ASSIGNMENT 1

REQUIREMENTS ENGINEERING

CSE2115 Software Engineering Methods
Team 90

Made by:

Victor Roest
Tim Numan
Maikel Houbaer
Jesse Nieland
Eduard Filip

CONTENTS

Exercise 1 - Requirement Engineering

1

FUNCTIONAL REQUIREMENTS

For the game Pool, the requirements regarding functionality and service are grouped under the Functional Requirements. Within these functional requirements, four categories can be identified using the MoSCoW model[1] for prioritizing requirements.

1.1 MUST HAVES

- The game shall start with a login window
- The game shall allow the user to log in with its current account (username and password)
- If the user tries to login with an existing username, the game shall get the users credentials from the database, validate the entered login data, and depending on the outcome either log the user in or display an error message stating the password was not correct
- If the user tries to login with an unknown username, the game shall display an error message informing the player that the account does not exist
- When the game needs data from the database
    - the client shall send a request to the server
    - the server shall send a request to the database, following the specifications stated in the non-functional requirements
    - the database shall send back the requested data to the server
    - the server shall send the data to the client
    - the client should use this data to perform the needed action
- After login/the end of a game, the game shall show the button: “Play a game”
- When the player selects “Play a game”, the game should start a new pool match with another local player

[1] http://en.wikipedia.org/wiki/MoSCoW_method

- The game shall start a new pool match by showing a pool table with 15 balls arranged in a triangle shape on the right side, and the white ball anywhere behind the service line (default = center) before the actual pool match starts
- The game shall randomly arrange the 14 balls, the black one always being in the middle
- The game shall randomly assign one of the 2 players the first turn
- For each turn, the game shall let the player that has the current turn shoot the white ball once
- The game shall allow the player to be able to rotate the cue around the white ball with the mouse, only if it is his turn
- The game shall allow the player to choose the strength of the ball hit by dragging the cue with the mouse, only if it is his turn
- The game shall move the white ball in the direction given by the cue, the position where the cue hit the ball (default = middle) and the speed given by strength inputted by the user
- The game shall move the other balls if they collide with the white ball
- The game shall not allow any ball to go outside the table
- The game shall score a ball if it goes into one of the 6 pockets
- The game shall assign the first player who pots a ball the category of the just potted ball (stripes or solids)
- The game shall assign the other player the opposite category of the ball (stripes or solids)
- The game shall give the player an extra turn in the following situations:
    - Potting the first ball of any category without potting the white ball
    - Potting a ball of his category without potting the white ball
- The game shall give the other player a turn in the following situations:
    - Current player not potting any of the balls of his category
    - Current player getting a foul
- The game shall give the player a foul in the following situations:
    - Failing to hit a ball
    - Hitting one of the other balls before hitting one of his own category
    - Potting the white ball
- When a player commits a foul, next to giving the other player a turn, the game shall allow the other player to place the white ball anywhere on the table
- The game shall end the current pool match when a player pots the black ball
- The player shall win the match in the following situations:
    - The player pots the black ball after potting all the other balls of his category
    - The other player pots the black ball before potting all the other balls of his category
- The player shall lose the match in the following situations:
    - The other player pots the black ball after potting all the other balls of his player's category
    - The player pots the black ball before potting all the other balls of his category

- The game shall show a message at the end of the game displaying whether the player has won or lost the match
- At the end of each play, the game shall ask the user whether he wants his score to be stored in the database and if so, store the recorded score together with the user name
- At the end of each play, the game shall show the top 5 scores that have ever been recorded

