

A+Interface Game Launcher - Project Proposal

Subject	Type	Status	Assignee	Version	Progress (%)
Games	Feature	New		v1.0.0	0
Description: <p>What is the goal of this task?: The Goal of this task is present and array of games for user's to launch.</p> <p>Is this a subtask of another task?: Yes, Game launcher.</p> <p>What subtasks define this task?: Launching a game.</p> <p>What nonuser interface functions does this task require?: This task requires the system to hide the game launcher and load the games .exe</p> <p>What kinds of inputs or actions does this task require from the user?: The game selected launches, the game launcher is minimized.</p> <p>What kinds of outputs or results occur by virtue of performing this task?: The main game launcher window is expected to open.</p> <p>What automatic actions does this task expect from the system?: This task expected the system to launch the selected game and minimize the game launcher.</p> <p>Is there a subtask that must come immediately before?: A user must select a game.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? The user must select a game.</p> <p>What, if any, primary entities are involved in this subtask? User, System.</p> <p>How can this task fail?: The game file doesn't exist or can't be found?</p> <p>How frequently is this task performed?: Anytime a user launches a game.</p> <p>How rigid is this task? This task is rigid. A game is launched, or it isnt.</p> <p>Are there any situational constraints?</p> <p>What are the specific usability expectations and how do we determine if we have satisfied these expectations?: The game selected is launched, the game launcher is minimized.</p>					
Login/Signup Form	Feature	New		v1.0.0	0
Description: <p>What is the goal of this task?: The Goal of this task is to open the appropriate Form(Window) that the user expects.</p> <p>Is this a subtask of another task?: No, This is the first task after opening the .exe.</p> <p>What subtasks define this task?: Logging in, Signing up, Logging in as Guest.</p> <p>What nonuser interface functions does this task require?: This task requires opening the</p>					

appropriate form window the user expects.

What kinds of inputs or actions does this task require from the user?: This task requires the user to open the .exe or click signup to change the form.

What kinds of outputs or results occur by virtue of performing this task?: The output of this task will open a form(window)

What automatic actions does this task expect from the system?: The system is expected to securely store the user's information, and create a UserId that can relationally connect data to the user. This task also manages opening the main platform/

What special characteristics of this task should we record?

Is there a subtask that must come immediately before?: Clicking the .exe file to launch our software.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? No, This is the first task.

What, if any, primary entities are involved in this subtask? User, System.

How can this task fail?: User cancels, User closes the form, User enters the wrong information and isn't allowed to proceed. The form isn't able to open the main platform.

How frequently is this task performed?: This task will happening anytime a user starts our software.

How rigid is this task? This task is pretty rigid. It will remain the same once built.

Are there any situational constraints? No.

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: A User has clear options on how to access our platform (Signing in, signing up, or signing in as guest). It will be satisfied if a User gets to the main software.

User Account	Feature	New		v1.0.0	0
Description:					
What is the goal of this task?: The Goal of this Task is for the system to give functionality for User Profile information stored in Database.					
Is this a subtask of another task?: No, this is the parent.					
What subtasks define this task?: Get User, Getlist of Users, Add User, Set User, Del User.					
What nonuser interface functions does this task require?: All of the tasks under this parent are non-user.					
What kinds of inputs or actions does this task require from the user?: This task requires a user to interface with a task that needs to access the database information.					
What kinds of outputs or results occur by virtue of performing this task?: The output of this task is user information based back to the client, or user information sent to the database.					

What automatic actions does this task expect from the system?: The system is expected to get, set, add, del when expected and to validate all

What special characteristics of this task should we record? This task is completely handled by the System.

Is there a subtask that must come immediately before?: The User must do a task that requires the system to get data from the database.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? This task is comes after the User forces the system to get data.

What, if any, primary entities are involved in this subtask? User, System, Database connection.

How can this task fail?: User information requested is invalid (UserId, Email, Username, Password etc...), User information sent is invalid (Email, Username, Password) <- Shouldn't happen.

How frequently is this task performed?: Ideally as little as possible. Unless information is edited or deleted, a .Get should happen on login. If system becomes real time or asynchronous this would happen more often.

How rigid is this task? This task is flexibly. This will be very modular on what can be set, added or gotten.

Are there any situational constraints?

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: High speed transactions, client should never request or send data that is invalid. Database prevents incorrect information from being stored. The user is prompted if an error accrues. Error should never happen.

User.Add	Feature	New		v1.0.0	0
<div><div>Description:</div><div><p>What is the goal of this task?: The Goal of this task is for the client to add a user using the database connection.</p><p>Is this a subtask of another task?: Yes, This is under User Account</p><p>What subtasks define this task?: There are currently no subtasks.</p><p>What nonuser interface functions does this task require?: This task is only handled by the client and database connection.</p><p>What kinds of inputs or actions does this task require from the user?: This requires a user to interact with a task that requires user information that the client is currently not storing.</p><p>What kinds of outputs or results occur by virtue of performing this task?: A user is created and a UserId is returned to the client. The User information is stored 'permanently'</p><p>What automatic actions does this task expect from the system?: The system is expected to securely store and create a user row.</p><p>What special characteristics of this task should we record?</p></div></div>					

Is there a subtask that must come immediately before?: A user interacts with a task that requires user information that the client is currently not storing.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user interacts with a task that requires user information that the client is currently not storing.

What, if any, primary entities are involved in this subtask? User, System.

How can this task fail?: This should never fail, as the client should filter requests that are invalid. This can fail if the email, or username already exists.

