

## System Narrative

I will be developing a Help Desk Ticket System. This system will give an employee of an organization who has a work related problem the opportunity to get their problem resolved via the Help Desk System. When a user (an employee who has logged into the Help System) has a problem or query, they will log onto the system using their e-mail and password and make a new support ticket from their user dashboard. The ticket is then submitted to the relevant person or department for resolution.

Administrators have special accounts and their own special admin dashboard after logging into the system. Administrator accounts may be assigned to I.T. Support personnel within a company as their main function is to solve users' tickets. They may view and respond to tickets and when the problem is solved they made close tickets. Administrators have extra options on their dashboard such as changing the status, priority, department, category of tickets submitted. They may also request system summary reports which may show certain information via charts such as number on tickets submitted based on department name.

## List of Requirements

There will be 3 different access types for the help desk system:

- Guest Account (Normal user dashboard)
- Registered User (Normal user dashboard)
- Administrator (Administrator dashboard)

*Guest account details:*

- Have the option to sign up and become registered user
- May only add and have 1 open ticket in the system at any one time
- May update ticket submitted
- May close ticket submitted
- May respond to Administrators to try provide any more details that may be needed to solve ticket

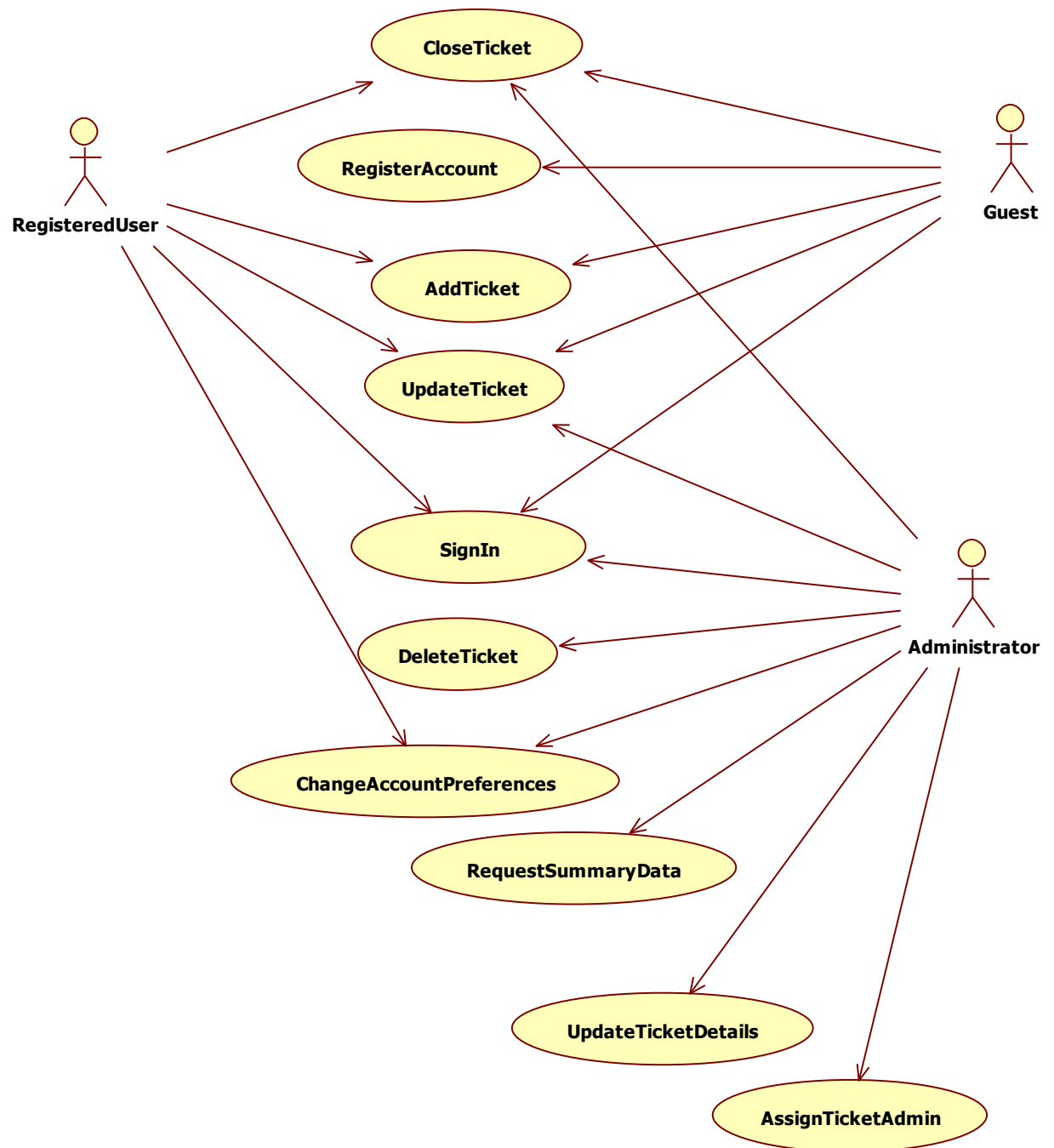
*Registered user account details:*

- Password needed to avoid un-authorized access
- May change account details in preferences tab (email, password, name, department etc)
- May add numerous tickets into the system at any one time
- May update tickets submitted
- May close tickets submitted
- May respond to Administrators to try provide any more details that may be needed to solve tickets

*Administrator account details:*

- Password needed to avoid un-authorized access
- May change account details in preferences tab (email, password, name, department etc)
- May view all tickets submitted (option to view tickets based on department, category, priority etc)
- May respond(update) to tickets submitted
- May close and delete solved tickets
- May change options such as ticket name, department, category and assign ticket priority
- May assign tickets to certain administrators.
- May request system summary data (tables, charts etc)

