**Use Case UC1: Registration to the system**

**Scope:** Chad game application

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User must be able to play a game with a unique username.
* The email and password should be unique. User will be able to resume games based on their log in credentials.

**Preconditions:** Account must be created and user must have a working email.

**Success Guarantee:**

* Once the account is created the user will not be removed from the database.

**Main Success scenario:**

* User can log in and out as much as he/she wants to.

**Extensions:** none.

**Special Requirements:**

* Login fields of the UI should be large enough to hold rather long usernames and emails.

**Technology and Data variations List:** none.

**Frequency of Occurrence:** Continuous. System will log in for the user, or user will login manually each time. Account creation happens once per user.

**Miscellaneous:** If user selects to allow the app to save data to their device the login process is automatic, otherwise they must do so manually.

**Use Case UC2: Registered users create and play games.**

**Scope:** Chad game application

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* A registered can start and play games with other players and be able to see their usernames in game.

**Preconditions:** Accounts must be created.

**Success Guarantee:**

* When a user starts a game, the game will start and the users will be able to play as normal.

**Main Success scenario:**

* User can have multiple games playing at the same time.

**Extensions:** none.

**Special Requirements:**

* Must have another player to start a game.

**Technology and Data variations List:** none.

**Frequency of Occurrence:** Continuous.

**Miscellaneous:** Users can have multiple games running, but must wait for another player to make his/her turn.

**Use Case UC3: Registered users invite other registered users to created game**

**Scope:** Chad game application

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User can invite another registered user or multiple users to start a created game.

**Preconditions:** The user who will be inviting others to the game will be the one who creates the game.

**Success Guarantee:**

* Once a game is started, a user can invite another registered user and that user will get a notification to start a game.

**Main Success scenario:**

* A user receives a notification, accepts it, and the game will start with the user that invited him/her.

**Extensions:** none.

**Special Requirements:**

* All users must be registered in the system and a game must be created.

**Technology and Data variations List:** none.

**Frequency of Occurrence:** Quite often.

**Miscellaneous:** Users can invite as many other players as they would like and start as many different games as they would like.

**User case UC4: Users accept and reject game invites**

**Scope:** Chad game application

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User’s who have received a notification to play a game have the ability to decline or accept it.

**Preconditions:** A game must be created and sent to another user.

**Success Guarantee:**

* If the game is accepted, the users will start another game. If the player rejects the notification, the user who sent the invitation will be notified and a game will not start.

**Main Success scenario:**

* User has the ability to accept or decline a game sent by random users.

**Extensions:** none.

**Special Requirements:**

* A game invite must have been created and sent by another user.

**Technology and Data variations List:** none.

**Frequency of Occurrence:** Quite often.

**Miscellaneous:** User’s can have multiple invitations at a time. An invitation will remain active until the user has selected a desired action for the game.

**Use Case UC5: Different Games at Same Time**

**Scope:** Chad game application.

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User must be part of different games at the same time.
* The user must be able to save and make progress.

**Preconditions:** user is identified and authenticated.

**Success Guarantee:**

* User must not be deleted from any games; he or she was part of.

**Main Success scenario:**

* The user logs into the system. Once the user is identified, the user can see the list of all the games he/she can play. The user can choose to start a game. The user can also choose to start another game at the same time.

**Extensions:** none.

**Special Requirements:**

* Touch screen UI large enough. Text must be visible. Recover data if system crashes.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** nearly continuous.

**Miscellaneous:** can the user open games on different platforms e.g. iPad, laptop or smart phone.

**Use Case UC6: Access to games**

**Scope:** Chad game application.

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User: must be able to access games he/she is part of. The user must be able to save and make progress.
* Company: easy for user to access several games at a time. The information about the game is stored quickly.

**Preconditions:** user is identified and authenticated.

**Success Guarantee:**

* The user logs into the system. Once the user is identified, the user can see the list of all the games he/she can play.
* The user may start any game.

**Main Success scenario:**

* The user logs into the system. Once the user is identified, the user can see the list of all the games he/she can play. The user can access any game in that list.

**Extensions:** none.

**Special Requirements:** none.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** nearly continuous.

**Miscellaneous:** can the user open games on different platforms e.g iPad, laptop or smart phone.

**Use Case UC7: Quit game**

**Scope:** Chad game application.

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User: must be able to quit games he/she is part of. The game must be deleted from the list of games he/she is part.
* Company: fast and easy process for the user.