How frequently is this task performed?: Any user interacts with a task that requires user information that the client is currently not storing.

How rigid is this task? This task is pretty flexible. It can happen often and has a modular availability of information that can be sent and added.

Are there any situational constraints? A user already exists with this username or email.

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: A User is created on signup, and the information is stored and can be retrieved.

User.Get

Feature

New

v1.0.0

0

Description:

What is the goal of this task?: The Goal of this task is for the client to get user information from the database.

Is this a subtask of another task?: Yes, This is under User Account

What subtasks define this task?: There are currently no subtasks.

What nonuser interface functions does this task require?: This task is only handled by the client and database connection.

What kinds of inputs or actions does this task require from the user?: This requires a user to interact with a task that requires user information that the client is currently not storing.

What kinds of outputs or results occur by virtue of performing this task?: User information is retrieved and stored in client

What automatic actions does this task expect from the system?: The system is expected to securely retrieve and store data in the client

What special characteristics of this task should we record?

Is there a subtask that must come immediately before?: A user interacts with a task that requires user information that the client is currently not storing.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user interacts with a task that requires user information that the client is currently not storing.

What, if any, primary entities are involved in this subtask? User, System.

How can this task fail?: The retrieved information does not exist

How frequently is this task performed?: Any user interacts with a task that requires user information that the client is currently not storing.

How rigid is this task? This task is pretty flexible. It can happen often and has a modular availability of information that can be sent and retrieved.

Are there any situational constraints? The user doesn't have permission

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: The user information is retrieved and client info updated.

User.GetList(Admin)	Feature	New		v1.0.0	0
Description: Not implemented					
User.Set	Feature	New		v1.0.0	0
Description: <p>What is the goal of this task?: The Goal of this task is for the client to set user information using the database connection.</p> <p>Is this a subtask of another task?: Yes, This is under User Account</p> <p>What subtasks define this task?: There are currently no subtasks.</p> <p>What nonuser interface functions does this task require?: This task is only handled by the client and database connection.</p> <p>What kinds of inputs or actions does this task require from the user?: This requires a user to interact with a task that requires user information be updated.</p> <p>What kinds of outputs or results occur by virtue of performing this task?: A users information is updated in the database</p> <p>What automatic actions does this task expect from the system?: The system is expected to securely store and update user information.</p> <p>What special characteristics of this task should we record?</p> <p>Is there a subtask that must come immediately before?: A user interacts with a task that requires user information that the client update the stored info.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user interacts with a task that requires user information that the client be updated.</p> <p>What, if any, primary entities are involved in this subtask? User, System.</p> <p>How can this task fail?: This should never fail, as the client should filter requests that are invalid. This can fail if the client doesn't prevent invalid data from being sent</p>					

How frequently is this task performed?: Any user interacts with a task that requires user information that the client is currently not storing.

How rigid is this task? This task is pretty rigid. It can only be updated to specific values.

Are there any situational constraints? The information being set is invalid.

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: A user's information is updated.

User.Del	Feature	New		v1.0.0	0
<div><div>Description:</div><div><p>What is the goal of this task?: The Goal of this task is for the client to delete a user.</p><p>Is this a subtask of another task?: Yes, This is under User Account</p><p>What subtasks define this task?: There are currently no subtasks.</p><p>What nonuser interface functions does this task require?: This requires a user to click to delete their account. This task is handled by the client and database connection.</p><p>What kinds of inputs or actions does this task require from the user?: This requires a user to interact with a task that requires user information be deleted.</p><p>What kinds of outputs or results occur by virtue of performing this task?: A users information is deleted in the database</p><p>What automatic actions does this task expect from the system?: The system is expected to securely delete user information.</p><p>What special characteristics of this task should we record?</p><p>Is there a subtask that must come immediately before?: A user interacts with a task that requires user information be deleted.</p><p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user interacts with a task that requires user information be deleted.</p><p>What, if any, primary entities are involved in this subtask? User, System.</p><p>How can this task fail?: This should never fail, as the client should filter requests that are invalid.</p><p>How frequently is this task performed?: This should only happen once per user account and occurs rarely.</p><p>How rigid is this task? This task is pretty rigid. It can only be updated to specific values.</p><p>Are there any situational constraints?</p><p>What are the specific usability expectations and how do we determine if we have satisfied these expectations?: A user's information is securely deleted when requested.</p></div></div>					
User Signup	Feature	New		v1.0.0	0

Description:

What is the goal of this task?: The Goal of this Task is for a user to create an account for logging in and 'storing' game scores.

Is this a subtask of another task?: Yes, This is a subtask of Signup/Login Form

What subtasks define this task?: Enter Email, Enter Username, Enter Password, Re-Enter Password, Create User.

What nonuser interface functions does this task require?: This task requires validation of Username, Email, and Password. This is done by the interface.

What kinds of inputs or actions does this task require from the user?: This task requires a User to enter/choose a Username, enter their valid email, and chose a password that meets a certain criteria.

What kinds of outputs or results occur by virtue of performing this task?: The output of this task will be a User account being created for a user to store their scores.

What automatic actions does this task expect from the system?: The system is expected to securely store the user's information, and create a UserId that can relationally connect data to the user.

What special characteristics of this task should we record?

Is there a subtask that must come immediately before?: The User has the option to skip this task and Login, or sign-in as Guest.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? This task co-exists with Login and Sign-In as Guest.

What, if any, primary entities are involved in this subtask? User, System, Key-press, System Verification, System User Creation.