## Use Case Diagram



## Use Case Descriptions

<b>Use case:</b>	Close Ticket
<b>Actors:</b>	Registered User, Administrator, Guest
<b>Purpose:</b>	To allow Registered Users, Administrators, Guests to close their solved tickets
<b>Overview:</b>	Registered Users, Administrators, Guests close tickets which are already solved
<b>Preconditions:</b>	Ticket must be an open ticket in the system
<b>Postconditions:</b>	Ticket is closed and the open status is marked as closed

### Successful Scenario

1. This use case begins when Registered Users, Administrators, Guests wish to close a solved ticket
2. Registered Users, Administrators, Guests mark ticket status as closed

### Alternative Flow of Events

-

<b>Use case:</b>	RegisterAccount
<b>Actors:</b>	Guest
<b>Purpose:</b>	To allow Guests to upgrade their account to a Registered Account

<b>Overview:</b>	A guest chooses a password and upgrades account to Registered Account
<b>Preconditions:</b>	Must have a guest username
<b>Postconditions:</b>	Guest account is upgraded to registered account
 <b>Successful Scenario</b>	
<ol style="list-style-type: none"> <li>1. This use case begins when a guest wishes to upgrade their account to a registered account</li> <li>2. Guest opens login screen</li> <li>3. Guest enters their guest username</li> <li>4. Guest chooses password</li> <li>5. Guest submits details and system upgrades account to registered account</li> </ol>	
 <b>Alternative Flow of Events</b>	
Line 5: username incorrect. Indicate error. Return to Step 2. Password length/type invalid. Indicate error. Return to Step 3.	

<b>Use case:</b>	<a href="#">AddTicket</a>
<b>Actors:</b>	Guest, Registered User
<b>Purpose:</b>	To allow Guests and Registered Users submit a new ticket into the system
<b>Overview:</b>	A guest or registered user has a query or problem, enters their ticket details and submits ticket into system

**Preconditions:** Guest must not already have ticket in system

**Postconditions:** New ticket is entered into system

### Successful Scenario

1. This use case begins when a guest or registered user has a problem and wishes to submit a new ticket into the system
2. guest or registered user fills out ticket information
3. guest or registered user submits ticket into the system

### Alternative Flow of Events

Line 3:

Guest already has a ticket in the system. Indicate error.

**Use case:** [UpdateTicket](#)

**Actors:** Guest, Registered User, Administrator

**Purpose:** To allow Guests, Administrators and Registered Users update a ticket in the system

**Overview:** A guest, admin or registered user has some more information and wants to update the already open ticket in the system

**Preconditions:** Ticket must be open in the system

**Postconditions:** Ticket is changed and system updates the ticket

### Successful Scenario

1. This use case begins when a guest, admin or registered user has some more information and wishes to update a ticket in the system
2. guest, admin or registered opens ticket
3. guest, admin or registered updates the ticket and submits the new information

### Alternative Flow of Events

Line 3:

Ticket is already closed. Indicate error.

**Use case:** [SignIn](#)

**Actors:** Guest, Registered User, Administrator

**Purpose:** To allow Guests, Admins and Registered to log into the system and their dashboard

**Overview:** A guest, admin or registered user wishes to log into the system to access their dashboard

**Preconditions:** guest, admin or registered must enter in correct details

**Postconditions:** guest, admin or registered user have access to their dashboard

### Successful Scenario

1. This use case begins when a guest, admin or registered user wishes to log into the system to access their dashboard

2. guest, admin or registered user fills out account information
3. guest, admin or registered user submits account information into the system for authentication

### Alternative Flow of Events

Line 3:

Guest user name already in use. Indicate error. Return to Step 2

Invalid username or password. Indicate error. Return to Step 2.

**Use case:** DeleteTicket

**Actors:** Administrator

**Purpose:** To allow administrators to delete tickets from the system

**Overview:** An admin wishes to delete a closed ticket from the system

**Preconditions:** Ticket must be closed

**Postconditions:** Ticket is deleted from system

### Successful Scenario

1. This use case begins when an admin wishes to delete a closed ticket from the system.
2. admin opens ticket
3. admin submits ticket to system for deletion

### Alternative Flow of Events

Line 3:

Ticket is open and in use. Indicate error.



--

<b>Use case:</b>	ChangeAccountPreferences
<b>Actors:</b>	Administrator, Registered User
<b>Purpose:</b>	To allow Admins and Registered Users to change their account settings
<b>Overview:</b>	An admin or registered user wishes to change their account settings
<b>Preconditions:</b>	Account must have the correct privileges
<b>Postconditions:</b>	New account settings are updated and saved into the system

#### Successful Scenario

1. This use case begins when an admin or registered user wishes to change their account settings
2. admin or registered opens account preferences
3. admin or registered user submits new account info and submits it to the system to be updated and saved

#### Alternative Flow of Events

-

<b>Use case:</b>	RequestSummaryData
<b>Actors:</b>	administrator
<b>Purpose:</b>	To allow administrators to request system summary date
<b>Overview:</b>	A administrator wishes to access pie charts, tables with system summary data
<b>Preconditions:</b>	Must have correct privileges
<b>Postconditions:</b>	System summary data is outputted into a file for admin use
<b>Successful Scenario</b>	
<ol style="list-style-type: none"><li>1. This use case begins when an admin wishes to view system data in the form of tables and charts</li><li>2. admin submits request to the system</li><li>3. system outputs data into a file</li></ol>	
<b>Alternative Flow of Events</b>	
-	

