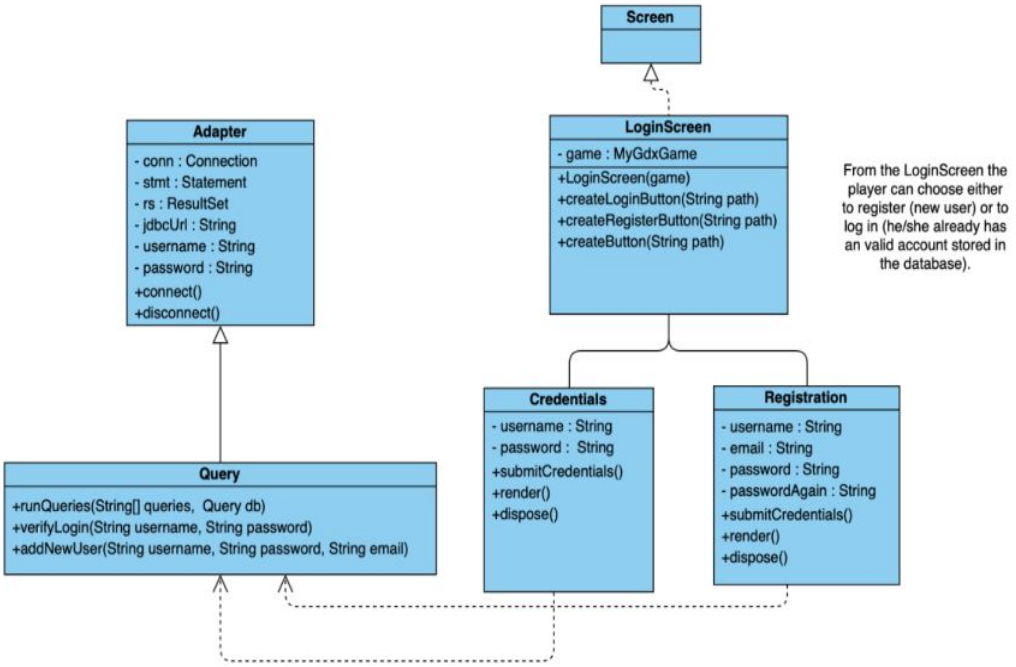
Assignment 2 Air Hockey

Game created by:
Matei Anton, Ben Provan-Bessel, Shaan Hossain,
Ioana Savu, Krzysztof Garbowicz