How can this task fail?: User cancels, User closes the form, User enters the wrong information and isn't allowed to proceed.

How frequently is this task performed?: Ideally once per user(Alternate accounts). Doesn't need to be performed at all if User want's to be a Guest.

How rigid is this task? This task is flexibly rigid. It will always be the same information and regex required, but will deviate and have different information from a User entered each time.

Are there any situational constraints?

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: Information required is clear, incorrect information is displayed well, and this is satisfied if a User can intuitively create and account.

Enter Email

Task

New

v1.0.0

0

Description:

Press Character Key(s)	Task	New		v1.0.0	0
<p>Description:</p> <p>Email will take any letter character both capital and miniscule.</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to input any letter character as part of their email.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Email.</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will run a check to see if the character can be accepted.</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to press any letter key on their keyboard.</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>The output will be the letter(s) appearing in the email textbox .</p> <p>What automatic actions does this task expect from the system?</p> <p>If an invalid letter is entered, the help icon should light up.</p> <p>What special characteristics of this task should we record?</p> <p>Users may try to enter characters with accents or from other languages.</p> <p>Is there a subtask that must come immediately before?</p> <p>User should navigate to the login/signup form before entering any characters.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?</p> <p>No.</p> <p>What, if any, primary entities are involved in this subtask?</p> <p>The user's computer and its components/peripherals.</p> <p>How can this task fail?</p> <p>If the user does not have a method of inputting letters, the task fails.</p> <p>How frequently is this task performed?</p> <p>This task should be performed as often as necessary during the sign up process, which will occur about once per user.</p> <p>How rigid is this task?</p> <p>The user either presses a key or doesn't. It doesn't get much more rigid than that.</p> <p>Are there any situational constraints?</p>					

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can enter a capital or miniscule letter of the English alphabet at their own leisure.

Press Accepted Symbol Key(s)

Task

New

v1.0.0

0

Description:

Menus

Feature

New

v1.0.0

0

Description:

Menu selector for Games, Profile, Leaderboard.

Press Tab or Enter Key

Task

New

v1.0.0

0

Description:

Profile

Feature

New

v1.0.0

0

Description:

What is the goal of this task?: For the user to see and update their info.

Is this a subtask of another task?: Yes, Game Launcher.

What subtasks define this task?: None.

What nonuser interface functions does this task require?: For the client to show user info and allow updating and deleting.

What kinds of inputs or actions does this task require from the user?: This requires the user to use their eyes to see their information, or click edit buttons for any values they'd like to modify.

What kinds of outputs or results occur by virtue of performing this task?: The user see's their information and can update it.

What automatic actions does this task expect from the system?: For the client to show the user their information, or update it.

What special characteristics of this task should we record?

Is there a subtask that must come immediately before?: Game Launcher.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user must click their profile and view or update it.

What, if any, primary entities are involved in this subtask? User, System.

How can this task fail?: The information updated is invalid.

How frequently is this task performed?: Anytime a user wants to update or view their information.

How rigid is this task? This task is flexibly rigid. the user can see their data which is static until updated.

Are there any situational constraints?

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: The user see's their information and can update it.

Leaderboard	Feature	New		v1.0.0	0
<div><div>Description:</div><div>What is the goal of this task?: For the user's to view the global leaderboard for all games.</div><div>Is this a subtask of another task?: Yes, Game Launcher.</div><div>What subtasks define this task?: None.</div><div>What nonuser interface functions does this task require?: For the client to show user scores.</div><div>What kinds of inputs or actions does this task require from the user?: This requires the user to use their eyes to see the scores.</div><div>What kinds of outputs or results occur by virtue of performing this task?: The user see's the scores.</div><div>What automatic actions does this task expect from the system?: For the client to show the user the scores.</div><div>What special characteristics of this task should we record?</div><div>Is there a subtask that must come immediately before?: Game Launcher.</div><div>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user must click their leaderboards.</div><div>What, if any, primary entities are involved in this subtask? User, System.</div><div>How can this task fail?: The scores are not valid from database. No scores.</div><div>How frequently is this task performed?: Anytime a user clicks on leaderboards.</div><div>How rigid is this task? This task is flexibly rigid. The ability to get the leaderboard is rigid. The scores are flexible</div><div>Are there any situational constraints?</div><div>What are the specific usability expectations and how do we determine if we have satisfied these expectations?: The user sees the leaderboard.</div></div>					
Press Backspace or Delete Key	Task	New		v1.0.0	0
<div><div>Description:</div></div>					

Game Launcher	Feature	New		v1.0.0	0
Description: What is the goal of this task?: The Goal of this task is to load the main game launcher window. Is this a subtask of another task?: Yes, Login/Signup What subtasks define this task?: Games, Profile, LeaderBoard. What nonuser interface functions does this task require?: This task requires the previous window to open this window and populate the login data. What kinds of inputs or actions does this task require from the user?: This requires a user to login or login as guest. What kinds of outputs or results occur by virtue of performing this task?: The main game launcher window is expected to open. What automatic actions does this task expect from the system?: For the launcher window to open and the login, signup pages hide/delete. What special characteristics of this task should we record? Is there a subtask that must come immediately before?: A user must login with their credentials or sign in as guest. In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? A user must login with their credentials or sign in as guest. What, if any, primary entities are involved in this subtask? User, System. How can this task fail?: This window should never open if a user doesn't enter a valid login How frequently is this task performed?: Anytime a user logs in with their credentials or as guest. How rigid is this task? This task is rigid. This task can only happen two ways. Are there any situational constraints? What are the specific usability expectations and how do we determine if we have satisfied these expectations?: The main launcher opens and is useable.					
Press any Other Key	Task	New		v1.0.0	0
Description:					
Click/Hover on Information Icon	Task	New		v1.0.0	0
Description:					
Enter Username	Task	New		v1.0.0	0
Description: User is prompted with:					

