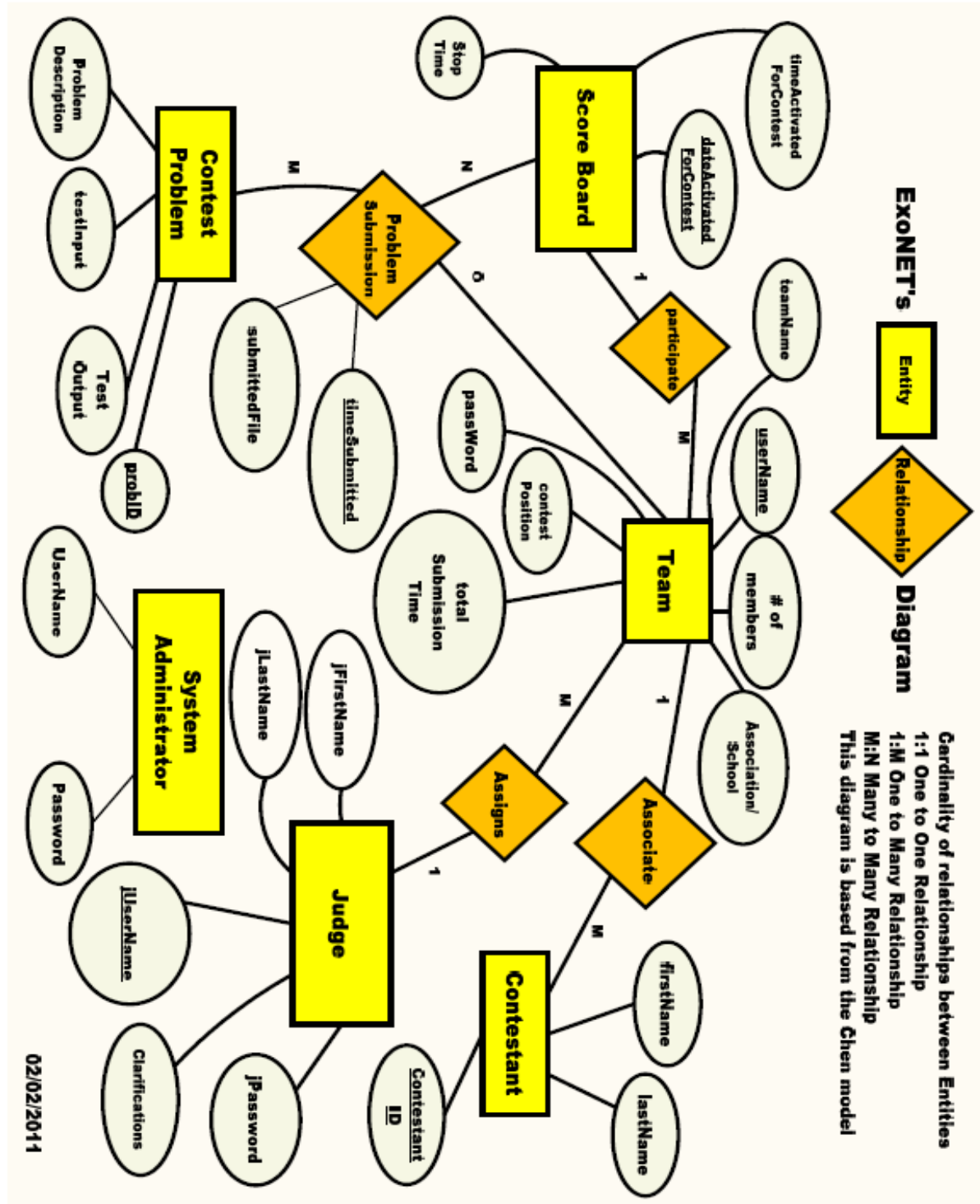
User Type	Data Name	Applicable to	Data Type	Data Size	Description
	languageIsSelected	System Administrator	Boolean	True/False	True/False for a given language selection
	logoutButton	System Administrator	Button	Not measurable at this time	This is the button that allows the System Administrator to log out of the system
	manageAccountsPanel	System Administrator	Tab	Not measurable at this time	This is the tab that displays all pertinent actions and queries for managing accounts
	memberNamesLabel	System Administrator	String	Up to 30 Characters	Header above the member names list
	moveProblemDownButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a selected problem to be moved down in the chosen problems list
	moveProblemUpButton	System Administrator	Button	Not measurable at this time	This is the button that triggers a selected problem to be moved up in the chosen problems list
	OrganizationNameLabel	System Administrator	String	Up to 50 characters	This is the label that is directly to the left of the organizationNameTextfield box
	User Account	System Administrator	Database Table	(N/A) at this time	A User Account created in our database to store information about a user in C.A.C.T.U.S.
	userAccountConfirmationPassword	System Administrator	String	5-20 Characters	A user account password that is confirmed by the System Administrator
	teamName	View, Process Submissions	String	5-20 Characters	A Contestant's Team Name
	removeProblemButton	System Administrator	Button	Not measurable at this time	This button removes a selected problem from the chosen problems for the contest
	removeSelectedProblemButton	System Administrator	Button	Not measurable at this time	This button removes a selected problem from the uploaded problems for the contest
	removeTeamButton	System Administrator	Button	Not measurable at this time	This button removes a team from the contest.



Data Name	Acceptable Input Format	Examples of Good Input	Examples of Bad Input	Notes
languageSelected	TRUE, FALSE	useOfJava = True; useOfC = false; useOfCplusplus = true;	Any non-boolean statement for the selection of a language in C.A.C.T.U.S.	Note that "useOfJava = True," "and useOfC=false," etc., are made up boolean variable names for the selection of languages in C.A.C.T.U.S.. This data type will come of should there be multiple languages that are selectable in our system.
logoutButton	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
manageAccountsPanel	A left click	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
memberNameLabel	All ASCII Characters	Member Names:	>30 characters	Just click it.
moveProblemDownButton	A left click.	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
moveProblemUpButton	A left click.	A Left Click.	Anything that has nothing to do with Left Clicking	Just click it.
OrganizationNameLabel	All ASCII Characters	Sienna College	>50 characters, (and nonsensical organization names)	
User Account	(N/A) at this time	(N/A) at this time	(N/A) at this time	
userAccountConfirmationPassword	A-Z, a-z, 0-9	goGoBuckeyes1969	>20 character submission	It's a user account password, confirmed only by the system administrator
teamName	A-Z, a-z, 0-9	Sienna Gold, Sienna White, UCONN Blues	ad, 77, j10-	Note that a Contestant is a team of players, that has one name
removeProblemButton	A left click.	A Left Click.	Anything that has nothing to do with Left Clicking	
removeSelectedProblemButton	A left click.	A Left Click.	Anything that has nothing to do with Left Clicking	
removeTeamButton	A left click.	A Left Click.	Anything that has nothing to do with Left Clicking	

