# Software Design Description

**Version 1.0**

## Quiz Game

**CS204301A-Team17**

**Alyssa Roche, Teamleader**

**Collett, Jared Mark**

**Lu, Yang**

1. Introduction

**1.1 Purpose**

The Quiz Game application provides a quiz game where a user could answer questions and receive a score based on their result, or create their own quiz. The intent of this document is to provide detailed information on the design of the Quiz Game application as a whole, as well as detailed information on the behaviors of its various functions.

**1.2 Scope**

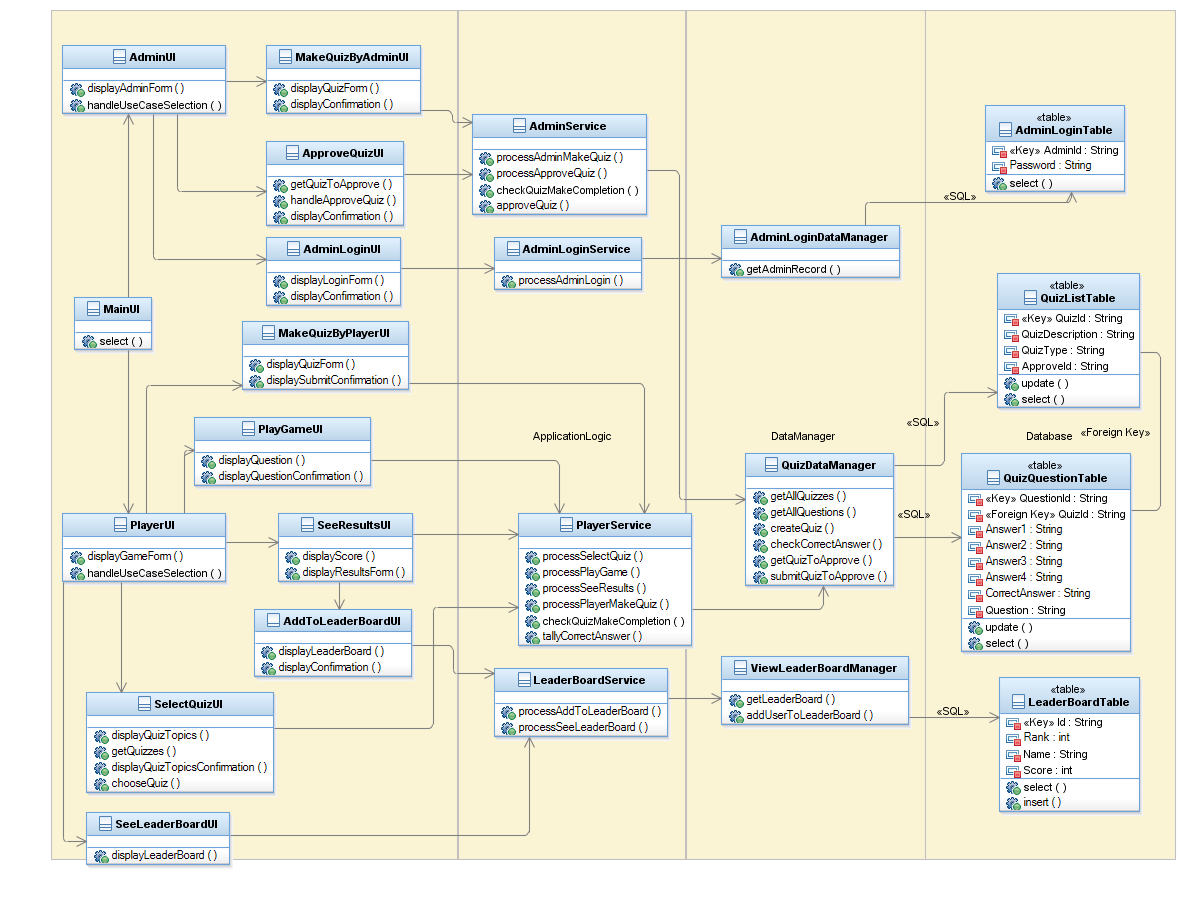
The Game Quiz application is intended to create a fun pastime for the users where they can play quizzes, as well as create their own. This application has two types of users: administrators and players. Players can test their knowledge with quizzes that other users have made, and attempt to have a high enough score to appear on the leaderboard. The user may also create their own quizzes, but if the user is not logged in as an administrator, their quiz must be approved by an administrator before it can be played.

**1.3 Context**

Section 2.1 uses a layered design class diagram to describe the Quiz Game system. The four layers are User Interface at the top, followed by Application Logic, Data Management and Database at the bottom. Section 2.2 uses sequence diagrams to describe in detail the function of each use case.