- textbox to enter username
- label for the textbox
- hoverable/clickable information icon

Press Character Key(s)	Task	New		v1.0.0	0
Description:	Username will take any letter character both capital and miniscule.				
	What is the goal of this task?				
	The goal of this task is to allow the user to input any letter character as part of their unique username.				
	Is this a subtask of another task?				
	This task is a subtask of the task Enter Username .				
	What subtasks define this task?				
	This task has no subtasks.				
	What nonuser interface functions does this task require?				
	This task will run a check to see if the character can be accepted.				
	What kinds of inputs or actions does this task require from the user?				
	This task requires the user to press any letter key on their keyboard.				
	What kinds of outputs or results occur by virtue of performing this task?				
	The output will be the letter(s) appearing in the username textbox .				
	What automatic actions does this task expect from the system?				
	If an invalid letter is entered, the help icon should light up.				
	What special characteristics of this task should we record?				
	Users may try to enter characters with accents or from other languages.				
	Is there a subtask that must come immediately before?				
	User should navigate to the login/signup form before entering any characters.				
	In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?				
	No.				
	What, if any, primary entities are involved in this subtask?				
	The user's computer and its components/peripherals.				
	How can this task fail?				
	If the user does not have a method of inputting letters, the task fails.				

How frequently is this task performed?
This task should be performed as often as necessary during the sign up process, which will occur about once per user.
How rigid is this task?
The user either presses a key or doesn't. It doesn't get much more rigid than that.
Are there any situational constraints?
None
What are the specific usability expectations and how do we determine if we have satisfied these expectations?
We will have satisfied these conditions if the user can enter a capital or miniscule letter of the English alphabet at their own leisure.

Press Accepted Symbols Key(s)	Task	New		v1.0.0	0
<div><div>Description:</div><div>Username will take any number, underscore, or period character</div><div>What is the goal of this task?</div><div>The goal of this task is to allow the user to input any accepted symbol as part of their unique username.</div><div>Is this a subtask of another task?</div><div>This task is a subtask of the task Enter Username .</div><div>What subtasks define this task?</div><div>This task has no subtasks.</div><div>What nonuser interface functions does this task require?</div><div>This task will run a check to see if the symbol can be accepted.</div><div>What kinds of inputs or actions does this task require from the user?</div><div>This task requires the user to press any symbol key on their keyboard.</div><div>What kinds of outputs or results occur by virtue of performing this task?</div><div>The output will be the symbols(s) appearing in the username textbox .</div><div>What automatic actions does this task expect from the system?</div><div>If an invalid symbol is entered, the help icon should light up.</div><div>What special characteristics of this task should we record?</div><div>Users may try to enter symbols from other languages.</div><div>Is there a subtask that must come immediately before?</div><div>User should navigate to the login/signup form before entering any characters.</div></div>					

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can enter an accepted symbol at their own leisure.

Press Tab or Enter Key	Task	New		v1.0.0	0
<p>Description:</p> <p>Pressing 'tab' or 'enter' will move the focus to the next text box in line</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to 'tab' to the next line, in this case, password line</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Username .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will run a check to see if the input can be accepted and will switch focus to the next textbox.</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to press the tab or enter key on their keyboard.</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p>					

The result will be the switching of focus from the username textbox to the password textbox.

What automatic actions does this task expect from the system?

The focus should be switched.

What special characteristics of this task should we record?

Users may miss-press keys.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this, and should probably completely fill in their username before pressing tab or enter.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

If the tab orders are not assigned, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can tab to the password textbox.

Press Backspace or Delete Key	Task	New		v1.0.0	0
Description: <p>Pressing backspace or delete will perform its intended function</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to delete characters if they've made a mistake.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Username .</p> <p>What subtasks define this task?</p>					

This task has no subtasks.

What nonuser interface functions does this task require?

This task will run a check to see if the input can be accepted and will either backspace or delete depending on the input.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press the tab or enter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The result will be the elimination of a character from the username textbox.

What automatic actions does this task expect from the system?

Character should disappear from textbox

What special characteristics of this task should we record?

Users may miss-press keys.

If there is no character in the textbox, nothing will happen.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can erase a character from the username textbox.

Press Any Other Key	Task	New		v1.0.0	0
Description: <p>Any other key press will not appear in the textbox. An error noise will sound and the information will display from the icon</p> <p>What is the goal of this task?</p> <p>The goal of this task is to deny the user a chance to input any invalid characters.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Username .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will run a check to see if the input can be accepted and will not allow the user to input an invalid character.</p> <p>If the character is invalid, an error noise will play and the information icon will blink.</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to press any invalid key on their keyboard.</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>The result will be an error noise and the information box blinking, with a tip popping up to aide the user</p> <p>What automatic actions does this task expect from the system?</p> <p>Character should not appear in textbox.</p> <p>Noise will sound.</p> <p>Information tool tip will pop up informing the user of their error</p> <p>What special characteristics of this task should we record?</p> <p>Users may miss-press keys.</p> <p>Is there a subtask that must come immediately before?</p> <p>User should navigate to the login/signup form before this.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?</p> <p>No.</p> <p>What, if any, primary entities are involved in this subtask?</p> <p>The user's computer and its components/peripherals.</p> <p>How can this task fail?</p> <p>If the user does not have a method of inputting symbols, the task fails.</p> <p>If the user inputs a correct key, the task fails.</p>					

How frequently is this task performed?
 This task should be performed as less often as possible during the sign up process, [the sign up] will occur about once per user.
 How rigid is this task?
 The user either presses a key or doesn't. It doesn't get much more rigid than that.
 Are there any situational constraints?
 None
 What are the specific usability expectations and how do we determine if we have satisfied these expectations?
 We will have satisfied these conditions if the user cannot input a foreign character into the username textbox.