**Preconditions:** user is identified and authenticated.

**Success Guarantee:**

* User can quit and shutdown a game. User is no longer playing the game.

**Main Success scenario:**

* The user logs into the system. Once the user is identified, the user can choose to start a game and after some time is able to successfully quit the game.

**Extensions:** none.

**Special Requirements:**

* Touch screen UI large enough. Text must be visible. Recover data if system crashes.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** nearly continuous.

**Miscellaneous:** can the user quit games on different platforms e.g. iPad, laptop or smart phone.

**Use Case UC8: Un-register from the system**

**Scope:** Chad game application.

**Level:** User “goal”.

**Primary Actor:** User

**Stakeholders and Interests:**

* User: must be able to quit from the system. The profile information must be safely deleted from the system
* Company: easy and fast to quit. It no longer has access to the personal information of the user.

**Preconditions:** user is identified and authenticated.

**Success Guarantee:**

* The user no longer has access to any features of the game.

**Main Success scenario:**

* The user logs into the system. Once the user is identified, the user can view his/her profile. The user deletes the profile.

**Extensions:** none.

**Special Requirements:** none

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** nearly continuous.

**Miscellaneous:** none.

**Use Case UC9: Record of games played**

**Scope:** Chad game application.

**Level:** User “goal”.

**Primary Actor:** User.

**Stakeholders and Interests:**

* System records games played by the user. Game must record who the user was, the start date and time of the game, end date of the game, and the end result/

**Preconditions:** user is identified and authenticated.

**Success Guarantee:**

* The user can look at his/her history of past games played.

**Main Success scenario:**

* User checks his his/her history of past games played and can see when the game started, when it finished, and who won/lost/tied.

**Extensions:** none.

**Special Requirements:**

* A menu that allows you to access “History”. Text must be visible. Recover data if system crashes.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** nearly continuous.

**Miscellaneous:** none.

**Use Case UC10: User Profile**

**Scope:** Chad game application.

**Level:** User goal.

**Primary Actor:** User.

**Stakeholders and Interests:**

* User has their own profile which has a nickname picked out by the user; must be unique.
* User profile must allow user to see their history of games.

**Preconditions:** user is registered and authenticated.

**Success Guarantee:**

* User can see their nickname displayed in game and in their profile.
* The user can look at his/her history of past games played.

**Main Success scenario:**

* User has the ability to display their nickname during game play and can visibly see it under their profile and check their history of past games.

**Extensions:** none.

**Special Requirements:**

* Nickname must be unique and no two users can have the same nickname. Text must be visible. Recover data if system crashes.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** Continuous.

**Miscellaneous:** none.

**Use Case UC11: players required (game play)**

**Scope:** Chad game application.

**Level:** User goal.

**Primary Actor:** Users.

**Stakeholders and Interests:**

* Game must have two users to start a game.

**Preconditions:** two users must be present to start a game, otherwise error.

**Success Guarantee:**

* One user invites another and opponent accepts.

**Main Success scenario:**

* User invites another user being his opponents to start game, once opponent accepts the game will start.

**Extensions:** none.

**Special Requirements:**

* Opponent must accept invitation to play otherwise the game will not start.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** Whenever someone invites another user to play a game.

**Miscellaneous:** none.

**Use Case UC12: Game start**

**Scope:** Chad game application.

**Level:** User goal.

**Primary Actor:** Users.

**Stakeholders and Interests:**

* Once an opponent accepts invitation to start a game another player cannot join pre-existing game.

**Preconditions:** Two users are in a game against each other.

**Success Guarantee:**

* User trying to join game getting a message saying he/she can’t join.

**Main Success scenario:**

* User trying to join another person’s game that is already started will get a message saying he/she can’t join and ask if they would like to invite user to start a game with them

**Extensions:** none.

**Special Requirements:**

* An error message on user’s screen that is attempting to join a game in motion.

**Technology and Data Variations List:** none.

**Frequency of Occurrence:** Only when user is trying to join a game that already exists and is being played by two users.

**Miscellaneous:** none.

**Use Case UC13: Determine which Player Starts First**

**Scope:** Chad game application

**Level:** User-goal.

**Primary Actor:** User

**Stakeholders and Interests:**

* System must determine who starts the game.
* User with White chess pieces must move first.