## 2.1 ER Diagram

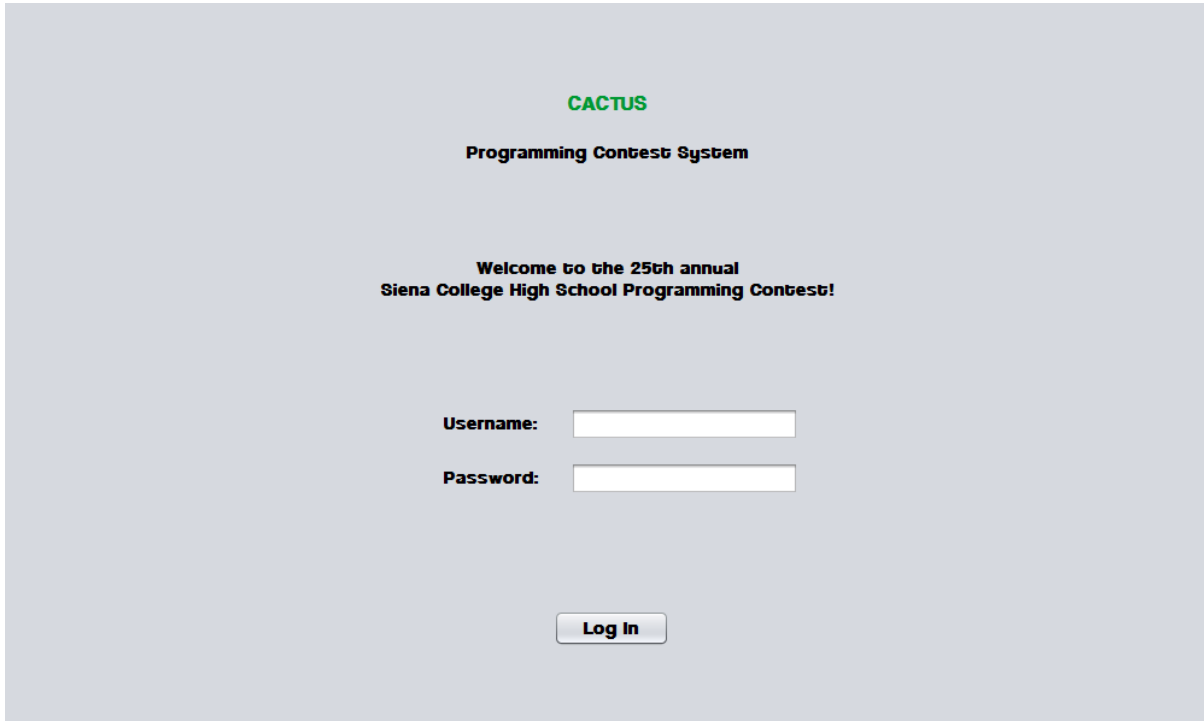
An ER diagram is an abstract and conceptual representation of data. It is used to represent the logical format of a database.



### 3.1 Prototype Screens

#### 3.1.1 Log in Screen

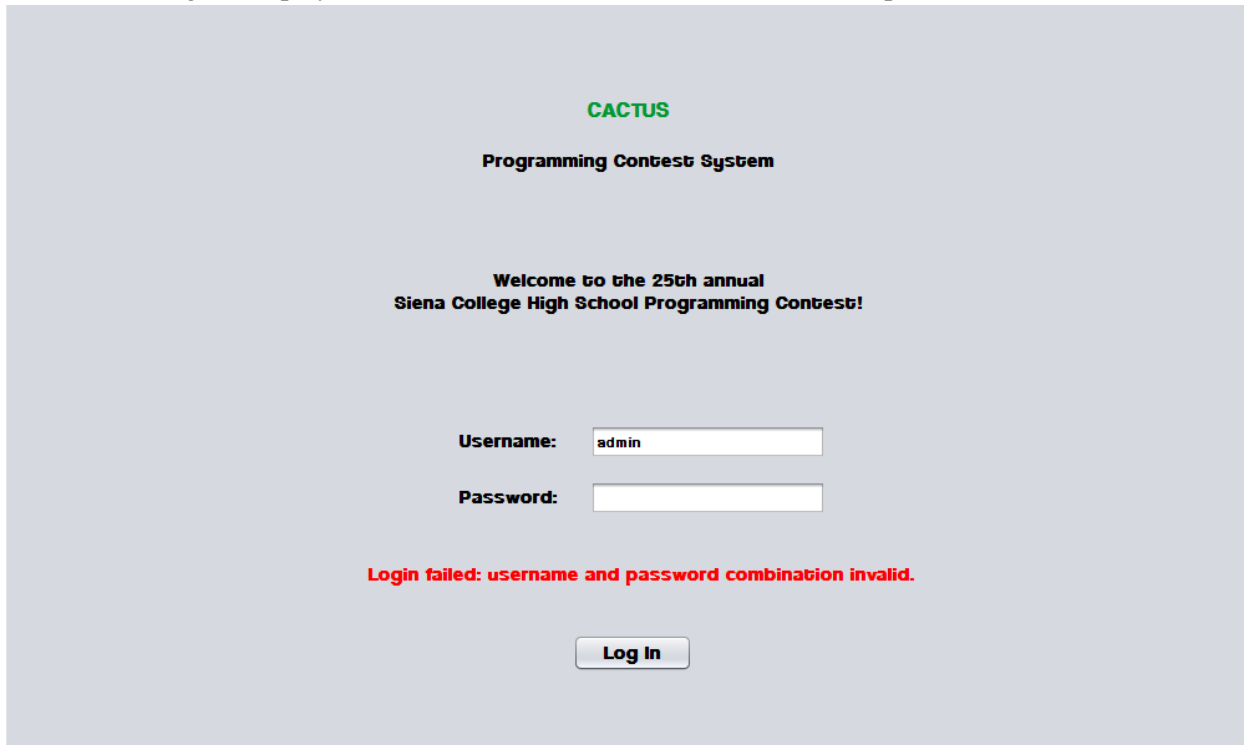
The login screen allows users to login to C.A.C.T.U.S. using a username and password.