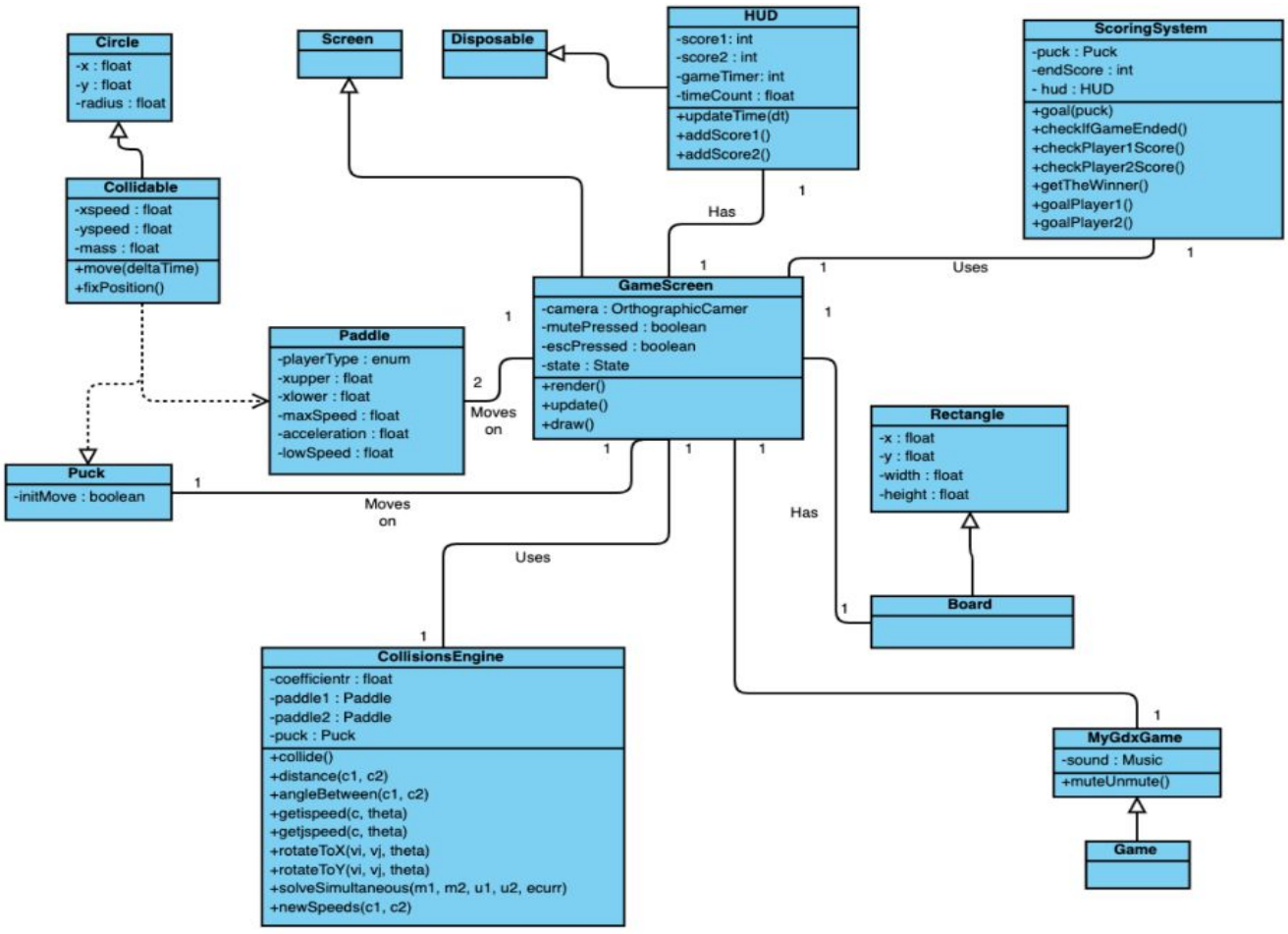
Exercise 1 (Class Diagrams):

Authentication



After opening the game, the player can see the LoginScreen which implements the interface Screen. There are 2 relevant methods: one which redirects the user to the Credentials Screen and one which redirects the user to the Registration Screen. This methods are createLoginButton() and createRegisterButton(). There is also an extra helper method which creates an simple button, to avoid duplicate code, called createButton(). The player would choose to go to Credentials if he/she already has a valid account stored in the database. Otherwise, the player would choose Registration in order to play the game.
In either case, after entering the credentials, they are checked by 2 methods present in Query class: verifyLogin would check if the entered credentials are in the database, addNewUser would check if the new account can be created. If any of them fails, the user will be notified by a pop-up window which details the error (account already exists or wrong credentials).
The class Query extends class Adapter, where the connection to the database is created.
Please note that LoginScreen contains some more methods which are not part of the diagram, since they deal with the GUI.