Click/Hover on Information Icon	Task	New		v1.0.0	0
Description: <p>Textbox appears which informs the user of the accepted characters</p> <p>What is the goal of this task?</p> <p>The goal of this task is to display to the user which characters can be accepted in the username textbox</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Username .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will set the tool tip to be displayed</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to hover their mouse over or click the icon</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>Tooltip should be displayed while the users mouse is hovering over the icon.</p> <p>What automatic actions does this task expect from the system?</p> <p>Tool tip should be displayed.</p> <p>If user moves their mouse away, the tool tip should disappear.</p> <p>What special characteristics of this task should we record?</p> <p>Users may accidentally hover over the icon, restricting their view of the username textbox</p> <p>Is there a subtask that must come immediately before?</p>					

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

Typing in any of the textboxes can co-occur with this task.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of hovering over the icon, the task fails

How frequently is this task performed?

This task should be performed as often as needed during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user may accidentally move their mouse over the icon, triggering the tool tip

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can hover over the information box and have a tool tip displayed to them

Enter Password	Task	New		v1.0.0	0
Description: <div> User is prompted with: <ul style="list-style-type: none"> • textbox to enter password • label for the textbox • hoverable/clickable information icon </div>					
Press Character Key(s)	Task	New		v1.0.0	0
Description: <div> Password will take any letter character both capital and miniscule <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to input any letter character as part of their unique username.</p> <p>Is this a subtask of another task?</p> </div>					

This task is a subtask of the task Enter Password .

What subtasks define this task?

This task has no subtasks.

What nonuser interface functions does this task require?

This task will run a check to see if the character can be accepted.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press any letter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The output will be the placeholders for the letter(s) appearing in the password textbox .

What automatic actions does this task expect from the system?

If an invalid letter is entered, the help icon should light up.

What special characteristics of this task should we record?

Users may try to enter characters with accents or from other languages.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before entering any characters.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting letters, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, which will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can enter a capital or miniscule letter of the English alphabet at their own leisure.

Ensure Password Cannot Be Read	Task	New		v1.0.0	0
<p>Description:</p> <p>On key press, the character will not show in the password textbox, it will be replaced with a dot/asterisk to ensure privacy</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to input any character and keep it private.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Password.</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will replace any valid character with a placeholder.</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to press any letter key on their keyboard.</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>The output will be the placeholder(s) appearing in the password textbox .</p> <p>What automatic actions does this task expect from the system?</p> <p>The inputted character should be made sure never to be displayed in the password textbox, only the placeholder character.</p> <p>What special characteristics of this task should we record?</p> <p>Users may not be able to remember what key they had previously pressed.</p> <p>Is there a subtask that must come immediately before?</p> <p>User should navigate to the login/signup form before entering any characters.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?</p> <p>No.</p> <p>What, if any, primary entities are involved in this subtask?</p> <p>The user's computer and its components/peripherals.</p> <p>How can this task fail?</p> <p>If the user does not have a method of inputting letters, the task fails.</p> <p>If the inputted character itself is displayed, the task fails.</p> <p>How frequently is this task performed?</p> <p>This task should be performed as often as necessary during the sign up process, which will occur about once per user.</p>					

How rigid is this task?

This task is not handled by the user.

Are there any situational constraints?

Heavy privacy constraints

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user cannot see what they have entered in the password textbox and their privacy is protected.

Press Accepted Symbols Key(s)	Task	New		v1.0.0	0
<div><div>Description:</div><div><p>Password will take any number, as well as any symbols with the exception of quotation marks and colons/semicolons, as well as mathematical and bitwise operators</p><p>What is the goal of this task?</p><p>The goal of this task is to allow the user to input any accepted symbol as part of their unique username.</p><p>Is this a subtask of another task?</p><p>This task is a subtask of the task Enter Password .</p><p>What subtasks define this task?</p><p>This task has no subtasks.</p><p>What nonuser interface functions does this task require?</p><p>This task will run a check to see if the symbol can be accepted.</p><p>What kinds of inputs or actions does this task require from the user?</p><p>This task requires the user to press any symbol key on their keyboard.</p><p>What kinds of outputs or results occur by virtue of performing this task?</p><p>The output will be the placeholder for the symbols(s) appearing in the password textbox .</p><p>What automatic actions does this task expect from the system?</p><p>If an invalid symbol is entered, the help icon should light up.</p><p>What special characteristics of this task should we record?</p><p>Users may try to enter symbols from other languages.</p><p>Is there a subtask that must come immediately before?</p><p>User should navigate to the login/signup form before entering any characters.</p><p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?</p><p>No.</p></div></div>					

What, if any, primary entities are involved in this subtask?
The user's computer and its components/peripherals.
How can this task fail?
If the user does not have a method of inputting symbols, the task fails.
How frequently is this task performed?
This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.
How rigid is this task?
The user either presses a key or doesn't. It doesn't get much more rigid than that.
Are there any situational constraints?
None
What are the specific usability expectations and how do we determine if we have satisfied these expectations?
We will have satisfied these conditions if the user can enter an accepted symbol at their own leisure.