The screenshot shows a 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 is a welcome message: "Welcome to the 25th annual Siena College High School Programming Contest!". The login form consists of two input fields: "Username:" and "Password:", both with empty text boxes. A "Log In" button is centered below the fields.

#### 3.1.2 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 is displayed: "Login failed: username and password combination invalid." The "Log In" button remains at the bottom.

### 3.1.3 Contestant Screens

#### 3.1.3.1 View Clarifications

Contestants can view all Clarifications sent out by the Judges.

Welcome, Team 1 Logout

Submissions Problems **Clarifications** Scoreboard

**Clarifications**

Problem	Clarification Number
Problem 1	1
Problem 2	1

**Viewing Clarification: Problem 1, Number 1**

Problem 1 clarification

#### 3.1.3.2 Problems List

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

Welcome, Team 1 Logout

Submissions Problems **Clarifications** Scoreboard

Select Problem: Problem 1

### 3.1.3.3 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									

### 3.1.3.4 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

Select Problem:

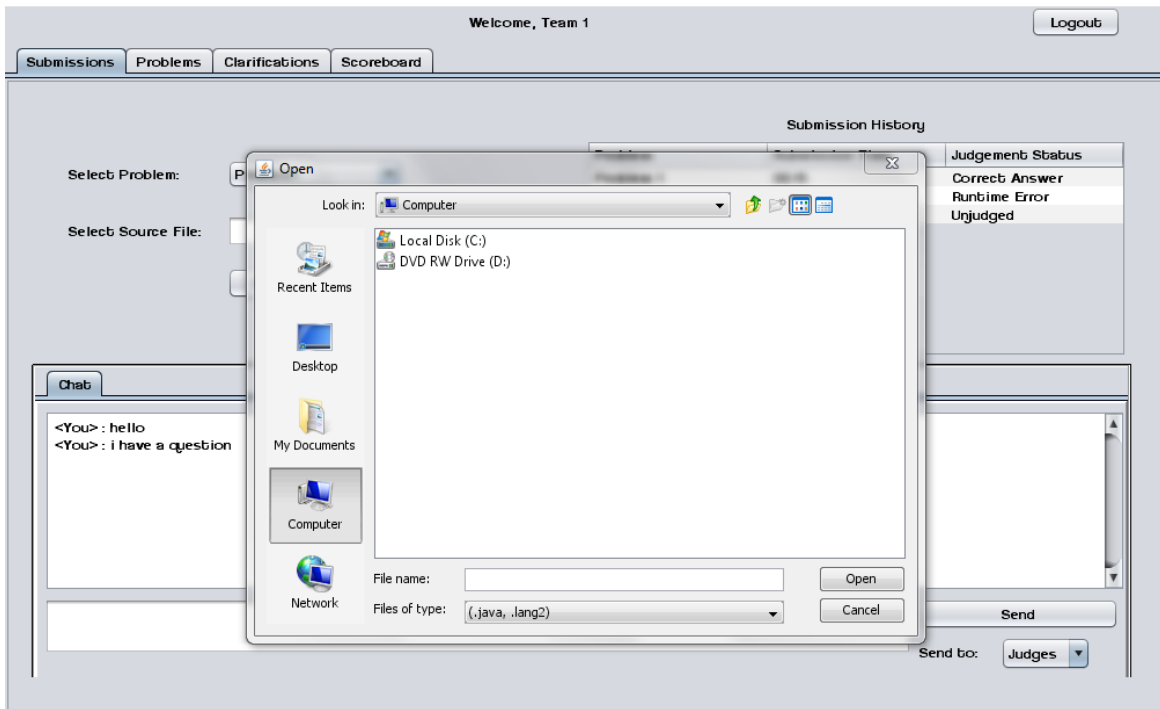
Select Source File:

Chat

Send to:

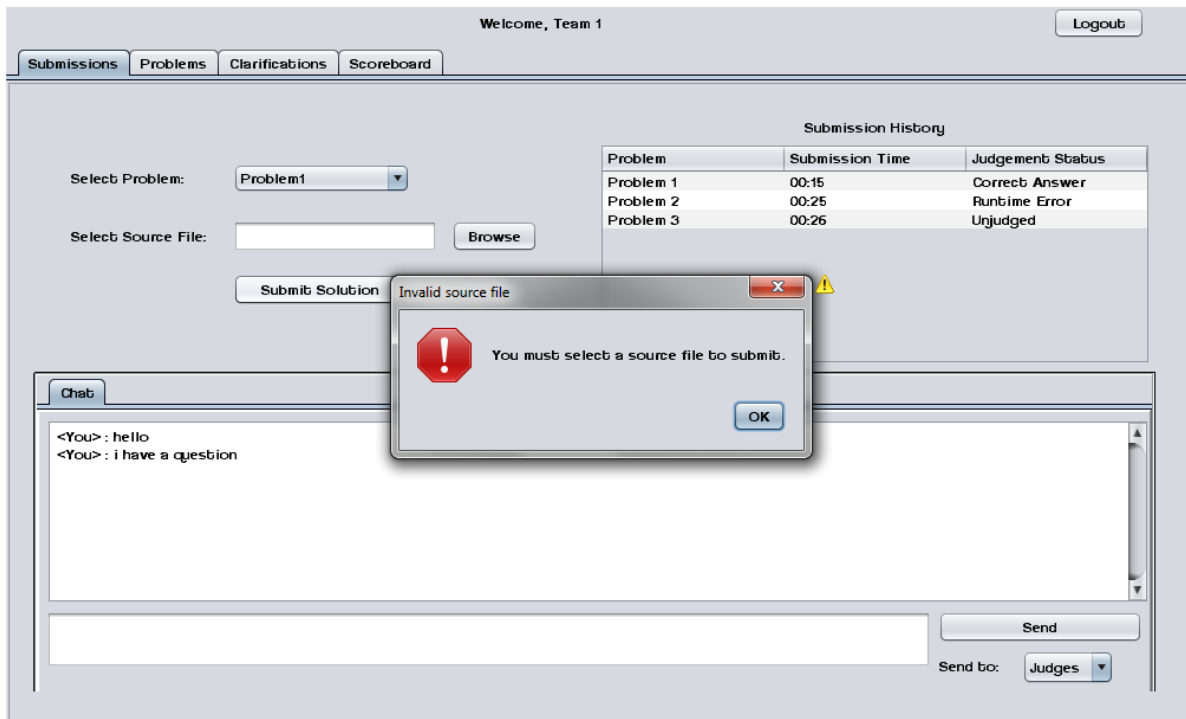
### 3.1.3.5 Submission Screen - Browse for source file

Contestants use a dialog window to browse for source files.