**Preconditions:** User authenticated and other user has accepted invite.

**Success Guarantee:**

* Game has started.

**Main Success Scenario:**

* Game is started, user with white chess pieces make the first move.

**Extensions:** N/A

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Once per game.

**Miscellaneous:** N/A

**Use Case UC14: Determine Player Turn:**

**Scope:** Chad game application

**Level:** User-goal

**Primary Actor:** User

**Stakeholders and Interests:**

* User will take turns moving one chess piece at a time.
* Turn ends once move has been made.

**Preconditions:** Both users must have started/accepted an invitation to game

**Success Guarantee:**

* The other appropriate user can now make their move.

**Main Success Scenario:**

* User makes move, system will allow other user to make move once current user has finished.

**Extensions:** N/A

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Continuous. After a user has finished their turn.

**Miscellaneous:** N/A

**Use Case UC15: Make Moves in Active Games**

**Scope:** Chat game application

**Level:** User-goal

**Primary Actor:** User

**Stakeholders and Interests:**

* User wants to be able to access only their active games.

**Preconditions:** User must have been invited to or started games.

**Success Guarantee:**

* User cannot access games that are not active

**Main Success Scenario:**

* User logs in, sees list of active games, given option to play active games.

**Extensions:** N/A

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Continuous

**Miscellaneous:** N/A

**Use Case UC16: Makes Move when Player’s Turn**

**Scope:** Chat game application

**Level:** User “goal”.

**Primary Actor:** User

**Stakeholders and Interests:**

* Users want to be able to make a move if and only if it’s their turn.

**Preconditions:** Game must have already been initiated.

**Success Guarantee:**

* Only the current user can make a move when it’s their turn.

**Main Success Scenario:**

* User starts game, white makes first move, opposing user’s turn to make move

**Extensions:** N/A

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Often

**Miscellaneous:** N/A

**Use Case UC17: Allow Valid Moves Only**

**Scope:** Game-play Mechanics

**Level:** Game-play Restriction

**Primary Actor:** User

**Stakeholders and Interests:**

* Game Engine: wants to ensure the game is being played according to the rules
* Opponent: wants to know that only valid moves are used against them.

**Preconditions:** a game has started

**Success Guarantee:**

* Invalid moves if tried are rejected and the games state does not change

**Main Success scenario:**

* User attempts to preform an invalid move
* System identifies the move as incorrect
* The games state is returned to whatever state it was in before the move was made
* The user is notified the move was invalid

**Extensions:** none.

**Special Requirements:**

* If the game rules are changed newly valid moves must not trigger the invalid move behavior

**Technology and Data Variations List:**

* Any move that is not allowed by the rules must be identified
* Any move that is allowed by the rules must not be treated as invalid

**Frequency of Occurrence:** occasional

**Miscellaneous:** N/A

**U****se Case UC18: Save the State of Active Games**

**Scope:** User Games

**Level:** Games In Progress

**Primary Actor:** User

**Stakeholders and Interests:**

* User: wants to resume an unfinished game after leaving the game for some amount of time

**Preconditions:** a game has been started but not finished and one of the players leaves the game

**Success Guarantee:**

* The user may resume playing the game as normal

**Main Success scenario:**

* The user leaves an unfinished game
* The user rejoins the game and may play there next move as if they had not left

**Extensions:**

* The opponent may make their next move if the user had finished their turn before leaving the game
* The system will update the game's state and when the user returns the game will show the opponent's move

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Occasional

**Miscellaneous:** N/A

**U****se Case UC19: Determine End of Game and Game Result**

**Scope:** Game-play Mechanics

**Level:** End of Game

**Primary Actor:** User

**Stakeholders and Interests:**

* User: wants to be notified when the game is finished and what was the outcome.
* User Statistics Recorder: wants to know the outcome of the game so it may update the number and outcomes of games the user has played

**Preconditions:** The game must be over according to the rules

**Success Guarantee:**

* The user is notified of the outcome of the game

**Main Success scenario:**

* A user makes a move that ends the game
* Both players are notified if they won, lost, tied or if the game ended in a draw according to the rules

**Extensions:**

* A player may forfeit the game without a game ending move occurring
* The game will end
* The player that forfeited will be notified they lost and the opponent will be notified they won

**Special Requirements:** N/A

**Technology and Data Variations List:** N/A

**Frequency of Occurrence:** Often

**Miscellaneous:** N/A