1.2 SHOULD HAVES

- The game shall prompt for sign-up if an unknown username is entered
- If the user wants to sign up, the game shall ask the player to enter a username (unique) and a password and store these in the database
- After login/the end of a game, the game shall show a menu with 3 buttons: “Play a game”, “Leaderboard” and “Sound”
- When the player selects “Leaderboard”, the game shall show the top 5 scores that have ever been recorded
- The “Sound” button shall allow the user to turn on or off the sound
- The game shall have specific sound effects in the following situations:
    - Collision sound when two balls collide
    - When the cue hits the ball
    - When a ball hits a wall
    - When a ball goes into a pocket
    - When the player wins the game
    - When the player loses the game
- The game shall show a power meter which shows the strength of a potential hit, when the player is dragging the cue
- The game shall allow the player to place the white ball anywhere behind the service line before the actual pool match starts
- The game shall make the user choose one pocket in which he wants to pot the black ball after he potted all the balls of his category (stripes or solids)
- The game shall allow the user to choose the hitpoint of the cue and the white ball
- The game shall allow a limited amount of time for each turn
- If the player runs out of time for his turn, the game shall give a foul and handle this as specified before
- The game shall display the following things during a game:
    - Which team each player is
    - The balls each player scored/has left
    - Time left for this turn
    - Whether it is Your/Opponent’s turn
- The game shall have an in-game menu that will allow the player to do the following:
    - Toggle the sound (on/off)
    - Exit the game

1.3 COULD HAVES

- When rotating the cue around the white ball, the game shall display the angle and a power ratio of the hit and white ball
- When the user enters his first ever game, the game shall make a small tutorial in order to help the player understand how the game works
- For the registration, instead of just one password field, the game shall let the user enter the same password in another field for confirmation
- At the end of the game shall allow the player to have a rematch with the current opponent
- The game shall play music
- After login/the end of a game, the game shall show a menu with 4 buttons: “Play a game”, “Leaderboard”, “Settings” and “Logout”
- The “Settings” menu shall allow the player to do the following:
    - Toggle the sound (on/off)
    - Toggle the music (on/off)
    - Toggle the screen between “windowed” and “full screen”
- The game shall have at the login screen a “Remember me” checkbox which enables the game to store the credentials of the current user and automatically log in next time the application is opened
- Given the “Remember me” checkbox was checked last time, the game shall automatically log in if credentials have been stored
- Choosing “Logout” in the game menu will delete the locally stored user credentials, if those were even stored

1.4 WOULD/WON’T HAVES

- The game shall assign a player a level based on the number of wins
- The game shall offer extra game modes (and for each of these display a button in the main menu, visible after login/the end of a game):
    - Playing with a local player
    - Playing with an AI player
    - Playing a “Practice” mode: a single-player version of the game with only 7 balls (1 team + black ball), which will end after all balls are potted. When the player commits a foul, the game shall revert this move.
    - Playing a game with a different number of balls

2

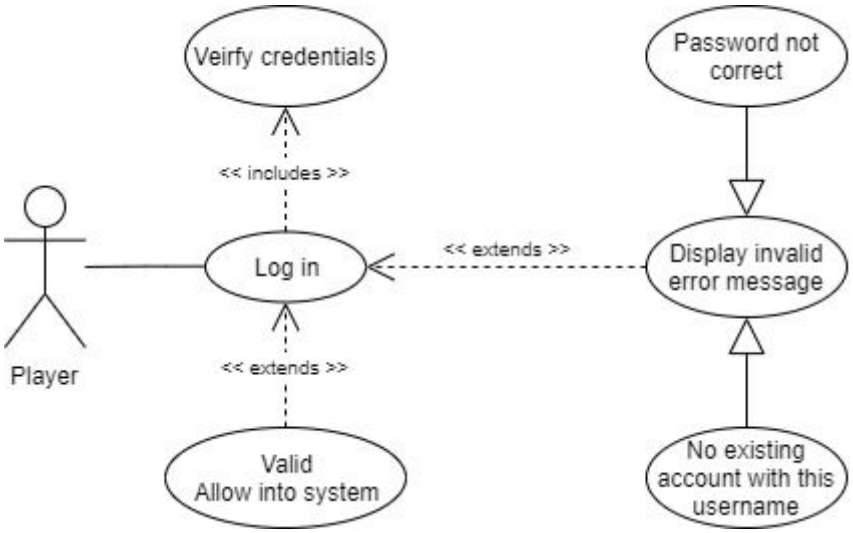
NON-FUNCTIONAL REQUIREMENTS

Besides the provided functionality and services, design constraints need to be included in the requirements specifications as well. These requirements do not indicate what the system should do, but instead indicate the constraints that apply to the system of the development process of the system.