### 3.1.3.6 Submission Screen - No file selected error

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





### 3.1.4 Judge Screens

#### 3.1.4.1 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, it says 'Welcome, Judge1' and has a 'Logout' button. Below that 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 'Selected Problem:' label followed by a dropdown menu showing 'Problem 1'. In the center, there is a large white text area with the title 'Problem 1 clarification'. At the bottom center, there is a 'Create Clarification' button.

#### 3.1.4.2 View Clarifications

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

The screenshot shows the 'View Clarifications' screen. At the top, it says 'Welcome, Judge1' and has a 'Logout' button. Below that 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 table titled 'Clarifications' with two columns: 'Problem' and 'Clarification Number'. The table has three rows: 'Problem 1' with '1', 'Problem 2' with '1', and 'Problem 1' with '2'. The last row is highlighted. Below the table, there is a 'View Clarification' button. To the right of the table, there is a section titled 'Viewing Clarification: Problem 1, Number 2' with a large white text area containing the text 'Problem 1 clarification'.

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

### 3.1.4.3 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.

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

### 3.1.4.4 Problems List

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

Select Problem: Problem 3

Problem 3 document.

### 3.1.4.5 Scoreboard View

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

Welcome, Judge1 Logout

Judge Submissions **Scoreboard** Problems Create Clarification View Clarifications

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									

Refresh Scoreboard

### 3.1.5 System Administrator Screens

#### 3.1.5.1 Change Admin password

The System Administrator may change the System Administrator Account password.

Welcome, System Administrator Logout

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

Old Password:

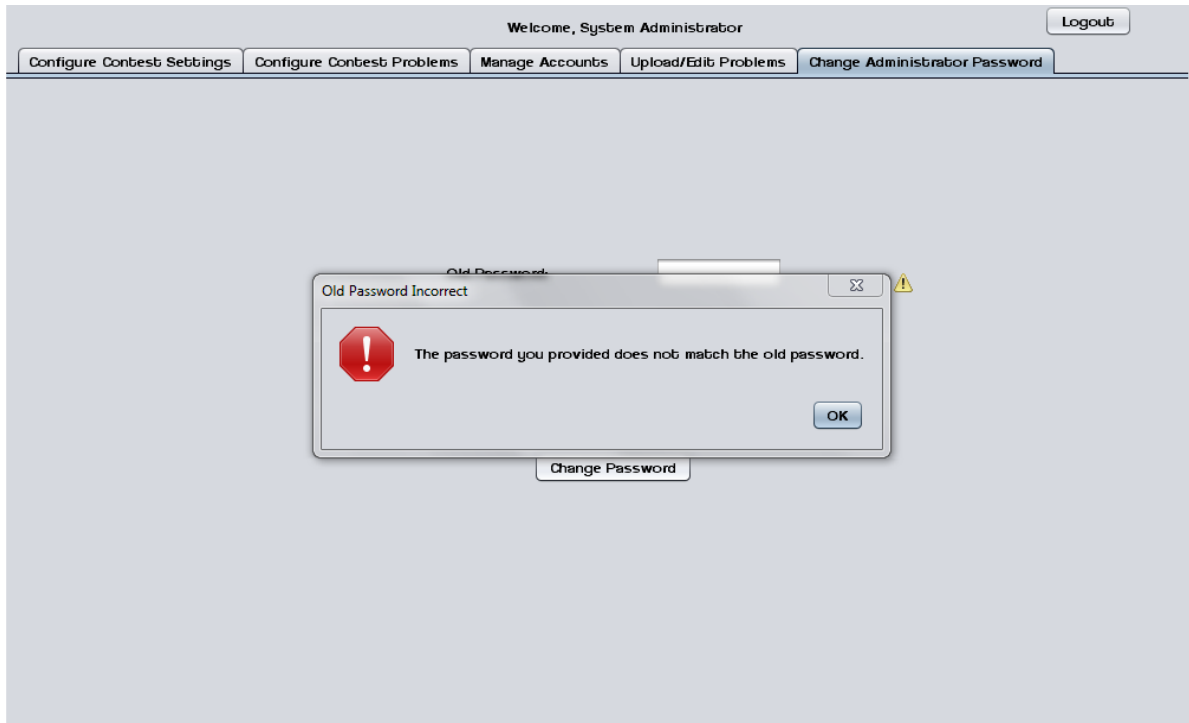
New Password:

Confirm New Password:

Change Password

### 3.1.5.2 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.



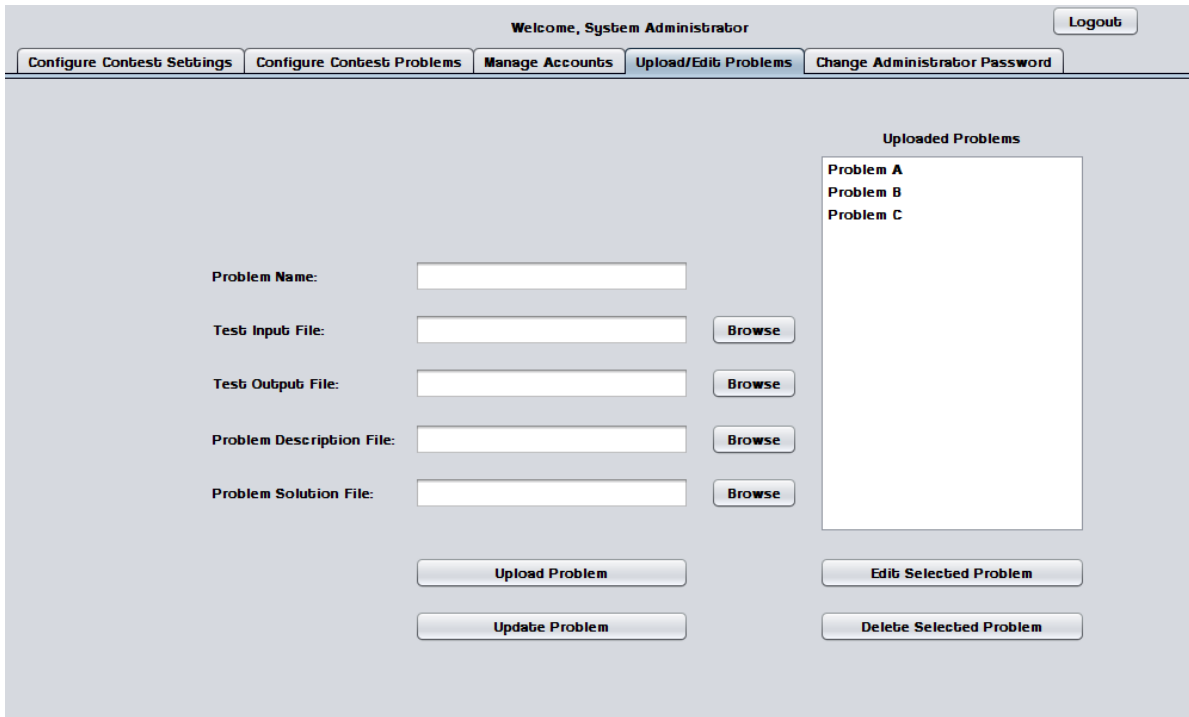
### 3.1.5.3 Change Admin password – New passwords do not match error