2. Body

**2.1 Design Module Viewpoint**

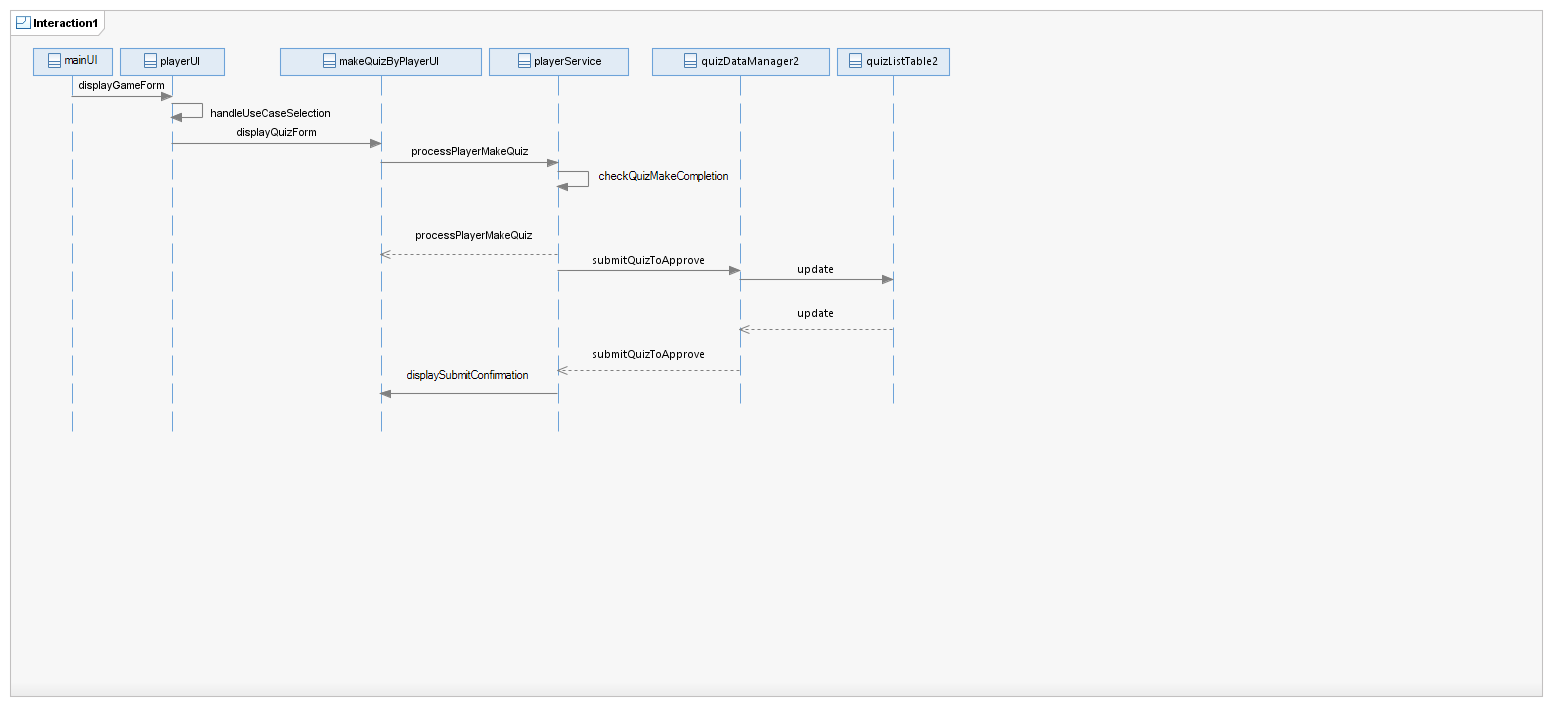
****

**2.2 Design Behavior Viewpoint**

The main user interface will be displayed and the user will have the option to proceed as either a player or administrator. If the user chooses the administrator option, a login user interface will be displayed and when the appropriate information is entered, they will be able to proceed to either make a quiz, or approve quizzes that players had made. If the user chooses to be a player, they will be presented with the options to view the leaderboard, to make a quiz, or to play a game. If they choose to play a game, they must then select the quiz they wish to play. After the player has completed a quiz, they must view their results, and are presented the option to add their quiz score to the leader board. Functions of the leaderboard and handled in the LeaderBoardService class. Functions related to game play are handled in the PlayerService class. Functions of the administrator are handled by the AdminService class. Functions of the administrator login are handled by the AdminLoginService class. The PlayerService and AdminService classes access the QuizDataManager class to retrieve information from the QuizListTable and the QuizQuestionTable in order to keep track of quizzes, quiz questions, and quiz answers.