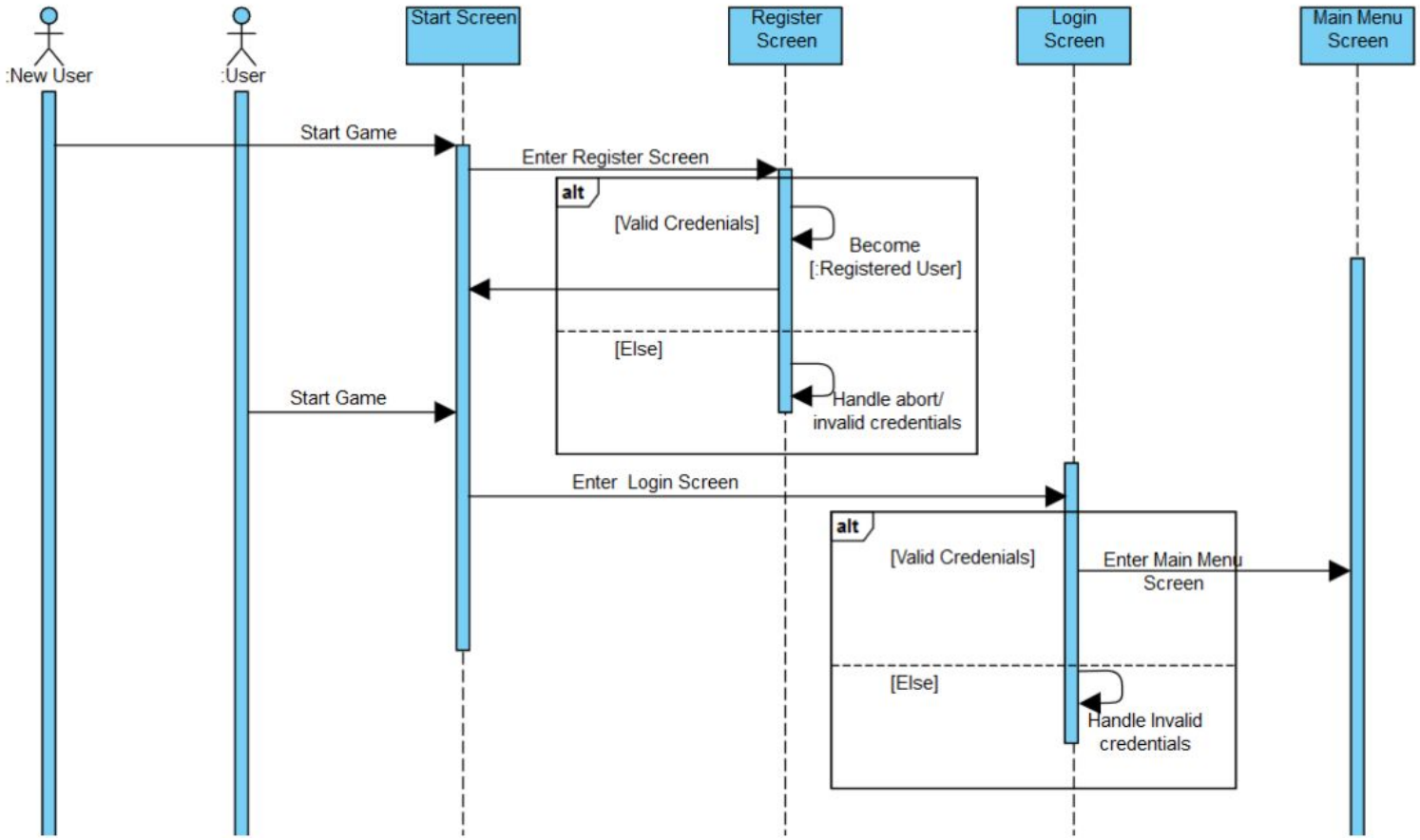
Main Game Loop



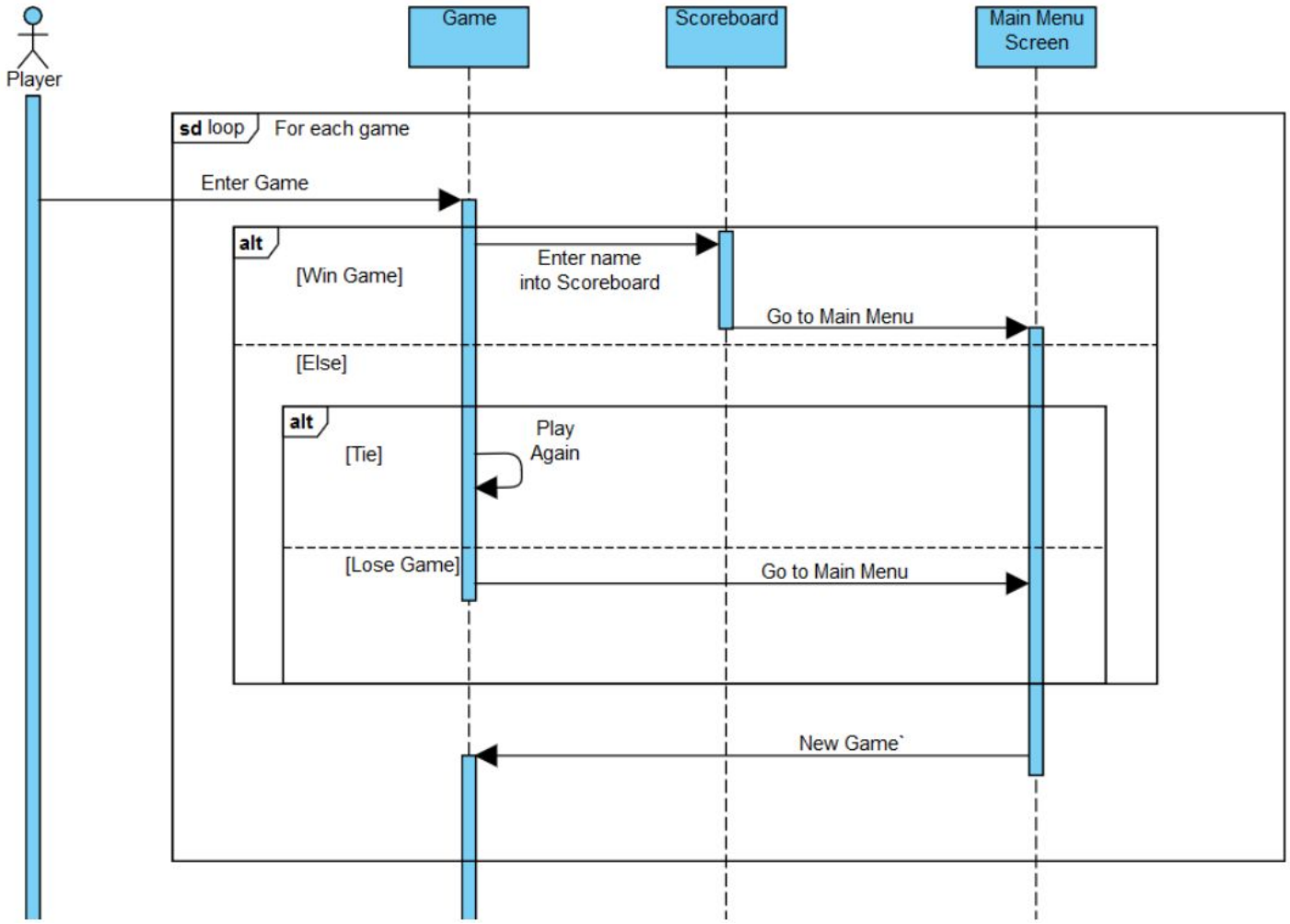
The game screen is the where the main objects of the game are used, and it is where the game is actually player and where the main game loop takes place (with the update, and render and draw methods). The Puck and Paddle are the main objects of the game. They extend the class Collidable, which in turn extends the Circle class (from libgdx). The users can control a respective Paddle, which can interact with the puck(through the move and setSpeeds methods). The CollisionsEngine is the core class the deals with the collisions between the puck and the paddles. It calculates the resulting velocities of the puck and paddles after a collision based on real life mathematical theory (through collide() and its helper methods). The board is used to hold the texture of the background. Scoring system is what keeps track of the score and detects when the puck lands in the goal. It also checks the winning and ending conditions of the game. The MyGdxGame class contains the spriteBatch and music and mute function. This extends the main Game class (libgdx).