<b>Use case:</b>	UpdateTicketDetails
<b>Actors:</b>	Administrator
<b>Purpose:</b>	To allow admins to update specific ticket details
<b>Overview:</b>	An admin wishes to update specific ticket information such as ticket name, department, priority
<b>Preconditions:</b>	Ticket must be in the system
<b>Postconditions:</b>	Ticket details are updated and saved into the system
<b>Successful Scenario</b>	
<ol style="list-style-type: none"><li>1. This use case begins when an admin wishes to change or correct certain ticket details</li><li>2. admin opens ticket window</li><li>3. admin changes specific details and submits new details into system</li></ol>	
<b>Alternative Flow of Events</b>	
-	

<b>Use case:</b>	<a href="#">AssignTicketAdmin</a>
<b>Actors:</b>	Administrator
<b>Purpose:</b>	To allow admins to assign specific tickets to specific admins
<b>Overview:</b>	An admin wishes to assign a certain ticket to another administrator
<b>Preconditions:</b>	Must be an open ticket in the system
<b>Postconditions:</b>	Tickets assigned admin is changed

#### **Successful Scenario**

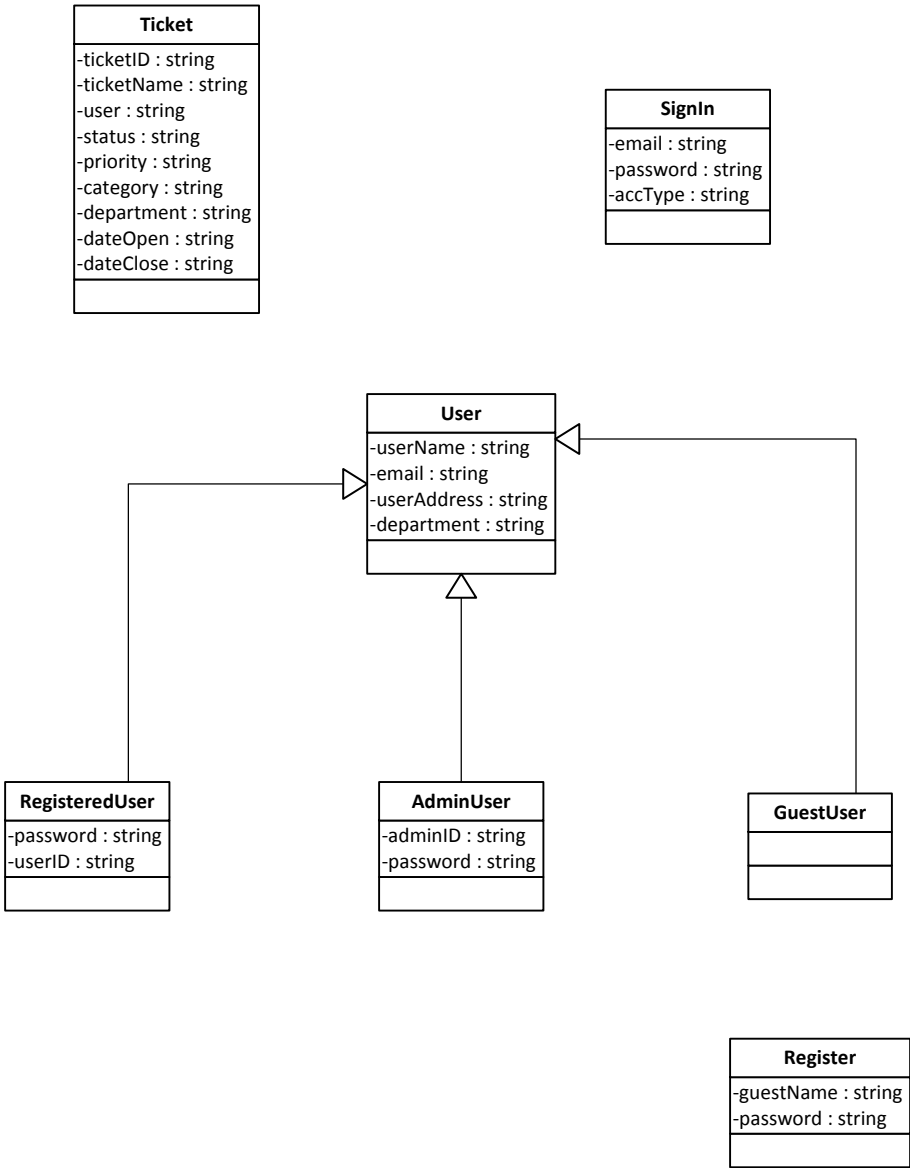
1. This use case begins when an admin wishes to assign a ticket to another admin in the system
2. admin opens ticket window
3. admin assigns ticket to another admin and submits it into the system

#### **[Alternative Flow of Events](#)**

Line 3:

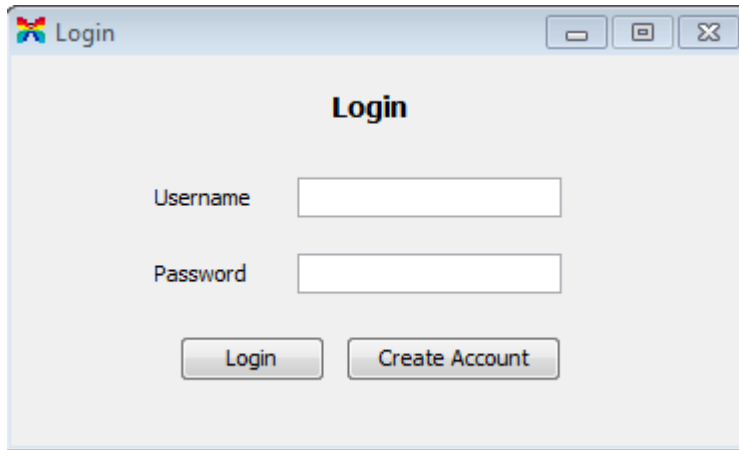
Ticket is already closed in the system. Indicate error.