- The game shall be playable on Windows (7 or higher), Mac OS X (10.8 or higher), and Linux
- The game shall be implemented in Java
- A first full working version of the game shall be delivered on January 24, 2020, before 18:45
- For the iterations after the first week, the Scrum methodology shall be applied
- Each scrum iteration will take 2 weeks, not considering holiday and exam weeks
- The implementation of the game shall have at least 75% of meaningful branch test coverage for non-GUI classes (where meaningful means that the tests actually test the functionalities of the game and for example do not just execute the methods involved)
- For connection to the database a SQL and JDBC driver shall be used
- For querying the database, prepared statements in Java shall be used to avoid code-injection
- The passwords shall be encrypted in the database
- The user shall be able to start a pool match within 5 clicks
- The game shall not be more than 500MB in size
- The game shall not use more than 512MB RAM

Exercise 2 - Modelling Use cases

1. User Authentication



Use case: User authentication
Author: Tim Numan
Date: 26/11/2019
Purpose: User authentication before playing Pool
Overview: The user starts the game. The log-in screen will be shown, which will ask the user to type in the username and password in order to log in. If the provided username does not exist, an error will be shown informing the user of this. Else, if the validation of the given combination is successful, the game will move on to the main menu. Otherwise, an error will be shown saying that either the username or password are invalid
Cross-reference: Requirements 1-4
Actors: User (primary actor) and the authentication server (secondary actor)
Pre conditions:

- The user is connected to the internet
- The user already has an account

Post conditions:

- The game displays the main menu with the button “Play a game” or displays an error message to the user

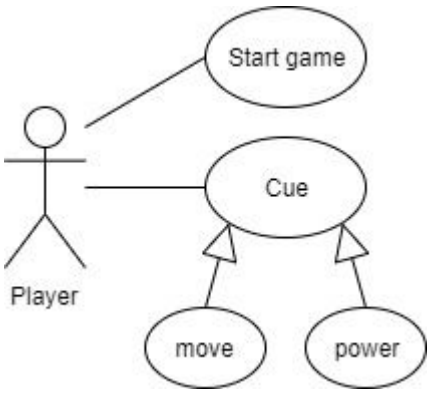
Flow of events:

User actions	System actions	Server actions
1. User starts the game	2. System displays the log-in screen	
3. User inserts username and password	4. System connects to the server	5. Server validates credentials
	6. System acknowledges and displays the main menu	

Alternative flow of events:

- Step 6: The user credentials are invalid. An error message will be displayed (“The username or password are invalid”) and the user is asked to authenticate again

2. Game interaction



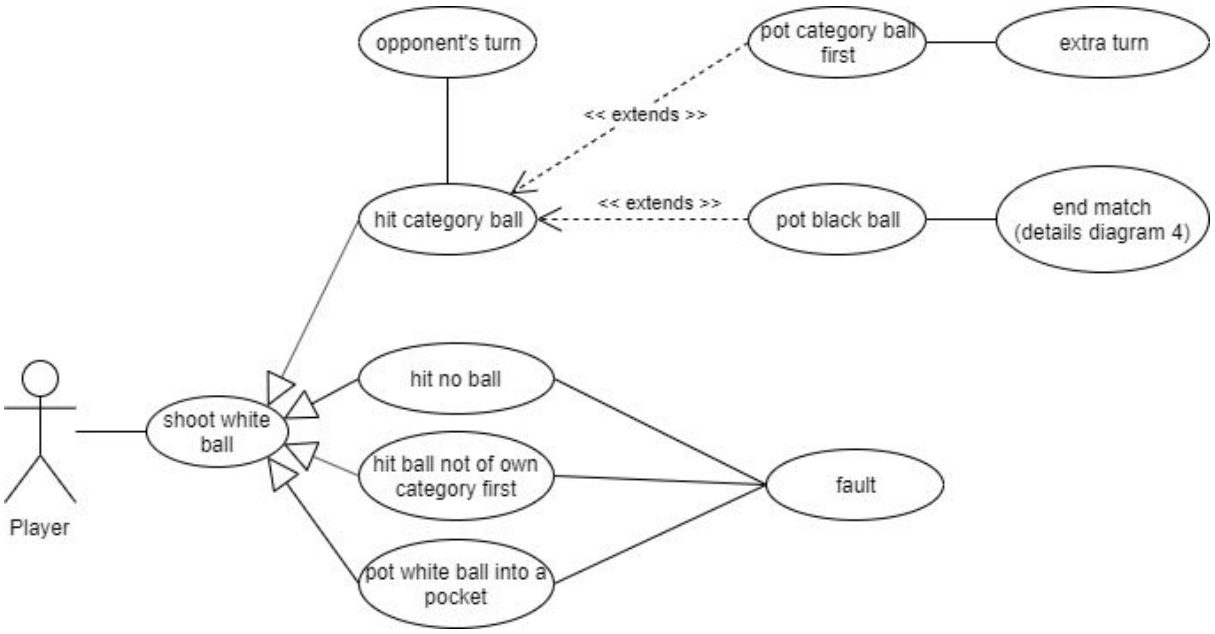
Use case: Game interaction
Author: Eduard Filip
Date: 26/11/2019
Purpose: User starts a new pool match and plays it by interacting with the cue
Overview: The user starts a new pool match by pressing the button “Play a game”. The game will show the table with the white ball on the left side and the 15 balls in the shape of a triangle to the left side. The user plays the game by rotating the cue around the white ball, then dragging the cue in order to set the power of the hit. The game will show the current power with the power meter on the left side of the screen. After that, the white ball will start moving in the direction pointed by the cue with the specified speed.
Cross-reference: Requirements 6-14
Actors: User (primary actor)
Pre conditions:
For interaction with the cue