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



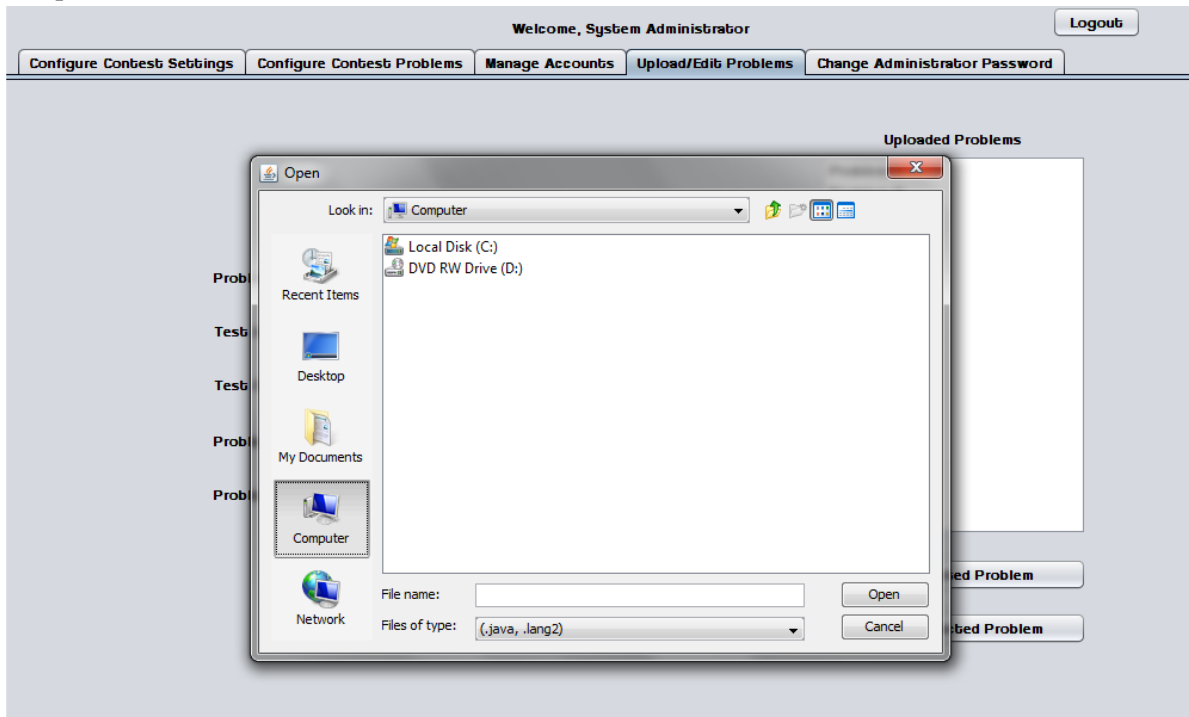
### 3.1.5.4 Upload/ Edit problems

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



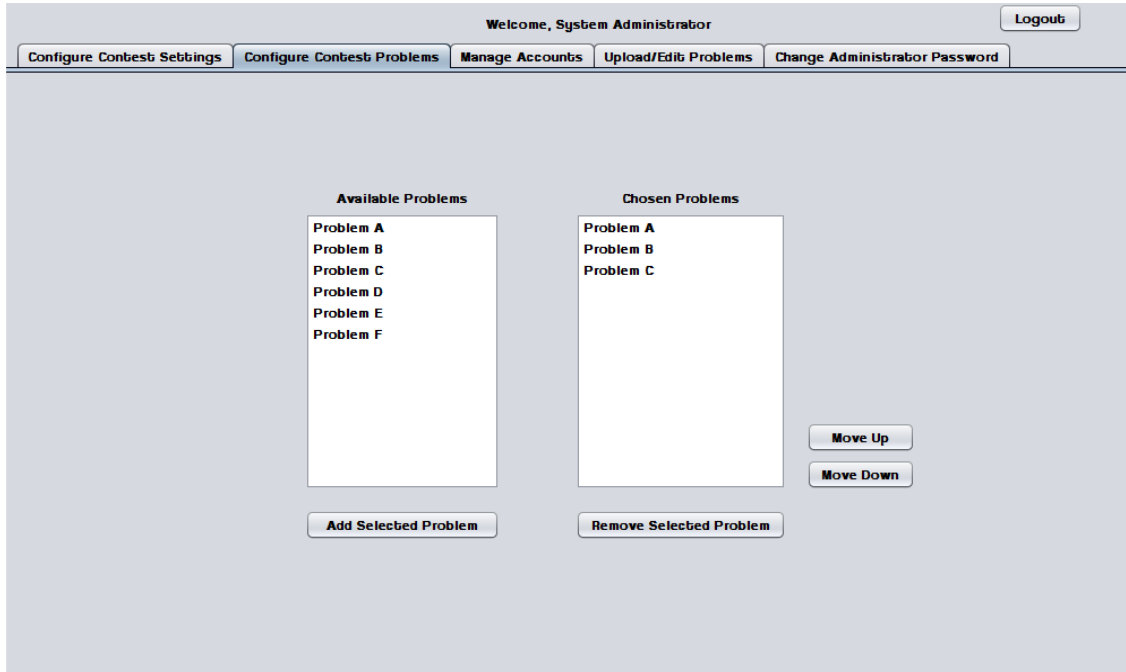
### 3.1.5.5 Upload Problems - Browse for file