Class Diagram



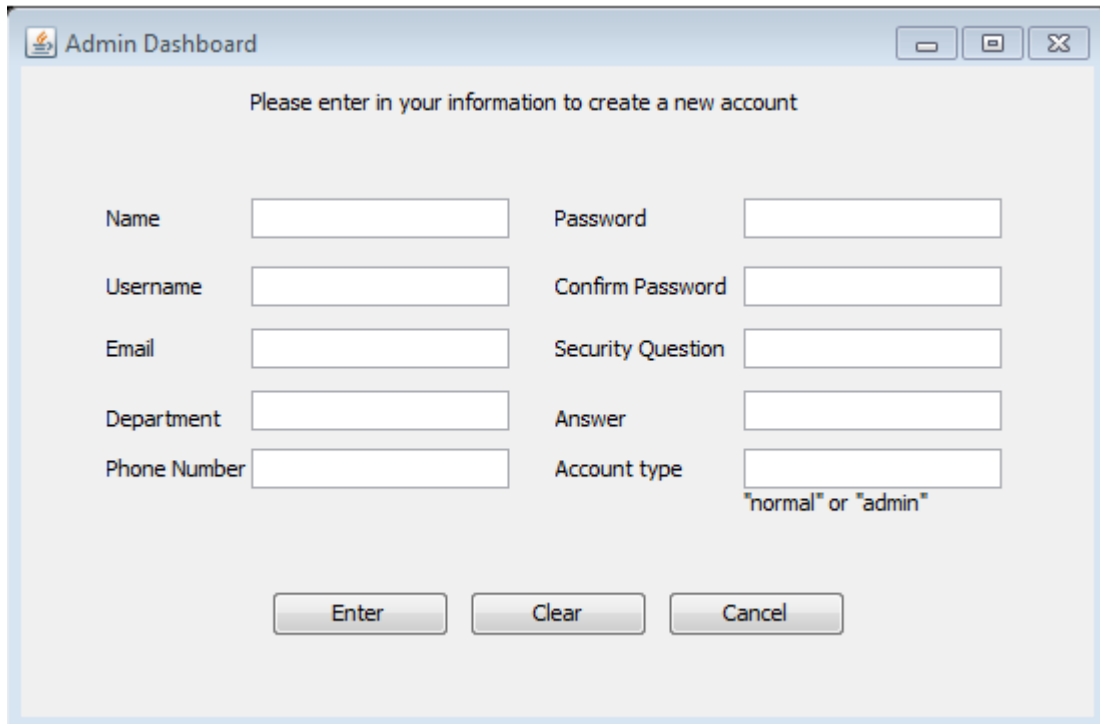
## Input Design

### Login

A screenshot of a login window titled "Login". The window has a light blue header bar with a small icon on the left and three standard window control buttons (minimize, maximize, close) on the right. The main content area is light gray. At the top center, the word "Login" is displayed in bold black text. Below this, there are two input fields: the first is labeled "Username" and the second is labeled "Password". Both labels are in a standard black font, and the input fields are white rectangles with thin gray borders. At the bottom of the window, there are two buttons: "Login" on the left and "Create Account" on the right. Both buttons are gray with a slight gradient and a thin border.

When the system is started the user will be prompted for their user name and password. If they have forgotten either username or password they will be prompted to make a new account. The user may login if the details are correct and will be brought to the admin/user dashboard. If a user does not have an account they have the option of creating one. They may also cancel the screen and shut down the program.

## Create Account



The image shows a screenshot of a web application window titled "Admin Dashboard". Inside the window, there is a form titled "Please enter in your information to create a new account". The form contains two columns of input fields. The left column includes fields for Name, Username, Email, Department, and Phone Number. The right column includes fields for Password, Confirm Password, Security Question, Answer, and Account type. Below the Account type field, there is a text label that reads "normal" or "admin". At the bottom of the form, there are three buttons: Enter, Clear, and Cancel.