- The game is in a valid state: the board contains the white ball, black ball and all not-scored other balls
- It is the turn of this player

Flow of events:

User actions	System actions
1. User presses the button “Play a game”	2. Game launches and creates initial game state
3. User moves the mouse to move the direction of the cue, then drags the cue to set the speed	4. white ball is moved in the direction of the cue and the given speed

3. White ball interaction with the table



Use case: White ball interaction with the other elements on the table
Author: Eduard Filip
Date: 26/11/2019
Purpose: White ball moves on the table
Overview: The white ball is hit and the games makes a certain choice depending on whether a ball is hit, what ball is hit first and if a ball is potted into a pocket.
Cross-reference: Requirements 15-17, 20-24
Actors: User (primary actor)
Pre conditions:

- The cue has a valid direction and a valid speed set in order to make the white ball move

Post conditions:

- Depending on the situation, either the same player or the opponent will have a valid turn, or the match will be ended

Flow of events:

User actions	System actions
1. User hits the white ball with the cue	2. The game moves the white ball in the direction and speed given by the cue
	3. White ball hits a ball of the player's category
	4. One of player's category balls is potted into a pocket
	5. The player has an extra turn

Alternative flow of events:

Alternative 1:

- Step 4: One of opponent's category balls is potted into a pocket
- Step 5: The other player is given a turn

Alternative 2:

- Step 3: White ball hits a ball of opponent's category
- Step 4: ~doesn't matter~
- Step 5: Foul. The other player is given a turn

Alternative 3:

- Step 3: White ball doesn't hit any ball
- Step 4: ~doesn't exist~
- Step 5: Foul. The other player is given a turn

Alternative 4:

- Step 3: White ball is potted into a pocket
- Step 4: ~doesn't exist~
- Step 5: Foul. The other player is given a turn

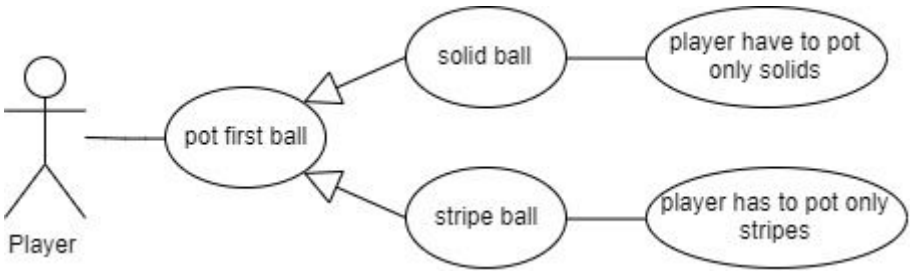
Alternative 5:

