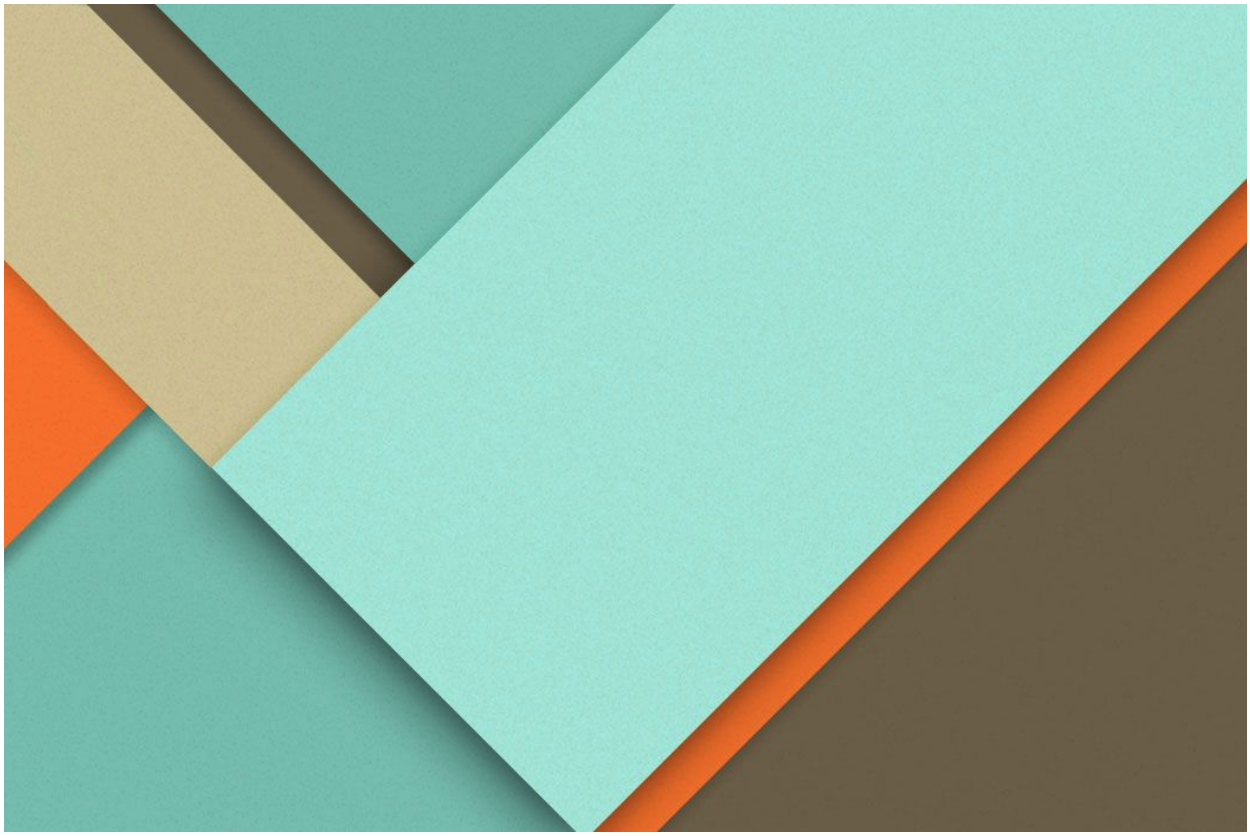

Air Hockey (group 42)

Ionut Constantinescu, Robert Minea, Jaron Rosenberg, Darwin Morris
Delft University of Technology
Software Engineering Methods

Non-functional requirements:

- The Game will be implemented in Java
- The game shall be playable at least on Windows 10
- A working version of the game shall be delivered before the last presentation for the Software Engineering Methods course.
- The Scrum methodology shall be applied throughout the project development
- The implementation of the game shall have at least 75% of meaningful line test coverage (where meaningful means that the tests actually test the functionalities of the game and for example do not just execute the methods involved)
- LibGDX will be used as the main game engine library (for ease of developments)
- Gradle will be used (with Jacoco, PMD, Checkstyle and SpotBugs)
- The game will have a SQL database and will connect to this database using JDBC
- Prepared statements shall be used for querying the database in order to avoid code-injection vulnerabilities

Functional requirements:

Must have:

- The player shall be able to create an account and be able to authenticate through login and password
- When the Game Starts the the two Pusher and the Puck will be displayed on the screen with the Puck being on one of the sides
- The game will be 2D
- At the beginning of the game the Air Hockey board with 2 gates is displayed
- When the game starts a scoreboard should be displayed on the screen
- When the game starts the 2 Players should be able to move their pushers using predefined buttons from the keyboard (WASD and IJKL)
- The player should be able to move the pusher in any direction
- The game should not allow the Player to move his pusher to the opponent's side of the screen
- The Game shall not allow for the Puck and the Pushers to stay out of bounds
- When the Puck hits the Pusher, it is going to be reflected
- When the Puck hits a wall it will ricochet
- When the puck enters a players goal the opposing players score is incremented and the puck is returned to the side of the player who didn't get a point

- After the Game ends, information about the game will be stored in the Database
- At the end of each play, the game should show the top 5 scores ever made in the game by other players
- The game shall have a leaderboard with the players with the biggest score.

Should have:

- The game shall have a theme song playing throughout the game
- The game shall have sound effects
- The game will include basic statistics about the performance of a specific Player
- The game shall have a Main Menu
- The Main Menu shall allow the Player to start a new Game
- The Main Menu shall allow the player to select some basic Game Settings (Like Sounds and AI difficulty if implemented)
- The game shall have a Single player mode
- In the Single player mode the player shall play against an AI bot

Could have:

- The game shall have a 2 vs 2 players mode
- The game shall have fancy Particles/Animations
- The Player shall have access to special abilities
- The Player shall be able to change the theme of the game
- The game shall have a Galactic Theme

Won't have:

- The game shall have a Multiplayer Online functionality
- The game shall be completely 3D
- The game shall have microtransactions
- The game shall be played from the Command Line
- The game shall have ads
- The game shall have an unbeatable AI bot