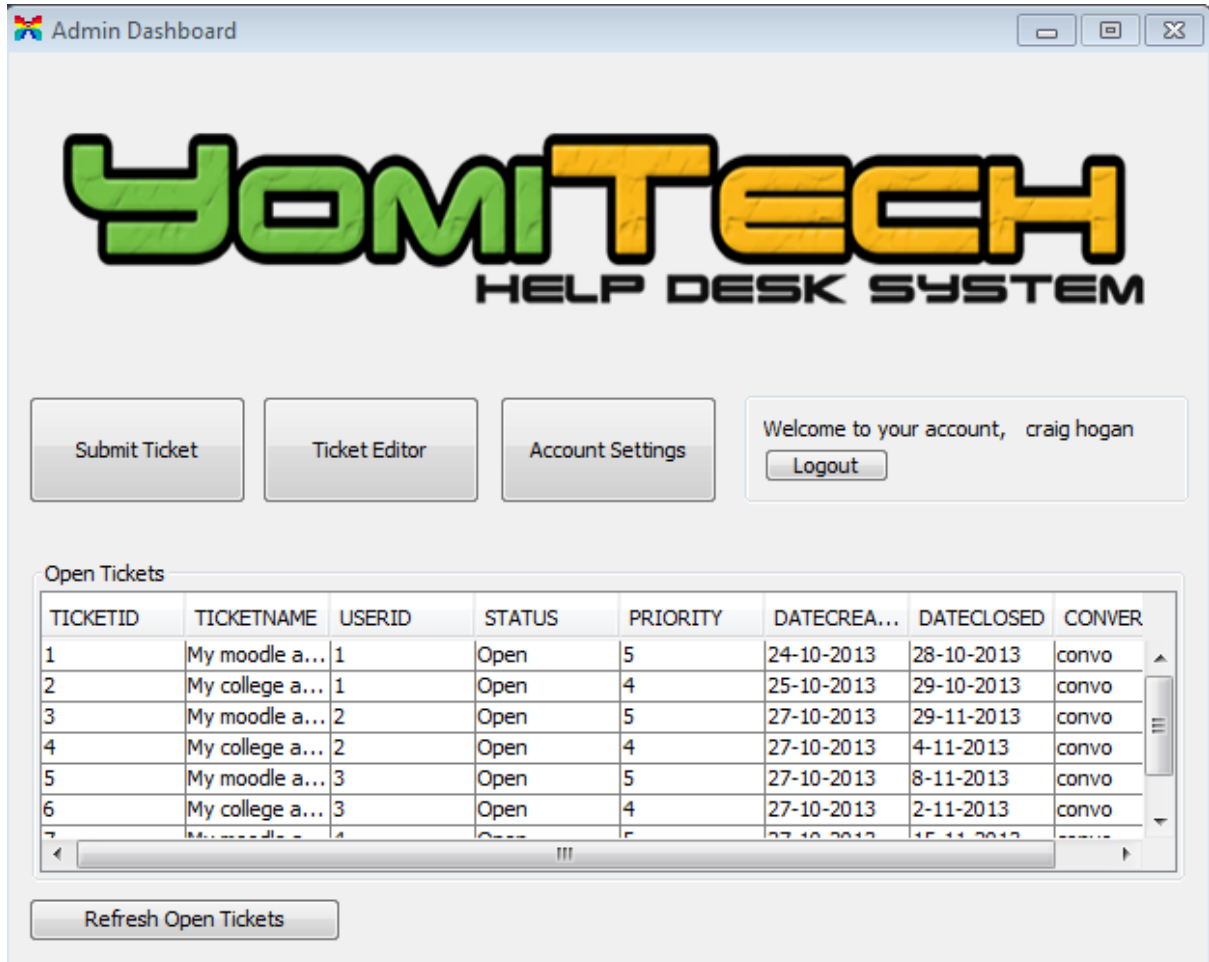
Field	Field
Name	Password
Username	Confirm Password
Email	Security Question
Department	Answer
Phone Number	Account type

"normal" or "admin"

Enter Clear Cancel

When a user clicks the create account button on the login page they are brought to the Create Account window. The user is prompted to enter in all their details including name, username, email, department, phone number, password, security question and answer. The details are checked and if correct they user may enter them and have their account created, they are then brought to their new accounts admin/user dashboard. If the details are incorrect the user is prompted to enter them in again. The user also has the option of entering in an admin or user account type. The user may also clear all fields or cancel to shut down the system.

## Admin Dashboard



The screenshot shows the YomiTECH Admin Dashboard. At the top, there's a window title bar with a small icon and the text "Admin Dashboard". Below this is the YomiTECH logo, which is stylized with "YOMI" in green and "TECH" in yellow, with "HELP DESK SYSTEM" in black text underneath. The dashboard features three main buttons: "Submit Ticket", "Ticket Editor", and "Account Settings". To the right of these buttons, there's a welcome message: "Welcome to your account, craig hogan" and a "Logout" button. Below the buttons, there's a section titled "Open Tickets" containing a table with the following data:

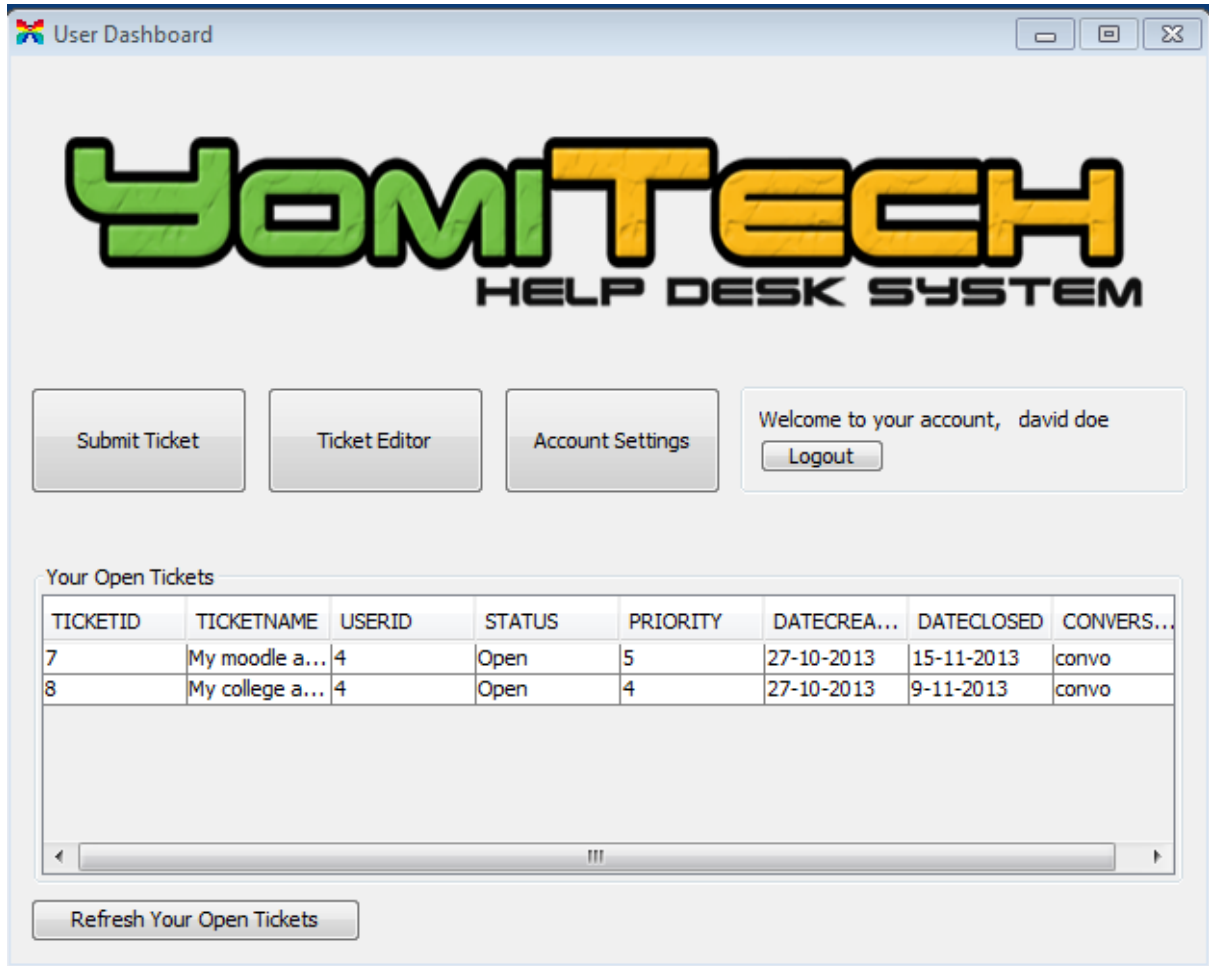
TICKETID	TICKETNAME	USERID	STATUS	PRIORITY	DATECREA...	DATECLOSED	CONVER
1	My moodle a...	1	Open	5	24-10-2013	28-10-2013	convo
2	My college a...	1	Open	4	25-10-2013	29-10-2013	convo
3	My moodle a...	2	Open	5	27-10-2013	29-11-2013	convo
4	My college a...	2	Open	4	27-10-2013	4-11-2013	convo
5	My moodle a...	3	Open	5	27-10-2013	8-11-2013	convo
6	My college a...	3	Open	4	27-10-2013	2-11-2013	convo
7	My moodle a...	4	Open	5	27-10-2013	15-11-2013	convo