Press Tab or Enter Key

Task

New

v1.0.0

0

Description:

Pressing 'tab' or 'enter' will move the focus to the next text box in line
What is the goal of this task?
The goal of this task is to allow the user to 'tab' to the next line, in this case, re-enter password line
Is this a subtask of another task?
This task is a subtask of the task Enter Password .
What subtasks define this task?
This task has no subtasks.
What nonuser interface functions does this task require?
This task will run a check to see if the input can be accepted and will switch focus to the next textbox.
What kinds of inputs or actions does this task require from the user?
This task requires the user to press the tab or enter key on their keyboard.
What kinds of outputs or results occur by virtue of performing this task?
The result will be the switching of focus from the password textbox to the re-enter password textbox.
What automatic actions does this task expect from the system?
The focus should be switched.

What special characteristics of this task should we record?
Users may miss-press keys.
Is there a subtask that must come immediately before?
User should navigate to the login/signup form before this, and should probably completely fill in their password before pressing tab or enter.
In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?
No.
What, if any, primary entities are involved in this subtask?
The user's computer and its components/peripherals.
How can this task fail?
If the user does not have a method of inputting symbols, the task fails.
If the tab orders are not assigned, the task fails.
How frequently is this task performed?
This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.
How rigid is this task?
The user either presses a key or doesn't. It doesn't get much more rigid than that.
Are there any situational constraints?
None
What are the specific usability expectations and how do we determine if we have satisfied these expectations?
We will have satisfied these conditions if the user can tab to the re-enter password textbox.

Press Backspace or Delete Key	Task	New		v1.0.0	0
Description: Pressing backspace or delete will perform its intended function What is the goal of this task? The goal of this task is to allow the user to delete characters if they've made a mistake. Is this a subtask of another task? This task is a subtask of the task Enter Password . What subtasks define this task? This task has no subtasks. What nonuser interface functions does this task require? This task will run a check to see if the input can be accepted and will either backspace or delete					

depending on the input.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press the tab or enter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The result will be the elimination of a character from the password textbox.

What automatic actions does this task expect from the system?

Character should disappear from textbox

What special characteristics of this task should we record?

Users may miss-press keys.

If there is no character in the textbox, nothing will happen.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can erase a character from the password textbox.

Press Any Other Key

Task

New

v1.0.0

0

Description:

Any other key press will not appear in the textbox. An error noise will sound and the information

will display from the icon

What is the goal of this task?

The goal of this task is to deny the user a chance to input any invalid characters.

Is this a subtask of another task?

This task is a subtask of the task Enter Password .

What subtasks define this task?

This task has no subtasks.

What nonuser interface functions does this task require?

This task will run a check to see if the input can be accepted and will not allow the user to input an invalid character.

If the character is invalid, an error noise will play and the information icon will blink.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press any invalid key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The result will be an error noise and the information box blinking, with a tip popping up to aide the user

What automatic actions does this task expect from the system?

Character should not appear in textbox.

Noise will sound.

Information tool tip will pop up informing the user of their error

What special characteristics of this task should we record?

Users may miss-press keys.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

If the user inputs a correct key, the task fails.

How frequently is this task performed?

This task should be performed as less often as possible during the sign up process, [the sign up]

will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user cannot input a foreign character into the password textbox.

Click/Hover on Information Icon	Task	New		v1.0.0	0
<p>Description:</p> <p>Textbox appears which informs the user of the accepted characters</p> <p>What is the goal of this task?</p> <p>The goal of this task is to display to the user which characters can be accepted in the password textbox</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Enter Password .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will set the tool tip to be displayed</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to hover their mouse over or click the icon</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>Tooltip should be displayed while the users mouse is hovering over the icon.</p> <p>What automatic actions does this task expect from the system?</p> <p>Tool tip should be displayed.</p> <p>If user moves their mouse away, the tool tip should disappear.</p> <p>What special characteristics of this task should we record?</p> <p>Users may accidentally hover over the icon, restricting their view of the password textbox</p> <p>Is there a subtask that must come immediately before?</p> <p>User should navigate to the login/signup form before this.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-</p>					

occur?

Typing in any of the textboxes can co-occur with this task.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of hovering over the icon, the task fails

How frequently is this task performed?

This task should be performed as often as needed during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user may accidentally move their mouse over the icon, triggering the tool tip

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can hover over the information box and have a tool tip displayed to them

Re-Enter Password	Task	New		v1.0.0	0
Description: <p>User is prompted with:</p> <ul style="list-style-type: none"> • textbox to enter password • label for the textbox 					
Press Character Key(s)	Task	New		v1.0.0	0
Description: <p>Password will take any letter character both capital and miniscule</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to input any letter character as part of their unique username.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Re-Enter Password .</p> <p>What subtasks define this task?</p>					

This task has no subtasks.

What nonuser interface functions does this task require?

This task will run a check to see if the character can be accepted.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press any letter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The output will be the placeholders for the letter(s) appearing in the re-enter password textbox .

What automatic actions does this task expect from the system?

If an invalid letter is entered, the help icon should light up.

What special characteristics of this task should we record?

Users may try to enter characters with accents or from other languages.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before entering any characters.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting letters, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, which will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can enter a capital or miniscule letter of the English alphabet at their own leisure.