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



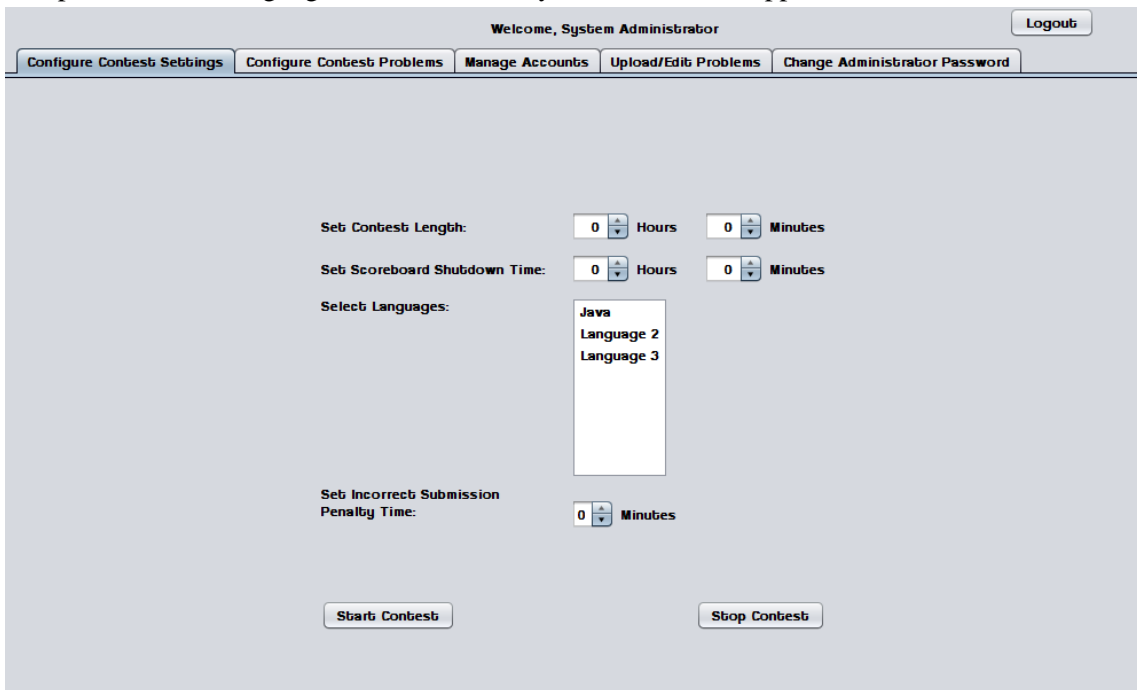
### 3.1.5.6 Configure Contest Problems

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



### 3.1.5.7 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.



### 3.1.5.8 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.

Welcome, System Administrator Logout

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

Set Contest Length: 0 Hours 0 Minutes

Set Scoreboard Shutdown Time: 0 Hours 0 Minutes

Select Languages: 

- Java
- Language 2
- Language 3

Set Incorrect Submission Penalty Time: 0 Minutes

Set Submission Runtime Limit: 1 Minutes

Start Contest Stop Contest

### 3.1.5.9 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.

Welcome, System Administrator Logout

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

Set Contest Length: 0 Hours 0 Minutes

Set Scoreboard Shutdown Time: 0 Hours 0 Minutes

Select Languages: Java

Invalid contest length ✖ ⚠

Contest time must be greater than or equal to one minute.

OK

Set Incorrect Submission Penalty Time: 0 Minutes

Set Submission Runtime Limit: 1 Minutes

Start Contest Stop Contest

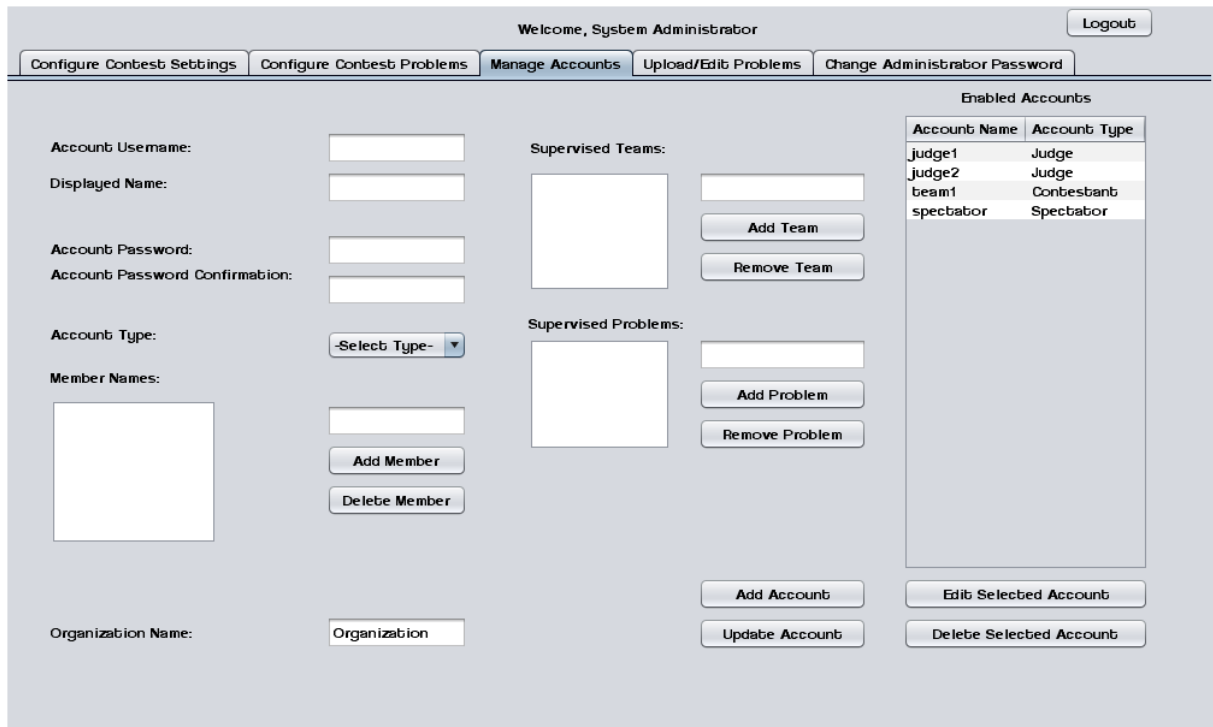
### 3.1.5.10 Invalid Selection Error

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



### 3.1.5.11 Manage Accounts

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





### 3.1.5.12 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.

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:

Organization Name:

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

### 3.1.5.13 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.

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:

Organization Name:

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

### 3.1.5.14 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:

Organization Name:

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

## 3.1.6 Spectator Screens

### 3.1.6.1 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.

### 3.1.6.2 Scoreboard View

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

Welcome, Spectator Logout

Scoreboard Problems Clarifications

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									

Refresh Scoreboard

### 3.1.6.3 View Clarifications

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

Welcome, Spectator Logout

Scoreboard Problems Clarifications

Clarifications

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

Viewing Clarification: Problem 1, Number 1

Problem 1 clarification

View Clarification

## 4.1 Testing Plan

A software project test plan is a document detailing a systematic approach to testing a system. It describes the objectives, scope, approach, and focus of a software testing effort. The process of preparing a test plan is a useful way to think through the efforts needed to validate the acceptability of a software product. The completed test plan will serve to help anyone that is not involved in the testing to understand the ‘why’ and ‘how’ of product validation. The test plan documents the approach that will be used to authenticate and make certain that a product or system that is being tested meets the requirements and any other conditions. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

For a detailed description of ExoNET’s testing plan, please see the Test Plan Document.