Below the table, there's a "Refresh Open Tickets" button.

This is the admin dashboard. Admins have the option to submit a ticket, open ticket editor or open account settings. They can also logout to safely close the program. All open tickets in the database are displayed in the Open Tickets section of the dashboard. The open tickets database display may be refreshed to update the tickets in the database after a ticket is submitted or deleted.



## User Dashboard

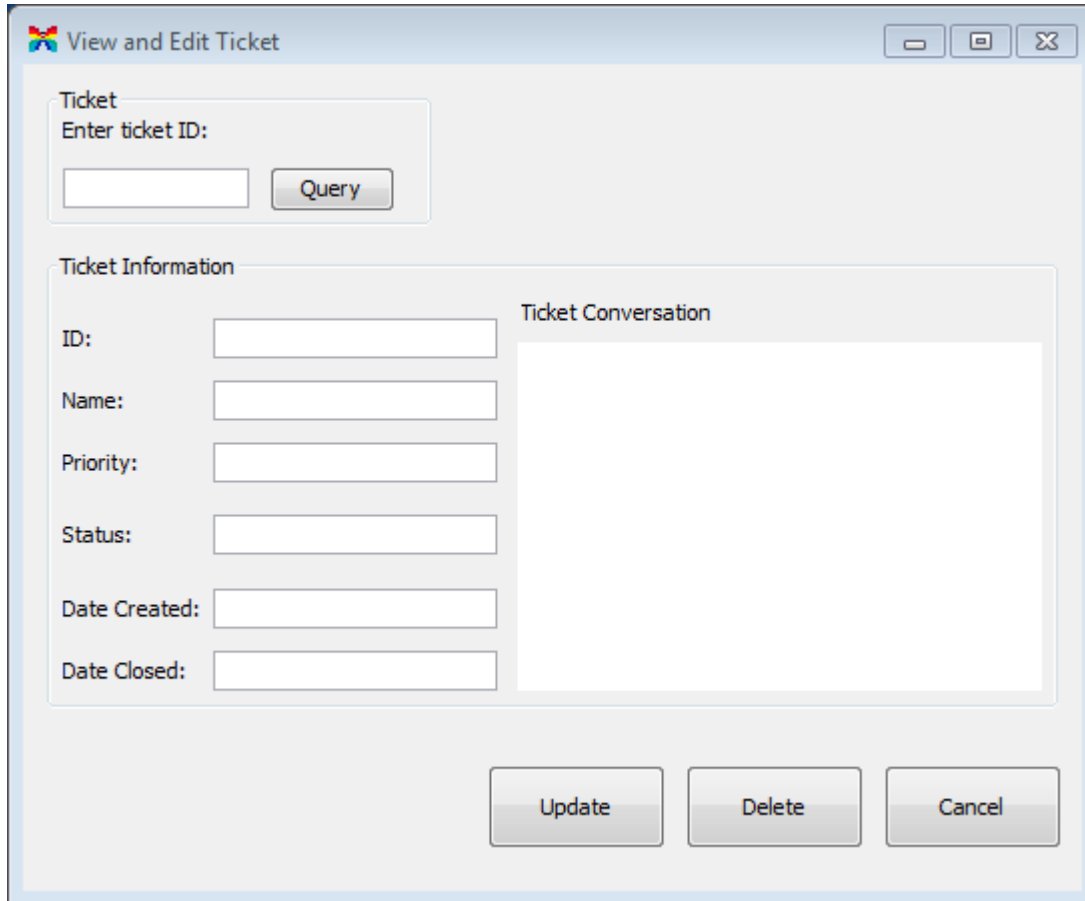


The screenshot shows a web application window titled "User Dashboard". At the top, there is a large logo for "YOMITECH HELP DESK SYSTEM" in a stylized green and yellow font. Below the logo, there are three buttons: "Submit Ticket", "Ticket Editor", and "Account Settings". To the right of these buttons, a welcome message says "Welcome to your account, david doe" with a "Logout" button below it. Below the buttons and message, there is a section titled "Your Open Tickets" containing a table with two rows of ticket data. At the bottom of the dashboard, there is a "Refresh Your Open Tickets" button.

