**General Concept of Chad System**

1. When the system fail, the system will restart to the initial but will not influence player’s data
2. The order of the use case section is choice on the process of a player ean to play a chad game on the game system

**Use Cases**

**1.) Register - Alber**

|  |  |
| --- | --- |
| **Use Case Name** | Register Account |
| **Scope** | Chad Game System |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Player: wants to register on the platform to play the game |
| **Preconditions** | 1. Player should have a valid email id 2. Player should not be a registered member of the platform 3. System would check if the email is repeated, or it will reply for an error |
| **Post conditions** | Player should be able to have a new account and a nick name on the system |
| **Main Flow** | 1. Player accesses register tab on the platform 2. Player fills all the required fields to register 3. System validates the input fields and register the player on the platform 4. Player should enter the nickname, email, password and authorize the password. |
| **Alternative Flows** | 1a. Player enters invalid format for any of the required fields   1. Displays error 2. System prompt the player to re-enter the required field |
| **Special Requirements** | None |
| **Technology and Data Variations** | None |
| **Frequency of Occurrence** | Once per player |

**2.) Log in - Zach**

|  |  |
| --- | --- |
| **Use Case Name** | Log in |
| **Scope** | Chad game system |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Player: Wants to be able to be recognized by the system by logging in with his unique credentials. |
| **Preconditions** | 1. Player needs to have access to the login portal 2. Player need to get on login page 3. Player trying to log in needs to have an active, registered account |
| **Postconditions** | Player is logged into the system and now able to access system features. |
| **Main Flow** | 1. Player accesses log in portal and inputs username and password 2. System validates account name exists and password is exactly correct 3. Home page is displayed |
| **Alternative Flows** | 1a. Player forgot username or password:   1. Display error 2. Attempt to recover username or password   2a. Username and password do not exactly match   1. Display error 2. User tries different password |
| **Special Requirements** | None |
| **Technology and Data Variations** | Verify username and password:  1. Username exists &  2. Password matches username |
| **Frequency of Occurrence** | Whenever login is attempted |

**3.) Log out - Zach**

|  |  |
| --- | --- |
| **Use Case Name** | Log out |
| **Scope** | Chad game system |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Player: Wants to be able to log out of the system so his/her profile cannot be accessed by someone else. |
| **Preconditions** | 1. Player needs to currently be logged in |
| **Postconditions** | Player is logged out of the system and now unable to access system features. |
| **Main Flow** | 1. Logged in player logs out 2. Log out portal is displayed |
| **Alternative Flows** | None |
| **Special Requirements** | None |
| **Technology and Data Variations** | None |
| **Frequency of Occurrence** | Whenever logout is attempted |

**4.) Create new game - Shun**

|  |  |
| --- | --- |
| **Use Case Name** | Create new game |
| **Use Case No** | 4 |
| **Scope** | Chad game system |
| **Level** | User-goal |
| **Stakeholders** | Player: User should create a game that he or she can invite players. In fact the Chad system is the true creator which create a game situation for players. |
| **Primary Actor** | Player |
| **Pre-Condition(s)** | 1. User should already login to the platform |
| **Main Flow** | 1. Users can create the game by click "Create new game”  2. The Chad system will create a board and initialize the board  3. The Chad system will load the creator’s log to store the history |
| **Alternative Flow** | None |
| **Technology and Data Variations** | None |
| **Post-Condition(s)** | The game will be created and creator can already invite players |
| **Frequency of Occurrence** | Whenever player tries to create new game |
| **Special Requirements** | None |

**5.) Invite player - Martin**

|  |  |
| --- | --- |
| **Use Case Name** | Invite player |
| **Use Case No** | 5 |
| **Brief Description** | User has to register to start playing the game. |
| **Priority** | Critical to user |
| **Pre-Condition(s)** | 1. User should not be a registered member of application  2. User should have a valid email id |
| **Main Flow** | 1. User will click ‘Invite a new account’.  2. System will redirect User to sign up page.  3. User fills in required fields, and clicks ‘Register’ in order to proceed to next step.  4. System validates the fields, register the user and redirect User to Login Screen.  5. User can invite player by search from the online list. |
| **Alternative Flow** | 1. At step 3, if any of the information given by client is incorrect or incomplete, it will ask the client to enter it correctly. This in other use case |
| **Non-Functional Requirement** | None |
| **Post-Condition(s)** | User should be able to have his/her new account. |