## 5.1 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:  
TSSCorp 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

## 6.1 Appendices

### 6.1.1 Appendix A: 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.

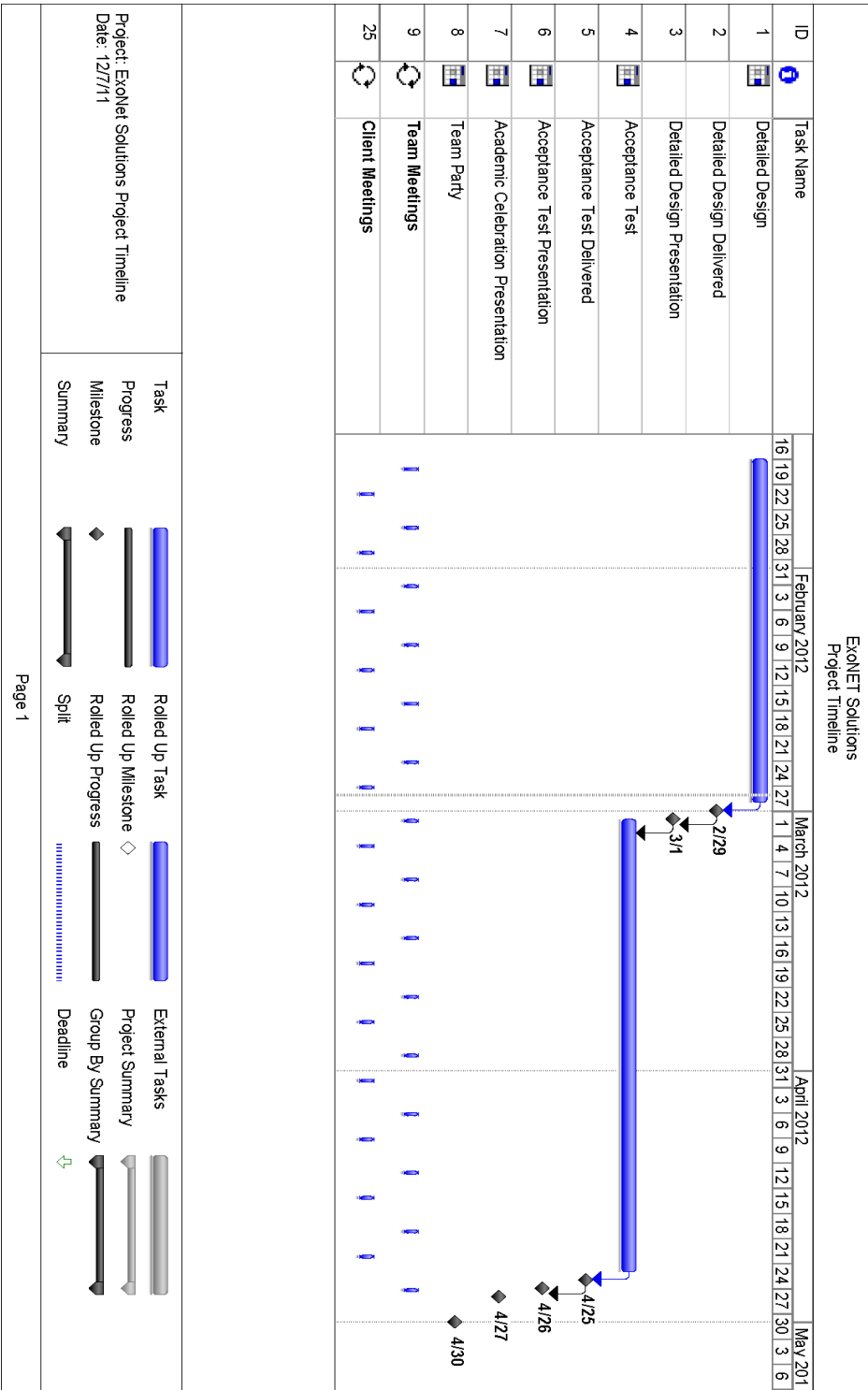
**User Case Narrative:** an explanation of the functions and abilities users have for a specific Software System.

**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.

### 6.1.2 Appendix B: Project Timeline (Gantt Chart)





# Testing Plan

A software project test plan is a document detailing a systematic approach to testing a system. It describes the objectives, scope, approach, and focus of a software testing effort. The process of preparing a test plan is a useful way to think through the efforts needed to validate the acceptability of a software product. The completed test plan will serve to help anyone that is not involved in the testing to understand the ‘why’ and ‘how’ of product validation. The test plan documents the approach that will be used to authenticate and make certain that a product or system that is being tested meets the requirements and any other conditions. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

## 1.1 Overview and Strategy

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 Plan Description

This plan serves as a template for the testing that ExoNET will be performing until the Acceptance Test. Although many of the major aspects are covered in this document, it is by no means finalized; Unit Tests and Test Cases may be added or deleted as necessary to fit the needs of our project, in relation to changes in our software. As tests are performed, the outcome of these tests will be documented to reflect their results. No further alterations will be made after the day of the Acceptance Test.

This Test Plan is the framework for the testing of C.A.C.T.U.S. (Competitive Algorithm Calculation and Testing in a Unified System), by ExoNET Solutions. There are tests for pertaining to all Users in the system; System Administrator, Judge, Contestant, and Spectator user accounts will be the main focus of these tests. There are also a few other Unit Tests that pertain to the overall performance of system. The tests that ExoNET will be managing include:

***Unit Test:*** These will serve as the backbone of our testing plan. Based on testing the functionality of a single action/component of the system, the test serves to check if a particular part of C.A.C.T.U.S. is working as intended (i.e. is free of errors, and fits Acceptance Test Criteria).