TICKETID	TICKETNAME	USERID	STATUS	PRIORITY	DATECREA...	DATECLOSED	CONVERS...
7	My moodle a...	4	Open	5	27-10-2013	15-11-2013	convo
8	My college a...	4	Open	4	27-10-2013	9-11-2013	convo

This is the user dashboard. Users have the option to submit a ticket, open ticket editor or open account settings. They can also logout to safely logout to close the program. All the users open tickets are displayed in the Open Tickets section of the dashboard. The user may refresh their open tickets after they have submitted a ticket or deleted one of their tickets.

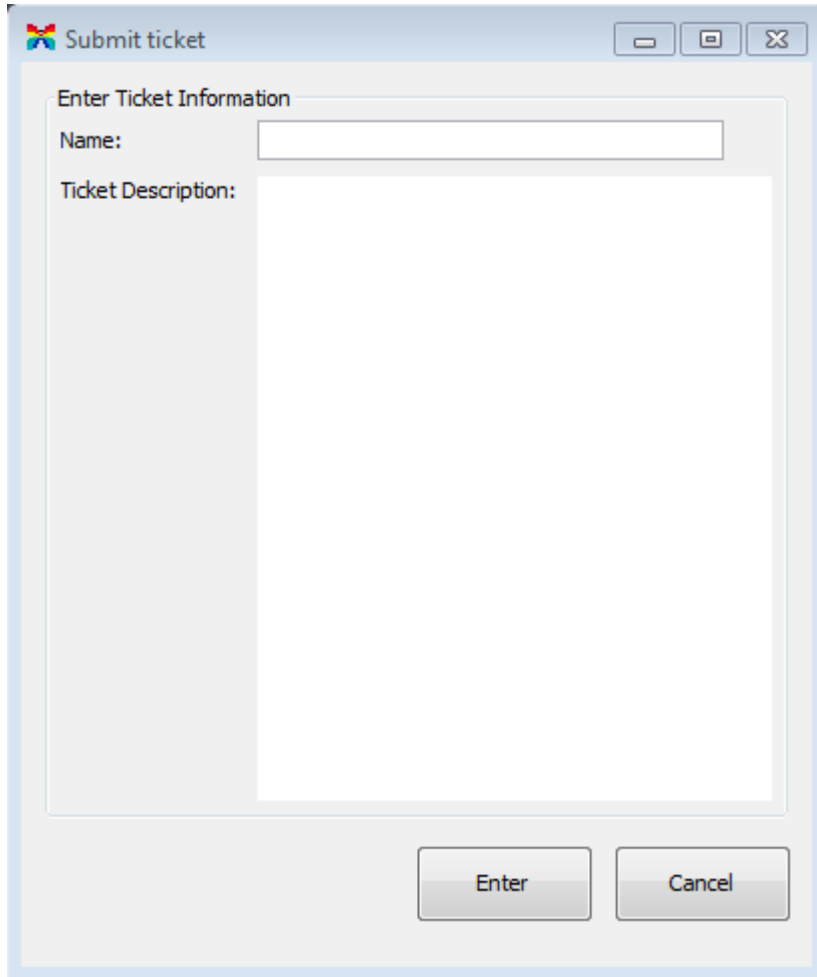
## View and Edit Ticket



The screenshot shows a software window titled "View and Edit Ticket". At the top left is a small logo. Below the title bar, there's a section labeled "Ticket" containing the text "Enter ticket ID:" followed by a text input field and a "Query" button. Below this is a larger section labeled "Ticket Information" which contains six labels with corresponding text input fields: "ID:", "Name:", "Priority:", "Status:", "Date Created:", and "Date Closed:". To the right of these fields is a section labeled "Ticket Conversation" which contains a large, empty text area. At the bottom of the window are three buttons: "Update", "Delete", and "Cancel".

The admin has the option to enter in any ticketID to retrieve it from the database. Users may also retrieve their tickets. The admin/user may answer/reply the ticket via the Ticket Conversation area. Users and admins may change the name of the ticket, priority, status, date created and date closed. The admin/user may also choose to cancel to close the window.

## Submit Ticket

A screenshot of a Windows-style dialog box titled "Submit ticket". The dialog box has a light blue title bar with standard minimize, maximize, and close buttons. The main content area is white and contains the text "Enter Ticket Information" at the top. Below this, there are two input fields: a single-line text box labeled "Name:" and a larger multi-line text box labeled "Ticket Description:". At the bottom of the dialog box, there are two buttons: "Enter" and "Cancel".