Ensure Password Cannot Be Read

Task

New

v1.0.0

0

Description:

On key press, the character will not show in the re-enter password textbox, it will be replaced with a dot/asterisk to ensure privacy

What is the goal of this task?

The goal of this task is to allow the user to input any character and keep it private.

Is this a subtask of another task?

This task is a subtask of the task Re-Enter Password.

What subtasks define this task?

This task has no subtasks.

What nonuser interface functions does this task require?

This task will replace any valid character with a placeholder.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press any letter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The output will be the placeholder(s) appearing in the re-enter password textbox .

What automatic actions does this task expect from the system?

The inputted character should be made sure never to be displayed in the re-enter password textbox, only the placeholder character.

What special characteristics of this task should we record?

Users may not be able to remember what key they had previously pressed.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before entering any characters.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting letters, the task fails.

If the inputted character itself is displayed, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, which will occur about once per user.

How rigid is this task?

This task is not handled by the user.

Are there any situational constraints?
Heavy privacy constraints
What are the specific usability expectations and how do we determine if we have satisfied these expectations?
We will have satisfied these conditions if the user cannot see what they have entered in the re-enter password textbox and their privacy is protected.

Press Accepted Symbols Key(s)	Task	New		v1.0.0	0
<div><div>Description:</div><div><p>Password will take any number, as well as any symbols with the exception of quotation marks and colons/semicolons, as well as mathematical and bitwise operators</p><p>What is the goal of this task?</p><p>The goal of this task is to allow the user to input any accepted symbol as part of their unique username.</p><p>Is this a subtask of another task?</p><p>This task is a subtask of the task Re-Enter Password .</p><p>What subtasks define this task?</p><p>This task has no subtasks.</p><p>What nonuser interface functions does this task require?</p><p>This task will run a check to see if the symbol can be accepted.</p><p>What kinds of inputs or actions does this task require from the user?</p><p>This task requires the user to press any symbol key on their keyboard.</p><p>What kinds of outputs or results occur by virtue of performing this task?</p><p>The output will be the placeholder for the symbols(s) appearing in the re-enter password textbox .</p><p>What automatic actions does this task expect from the system?</p><p>If an invalid symbol is entered, the help icon should light up.</p><p>What special characteristics of this task should we record?</p><p>Users may try to enter symbols from other languages.</p><p>Is there a subtask that must come immediately before?</p><p>User should navigate to the login/signup form before entering any characters.</p><p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?</p><p>No.</p><p>What, if any, primary entities are involved in this subtask?</p></div></div>					

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can enter an accepted symbol at their own leisure.

Press Tab or Enter Key	Task	New		v1.0.0	0
<p>Description:</p> <p>Pressing 'tab' or 'enter' will move the focus to the next tab order</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to 'tab' to the next line, in this case, Create User button</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Re-Enter Password .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will run a check to see if the input can be accepted and will switch focus to the Create User button.</p> <p>What kinds of inputs or actions does this task require from the user?</p> <p>This task requires the user to press the tab or enter key on their keyboard.</p> <p>What kinds of outputs or results occur by virtue of performing this task?</p> <p>The result will be the switching of focus from the re-enter password textbox to the Create User button.</p> <p>What automatic actions does this task expect from the system?</p> <p>The focus should be switched.</p>					

What special characteristics of this task should we record?
 Users may miss-press keys.
 Is there a subtask that must come immediately before?
 User should navigate to the login/signup form before this, and should probably completely fill in their password before pressing tab or enter.
 In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?
 No.
 What, if any, primary entities are involved in this subtask?
 The user's computer and its components/peripherals.
 How can this task fail?
 If the user does not have a method of inputting symbols, the task fails.
 If the tab orders are not assigned, the task fails.
 How frequently is this task performed?
 This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.
 How rigid is this task?
 The user either presses a key or doesn't. It doesn't get much more rigid than that.
 Are there any situational constraints?
 None
 What are the specific usability expectations and how do we determine if we have satisfied these expectations?
 We will have satisfied these conditions if the user can tab to the Create User button.

Press Backspace or Delete Key	Task	New		v1.0.0	0
Description: <p>Pressing backspace or delete will perform its intended function</p> <p>What is the goal of this task?</p> <p>The goal of this task is to allow the user to delete characters if they've made a mistake.</p> <p>Is this a subtask of another task?</p> <p>This task is a subtask of the task Re-Enter Password .</p> <p>What subtasks define this task?</p> <p>This task has no subtasks.</p> <p>What nonuser interface functions does this task require?</p> <p>This task will run a check to see if the input can be accepted and will either backspace or delete</p>					

depending on the input.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press the tab or enter key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The result will be the elimination of a character from the re-enter password textbox.

What automatic actions does this task expect from the system?

Character should disappear from textbox

What special characteristics of this task should we record?

Users may miss-press keys.

If there is no character in the textbox, nothing will happen.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

How frequently is this task performed?

This task should be performed as often as necessary during the sign up process, [the sign up] will occur about once per user.

How rigid is this task?

The user either presses a key or doesn't. It doesn't get much more rigid than that.

Are there any situational constraints?

None

What are the specific usability expectations and how do we determine if we have satisfied these expectations?

We will have satisfied these conditions if the user can erase a character from the re-enter password textbox.

Press Any Other Key

Task

New

v1.0.0

0

Description:

Any other key press will not appear in the textbox. An error noise will sound and the information

will display from the icon beside the password textbox

What is the goal of this task?