**6.) Accept/Reject invitation - Martin**

|  |  |
| --- | --- |
| **Use Case Name** | Register |
| **Use Case No** | 1 |
| **Brief Description** | User has to register to start playing the game. |
| **Priority** | Critical to user |
| **Pre-Condition(s)** | 1. User should not be a registered member of application  2. User should have a valid email id |
| **Main Flow** | 1. User will click ‘Create a new account’.  2. System will redirect User to sign up page.  Two steps for search and invite, receive and reply  3. User fills in required fields, and clicks ‘Register’ in order to proceed to next step.  4. System validates the fields, register the user and redirect User to Login Screen. |
| **Alternative Flow** | 1. At step 3, if any of the information given by client is incorrect or incomplete, it will ask the client to enter it correctly.  2. User fills in required fields, and clicks ‘Register’ in order to proceed to next step. |
| **Non-Functional Requirement** | None |
| **Post-Condition(s)** | User should be able to have his/her new account. |

**7.) Quit game - Shun**

|  |  |
| --- | --- |
| **Use Case Name** | Quit game |
| **Use Case No** | 7 |
| **Scope** | Chad game system |
| **Level** | User-goal |
| **Primary Actor** | Players |
| **Stakeholders** | Players: who wants to quit the game |
| **Pre-Condition(s)** | 1. Players should already joined a game, at least a creator |
| **Main Flow** | 1. Player quits game  2. Game ends  3. Game is recorded as a lose for the quitting player  4. Both of the player will be informed that the game is ended. |
| **Alternative Flow** | None |
| **Technology and Data Variations** | None |
| **Post-Condition(s)** | The game will quit. All the player will out of the game. |
| **Special Requirements** | None |
| **Frequency of Occurrence** | Whenever player decides to quits game |

**8.) Unregister account - Alber**

|  |  |
| --- | --- |
| **Use Case Name** | Unregister Account |
| **Scope** | Chad Game System |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Player: wants to unregister from the platform |
| **Preconditions** | 1. Player should be a registered member of the platform 2. Player should be logged in 3. User should have an account which is already registered to the system 4. The user should be listed on the system |
| **Post conditions** | Player should be able to unregister account |
| **Main Flow** | 1. Player unregisters his/her account 2. System redirects the player to login page 3. All the game log or information will be deleted or store into backup system if player want to recover his or her account |
| **Alternative Flows** | None |
| **Special Requirements** | None |
| **Technology and Data Variations** | None |
| **Frequency of Occurrence** | Once per player |

**9.) View profile - Paul**

|  |  |
| --- | --- |
| **Use Case Name** | View Profile |
| **Scope** | Chad game system |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Player: Wants to see nickname and game history of themselves or another registered player. |
| **Preconditions** | 1. Player doing the viewing is a registered player |
| **Postconditions** | Desired profile is displayed |
| **Main Flow** | 1. Player requests to see another player’s profile 2. System validates requested profile exists 3. Profile is displayed. Profile contains nickname and game history 4. Player can search player on the list which provide by system. |
| **Alternative Flows** | 1a. Requested Profile does not exist:  1. Display error  2. An unexist account would not show on the list of registered players |
| **Special Requirements** | None |
| **Technology and Data Variations** | None |
| **Frequency of Occurrence** | Whenever requested by a user |

**10.) Start Game - Paul**

|  |  |
| --- | --- |
| **Use Case Name** | Start Game |
| **Scope** | Chad Game System |
| **Level** | User-goal |
| **Primary Actor** | Player |
| **Stakeholders** | Players: want to play the game |
| **Preconditions** | 1. Enough players have joined the game to play |
| **Postconditions** | 1. A player has won the game 2. Game history has been recorded |
| **Main Flow** | 1. Players take turns making moves until a player wins the game 2. Record game history in both players’ profiles 3. If the other player draw, system will wait for the response of the player. Unitil time up. |
| **Alternative Flows** | 1a. A player quits the game   1. Game ends 2. Record game as a loss for the quitting player, a win for the other player |
| **Special Requirements** | None |
| **Technology and Data Variations** | None |
| **Frequency of Occurrence** | As often as players wish to play |