**Test Case:** The Test Case is an individual member of a Unit Test (i.e. many Test Cases make up a Unit Test). Each Test Case tests a specific state of C.A.C.T.U.S., with a specific input and output. Each Test Case has a pass/fail result.

**Integration Test:** The Integration Test tests combinations of components of C.A.C.T.U.S., with each of these components having a successful Unit Test. The purpose of this test is to ensure that all of the system components work together correctly and efficiently.

**System Test:** The System Test checks to ensure that the system meets all the Functional Requirements as listed in the Functional Requirements inventory.

**Performance Test:** The Scalability Test ensures that C.A.C.T.U.S. functions at an acceptable level when a large number of users are using the system concurrently.

**Acceptance Test** The Acceptance Test validates that the system meets the functional and non-functional requirements as specified in the Requirements Specification (which includes functional and non-functional requirement details).

## 2.1 Pass/Fail Criteria

### 2.1.1 Functional Requirements Inventory

The Functional Requirements Inventory will serve as a checklist for our Testing Plan; each of the specifications listed therein will be evaluated by various tests contained in this plan. Each particular test will reflect the intended design specified by Dr. Lim. These tests will have two states: ***PASSED*** and ***FAILED***. One of these states will be underlined to reflect the state that each particular requirement is in, displayed to the right of the specified test.

#### 2.1.1.1 System Administrator

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

58

#### 2.1.1.2 Judge

- Securely log in and out of C.A.C.T.U.S.....***PASSED/FAILED***

- View contest problems.....**PASSED/FAILED**
- Judge will be able to view test cases.....**PASSED/FAILED**
- View Scoreboard.....**PASSED/FAILED**
- View chat window.....**PASSED/FAILED**
- Judge will be able to send broadcast messages to all Contestants.....**PASSED/FAILED**
- Judge able to accept/decline submissions from Contestants.....**PASSED/FAILED**

*2.1.1.3 Contestant*

- Securely log in and log out of C.A.C.T.U.S.....**PASSED/FAILED**
- View contest problems.....**PASSED/FAILED**
- View Scoreboard.....**PASSED/FAILED**
- View broadcasted messages and chat window.....**PASSED/FAILED**
- Contestants will be able to send messages to assigned Judge.....**PASSED/FAILED**
- Submit and save submissions of source code.....**PASSED/FAILED**
- Compile their code.....**PASSED/FAILED**

*2.1.1.4 Spectator*

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

**2.1.2 Non-Functional Requirements Inventory**

- Must be easily used (user friendly); not easily measurable at this time.
- Stability; not easily measurable at this time.

**2.1.3 Exception Handling**

The ultimate goal of project C.A.C.T.U.S. is to create a working system that achieves all of the goals outlined in the Problem Definition of the Software Plan. As part of the system goals, project C.A.C.T.U.S. will include fail-safes that, in the event of an exception error, will handle it without

the whole system failing. At this stage of the project, the potential threats to the system are still unknown; however, one of the main issues that will require handling will focus on the loss of data due to an unintentional or unexpected exit of the software. Such an event could be due to error on the part of the system or the user. In either case, C.A.C.T.U.S. will be designed so that the backup of data will be automatic so that it may be recovered in the event of an error.

#### **2.1.4 Testing Approach**

C.A.C.T.U.S.'s testing approach will involve a series of various tests to check the system's compliance with the Functional and Non-Functional Requirements. The first of these will be the Unit Tests, which are displayed below. Each of these tests will challenge each particular portion of C.A.C.T.U.S. that is relevant to the Functional and Non-Functional Requirements. These tests, which are explained in detail in a separate spreadsheet document, will be tested numerous (to be determined) times, in an effort to obtain accurate and reliable test results. These tests will be either passed or failed; there is no "in between". After all of these Unit Tests are passed satisfactorily, an Integration Test will be put in place to ensure that all of the portions of C.A.C.T.U.S. are meeting the standards put in place by the Functional Requirements Document.

#### **2.1.5 Acceptance Criteria**

Acceptance Testing is concerned with knowledge about validating the functional and non-functional requirements of a purchased or acquired system. This unit includes knowledge about techniques for using the contract, the statement of work, the software requirements specification, and the request for proposal to ensure that the delivered system meets all of the requirements (as perceived by the purchasing or acquiring organization).

The Acceptance Criteria will be determined by ExoNET solutions and by the functional requirements inventory listed in section 1.5 of the Detailed Design document. Please keep in mind that these requirements are subject to change and can be added to as more information is gathered. The project will be broken down into units, based on these functional requirements, which will be tested individually. The units will be tested thoroughly until they all pass individual unit testing. Once all the units pass testing they will be brought back together and will be tested as one system.

### 3.1 Unit Tests

The following is a list of units that will be tested in C.A.C.T.U.S. After all of these tests pass, an Integration Test will follow. Should the Integration Test pass, a System Test will be put in place to ensure that the Functional and Non-Functional Requirements have been met.

Unit Tests are as follows:

Submission  
Scoreboard  
Authenticate  
Database  
Chat Window  
Configure Contest Settings  
Upload

For a more in depth view of these Unit Tests, please refer to *Exhibits 1 – 7* of C.A.C.T.U.S. Unit Tests.

#### 3.1.1 Unit Tests and Test Cases

Each Test Case is pertinent to a particular Unit Test. There are many Test Cases for each Unit Test; they are constructed as follows:

- Pass/Fail Status
- Test Number
- Description
- Action to Perform (input)
- Steps to be executed
- State before Test
- Expected Result
- Observed Result
- Comments
- Tested By
- Test Date

At the bottom of each Excel Spreadsheet, there is a Unit Summary, describing the number of tests in the Unit Test, the percentile of passing tests, the number of passed tests, the number of failed tests, and the date of the last test performed.

#### 3.1.2 Performance Test

The Performance Test is used to determine whether or not the system's performance and responsiveness are acceptable when it is being used by many users at the same time. ExoNET Solutions has communicated with P=NP Solutions regarding a potential Performance Test in the future. This test would entail 40+ users logging into C.A.C.T.U.S., and attempting to run a contest to validate the responsiveness and robustness of the system (S.L.I.C.E. will also be tested on the same day).

The current plan is to obtain enough users on a chosen day to simulate a full scale contest. Currently, this date is tentatively set for sometime in April, 2012. Should this plan commence, a series of tests will be facilitated to effectively run a stability/performance check on C.A.C.T.U.S.

Things such as multiple login attempts and multiple submission attempts (within seconds of each other) will be performed in an attempt to ensure that the system's functionality is unaffected by high traffic (slight lag is acceptable). Chat Window performance and Compiler performance will be evaluated in a similar manner.