The goal of this task is to deny the user a chance to input any invalid characters.

Is this a subtask of another task?

This task is a subtask of the task Re-Enter Password .

What subtasks define this task?

This task has no subtasks.

What nonuser interface functions does this task require?

This task will run a check to see if the input can be accepted and will not allow the user to input an invalid character.

If the character is invalid, an error noise will play and the information icon will blink.

What kinds of inputs or actions does this task require from the user?

This task requires the user to press any invalid key on their keyboard.

What kinds of outputs or results occur by virtue of performing this task?

The result will be an error noise and the information box blinking, with a tip popping up to aide the user

What automatic actions does this task expect from the system?

Character should not appear in textbox.

Noise will sound.

Information tool tip will pop up informing the user of their error

What special characteristics of this task should we record?

Users may miss-press keys.

Is there a subtask that must come immediately before?

User should navigate to the login/signup form before this.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur?

No.

What, if any, primary entities are involved in this subtask?

The user's computer and its components/peripherals.

How can this task fail?

If the user does not have a method of inputting symbols, the task fails.

If the user inputs a correct key, the task fails.

How frequently is this task performed?

This task should be performed as less often as possible during the sign up process, [the sign up]

will occur about once per user.
How rigid is this task?
The user either presses a key or doesn't. It doesn't get much more rigid than that.
Are there any situational constraints?
None
What are the specific usability expectations and how do we determine if we have satisfied these expectations?
We will have satisfied these conditions if the user cannot input a foreign character into the re-enter password textbox.

Create User	Feature	New		v1.0.0	0
Description:					
User Login	Feature	New		v1.0.0	0
Description: <p>What is the goal of this task?: The Goal of this Task is for a user to enter their login credentials and gain access to the game launcher.</p> <p>Is this a subtask of another task?: Yes, This is a subtask of Signup/Login Form</p> <p>What subtasks define this task?: Enter Email or Enter Username, Enter Password.</p> <p>What nonuser interface functions does this task require?: This task requires validation of Username, Email, and Password. This is done by the interface.</p> <p>What kinds of inputs or actions does this task require from the user?: This task requires a User to enter a valid Username, email and password.</p> <p>What kinds of outputs or results occur by virtue of performing this task?: The output of this task will be a User gaining access to the main client.</p> <p>What automatic actions does this task expect from the system?: The system is expected to validate the credentials entered by the user.</p> <p>What special characteristics of this task should we record?</p> <p>Is there a subtask that must come immediately before?: The User has the option to skip this task and Signup, or sign-in as Guest.</p> <p>In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? This task co-exists with Sign up and Sign-In as Guest.</p> <p>What, if any, primary entities are involved in this subtask? User, System, Key-press, System Verification, System User Creation.</p> <p>How can this task fail?: User enters the wrong information and isn't allowed to proceed. User signs</p>					

up or logins in as guest.

How frequently is this task performed?: Ideally once per user(Alternate accounts). Doesn't need to be performed at all if User want's to be a Guest.

How rigid is this task? This task is flexibly rigid. It will always be the same information and regex required, but will deviate based on whose logging in.

Are there any situational constraints?

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: Information required is clear, incorrect information is displayed well, and this is satisfied if a User can intuitively login.

Enter Email	Task	New		v1.0.0	0
Description:					
Enter Password	Task	New		v1.0.0	0
Description:					
Save Login	Task	New		v1.0.0	0
Description:					
Verify Login	Task	New		v1.0.0	0
Description:					
Is Username connected to a UserId	Task	New		v1.0.0	0
Description:					
Is Password Sha or Md5 match UserId	Task	New		v1.0.0	0
Description:					
Guest Login	Feature	New		v1.0.0	0
Description:					
What is the goal of this task?: The Goal of this Task is for a user to login as a guest user and not requiring to signup or login with their own information.					
Is this a subtask of another task?: Yes, This is a subtask of Signup/Login Form					
What subtasks define this task?: None.					
What nonuser interface functions does this task require?: This task requires the interface to					

populate the user credentials and profile with 'Guest' credentials.

What kinds of inputs or actions does this task require from the user?: This task requires a User to click 'Signin as Guest'

What kinds of outputs or results occur by virtue of performing this task?: The output of this task will be a User gaining access to the main client as a guest.

What automatic actions does this task expect from the system?: The system is expected to bypass the login validation and populate vars with 'Guest' vars.

What special characteristics of this task should we record?

Is there a subtask that must come immediately before?: The User has the option to skip this task and Login, or sign-up.

In this subtree is there a task that this task is the immediate predecessor of, or a task that can co-occur? This task co-exists with SignIn and Signup.

What, if any, primary entities are involved in this subtask? User, System.

How can this task fail?: This task should not fail.

How frequently is this task performed?: Ideally once per user(Alternate accounts). Doesn't need to be performed at all if User want's to use their own credentials.

How rigid is this task? This task is flexibly. It can be avoid and the user can user their own credentials to login.

Are there any situational constraints?

What are the specific usability expectations and how do we determine if we have satisfied these expectations?: A user is able to access the main client without signing up or logging in.

Set Username = Guest	Task	New		v1.0.0	0
Description:					
Ignore Password	Task	New		v1.0.0	0
Description:					
Icons	Task	New		v1.0.0	0
Description:					
Array of Game display (3*n)	Task	New		v1.0.0	0
Description:					

Hover effect	Task	New		v1.0.0	0
Description:					
On-click	Task	New		v1.0.0	0
Description:					