Exercise 2 (Sequence Diagrams):

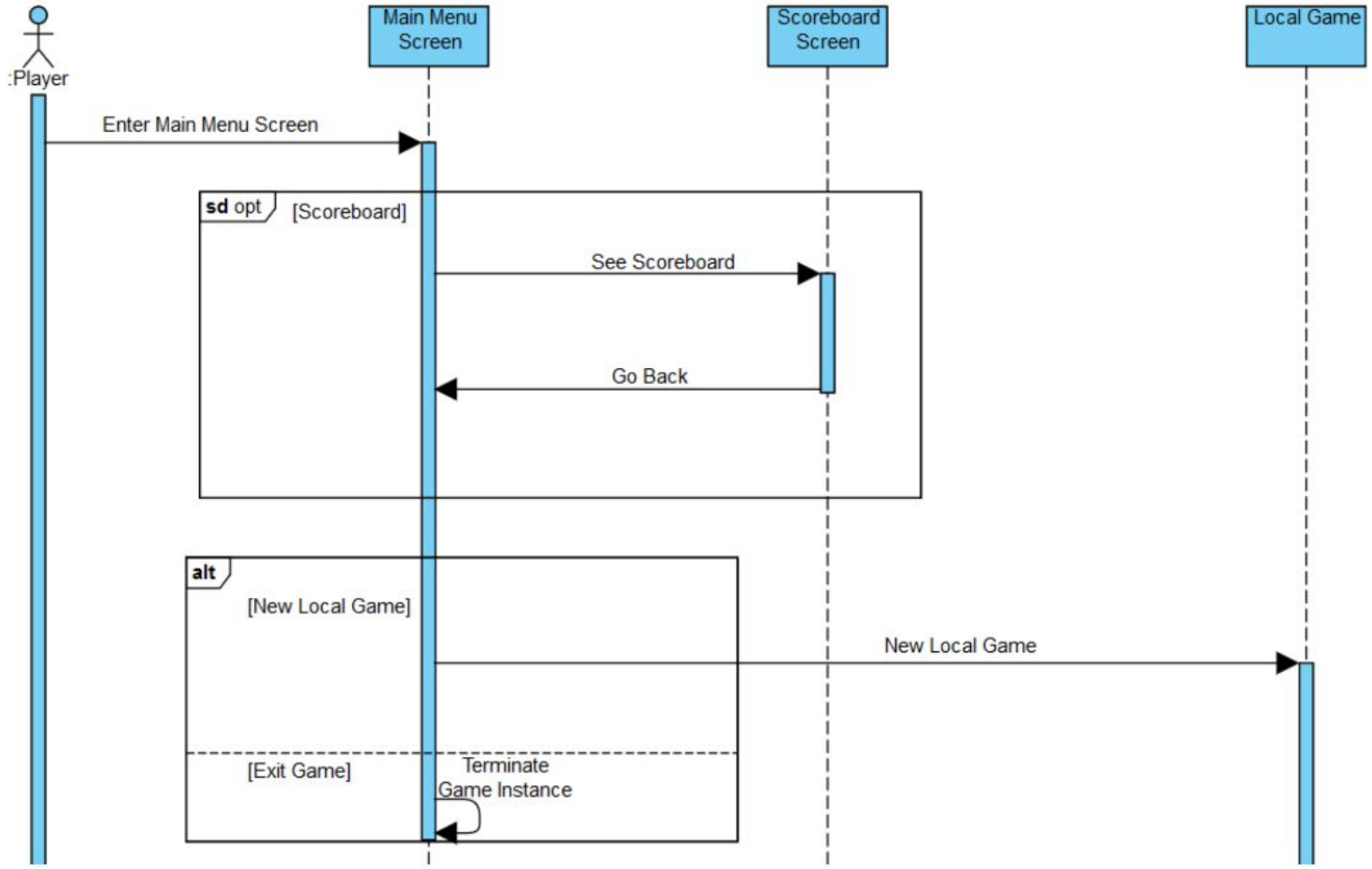
Credentials



Finishing Game



Main Menu


:Player
Main Menu Screen
Scoreboard Screen
Local Game
Enter Main Menu Screen
sd opt
[Scoreboard]
See Scoreboard
Go Back
alt
[New Local Game]
New Local Game
[Exit Game]
Terminate Game Instance