Submit ticket

Enter Ticket Information

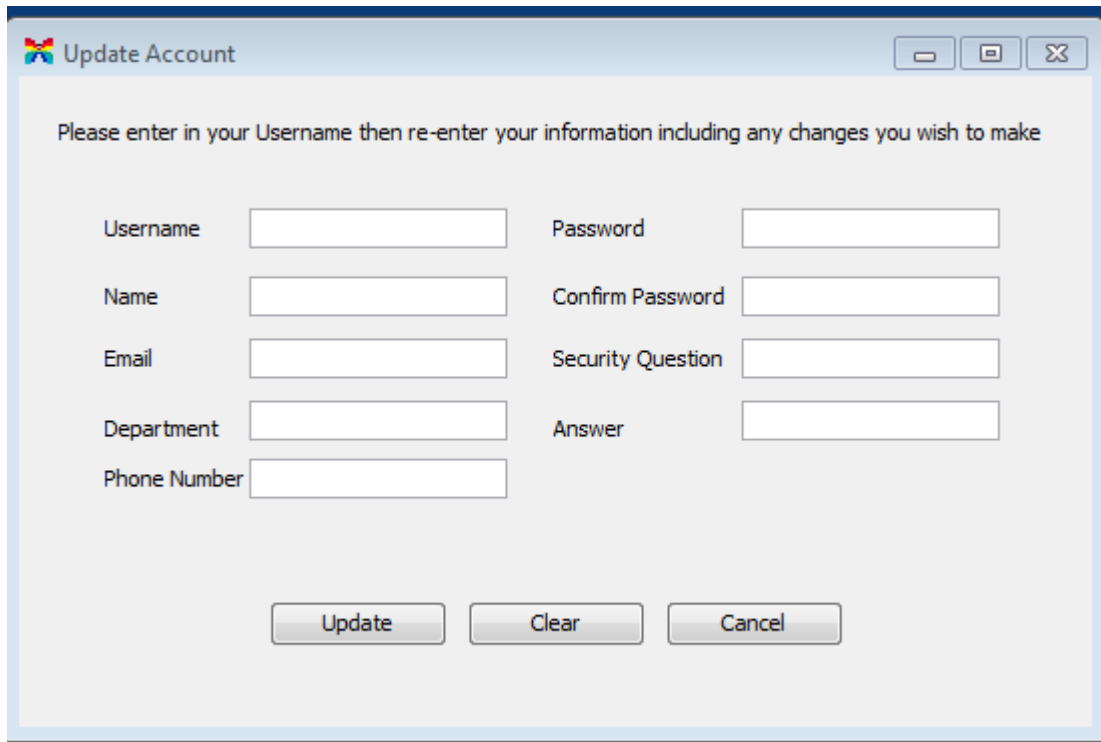
Name:

Ticket Description:

Enter Cancel

The user is prompted to enter in all ticket details before he adds a ticket into the system. When the user enters their ticket a screen pops up to let them know it has been added successfully to the database. User is required to enter in name of ticket, and the problem via ticket description. User may cancel to close window.

## Change Account Settings



The image shows a Windows-style dialog box titled "Update Account". It contains a message and several input fields for updating user information. At the bottom are three buttons: "Update", "Clear", and "Cancel".

Update Account

Please enter in your Username then re-enter your information including any changes you wish to make

Username	<input type="text"/>	Password	<input type="password"/>
Name	<input type="text"/>	Confirm Password	<input type="password"/>
Email	<input type="text"/>	Security Question	<input type="text"/>
Department	<input type="text"/>	Answer	<input type="text"/>
Phone Number	<input type="text"/>		

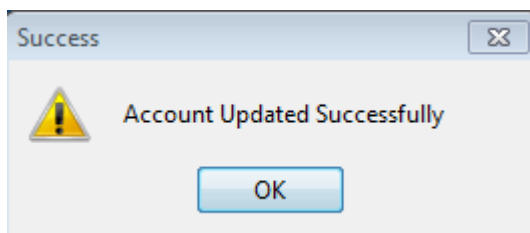
Update Clear Cancel

Admins and users may change their account settings via the Change Account Settings window. They may change all user settings that were entered upon account creation except their username. Options to clear all fields and exit window are also available. The user may cancel to close the window.

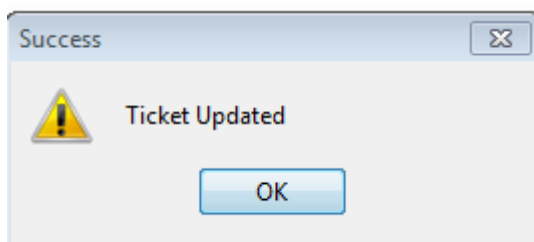
## **Output Design**

The output screens consist of 5 screens that will pop up and alert the user for each specific purpose.

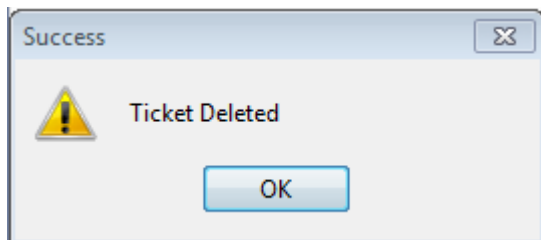
### **Account updated screen**



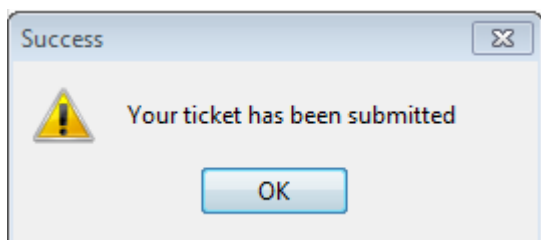
### **Ticket updated screen**



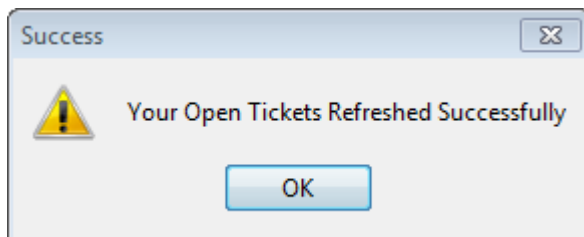
### **Ticket Deleted screen**



### **Ticket submitted screen**



### **Tickets Refreshed Successfully**



## **Database Design**

The database will be a simple layout. It will consist of 2 tables, the ticket table (with primary keys TicketID and TicketName) to hold all ticket information and a user table (with primary key UserID) which will hold all system user details.

Ticket	
PK	<u>TicketID</u>
PK	<u>TicketName</u>
	UserID Status Priority DateCreated DateClosed Conversation

creates

User	
PK	<u>UserID</u>
	UserName Userpassword SecurityQuestion SecurityAnsw AdminRights Email Fname PhoneNumber Department