- Step 4: None of the player's category balls is potted
- Step 5: The other player is given a turn

Alternative 6:

- Step 4: The balck ball is potted into a pocket
- Step 5: The game ends the current pool match

4. Potting the first ball



Use case: Potting the first ball
Author: Victor Roest
Date: 26/11/2019
Purpose: Player should be assigned a category once the first ball is potted into a pocket
Overview: The player hits the white ball and a first ball is potted into a pocket. The player which potted the first ball should be assigned the ball's category (solids/stripes). The opponent is assigned the other category. From that moment on, each player can only hit its category balls and has to pot those in order to come to the final part of the match: potting the black ball
Cross-reference: Requirements 18, 19, 20.1
Actors: User (primary actor)
Pre conditions:

- The cue has a valid direction and a valid speed set in order to make the white ball move
- No balls have been previously potted

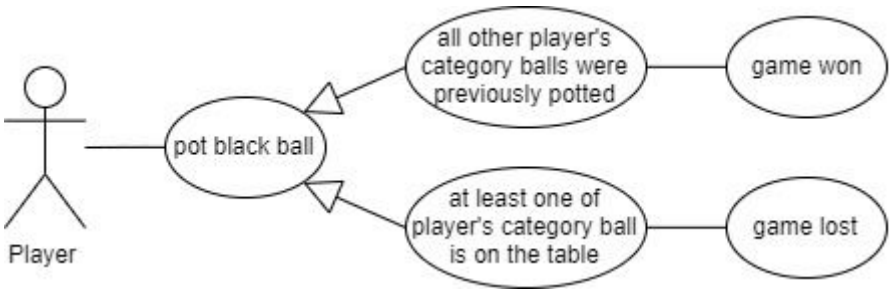
Post conditions:

- The teams are assigned and the player who potted the first ball has an extra turn

Flow of events:

User actions	System actions
1. Player hits the white ball	2. The game moves the white ball in the direction and speed given by the cue
	3. A first ball is potted into a pocket
	4. The game assigns the current player the category of the potted ball
	5. The game assigns the opponent the other category
	6. The current player has an extra turn

5. Potting the black ball



Use case: Potting the black ball
Author: Maikel Houbaer
Date: 26/11/2019
Purpose: The game ends the current match and makes a verdict depending on the current situation
Overview: The player pots the black ball into a pocket. The game ends the game and check whether the current player was allowed to pot the black ball. If he potted all the balls of his category before, then he wins the match. Otherwise, he loses and the opponent wins
Cross-reference: Requirements 24-27
Actors: User (primary actor)
Pre conditions:

- The cue has a valid direction and a valid speed set in order to make the white ball move

Post conditions:

- The game displays the message “You win” or “You lost”

Flow of events:

User actions	System actions
1. Player hits the white ball	1. The black ball is potted
	2. Game ands the current match
	3. Player potted all the balls of his category before this turn
	4. Player wins the match

Alternative flow of events:

- Step 3: Player still has remaining balls of his category on the table
- Step 4: The player loses the match

6. Processes after the end of the pool match

Use case: User authentication
Author: Jesse Nieland
Date: 26/11/2019
Purpose: Show what happens after the pool match ended and what the player can do with his score
Overview: The pool match is over. The game asks the player whether he wants to store his score in the database. Depending on his choice the game sends or not a command to the server to store the current score of the player. After that, the game displays the top 5 players, alongside with their high scores
Cross-reference: Requirements 27-29
Actors: User (primary actor) and the server (secondary actor)
Pre conditions:

- The pool match is finished
- The user already has an account

Flow of events:

User actions	System actions	Server actions
	1. Pool match is ended	
3. User accepts to store his score	2. Ask the user whether to store the score or not	
	4. User sends to the server a post command to store the player's score	5. Server stores the username of the player and the score in the database
	5. Game requests for top 5 scores	6. Server retrieves from the database top 5 scores with the corresponding usernames
	7. Show on the screen the top 5 players with their scores	

Alternative flow of events:

- Step 3: User refuses to store his score
- Steps 4,5: ~do not apply~