**2.2.1 MakeQuizByPlayer Realization**

**2.2.1.1 MakeQuizByPlayer Realization Sequence Diagram**

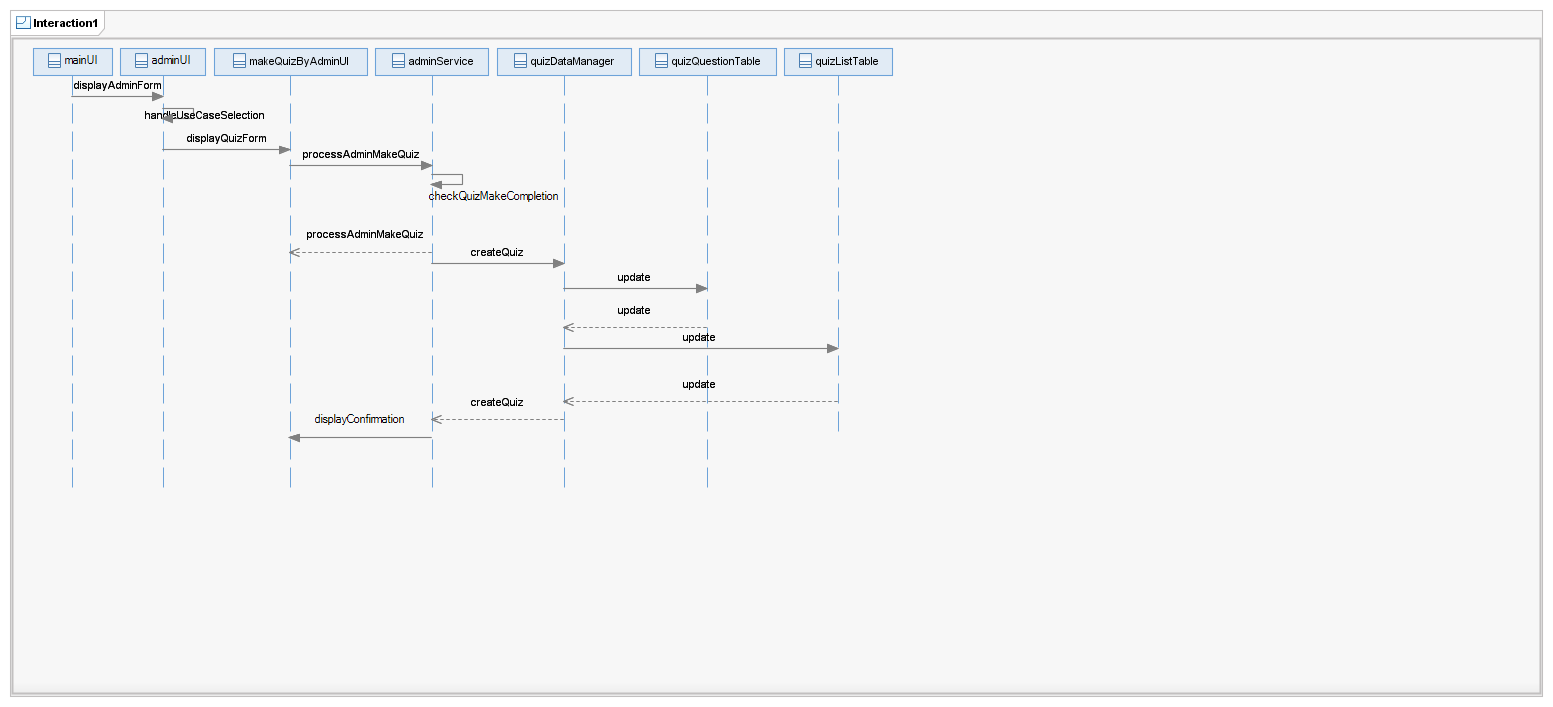
****

**2.2.1.2 MakeQuizByPlayer Realization Description**

From the main user interface, the user chooses to proceed through the game as a player, and then selects the option to make a quiz. The makeQuizByPlayer user interface is displayed, where the user will then fill in the information needed to make the quiz. Upon completing the form, the user will press a submit button, where a check will be done to ensure that the form has been filled out completely and the player is satisfied with the information they have submitted. Once the user has confirmed this, the quiz will be submitted and the quizListTable will be updated with this new quiz, with the attribute ApproveId, indicating that it has yet to be approved by an administrator. The player then receives confirmation that there quiz has been submitted for approval.

**2.2.2 MakeQuizByAdmin Realization**

**2.2.2.1 MakeQuizByAdmin Realization Sequence Diagram**

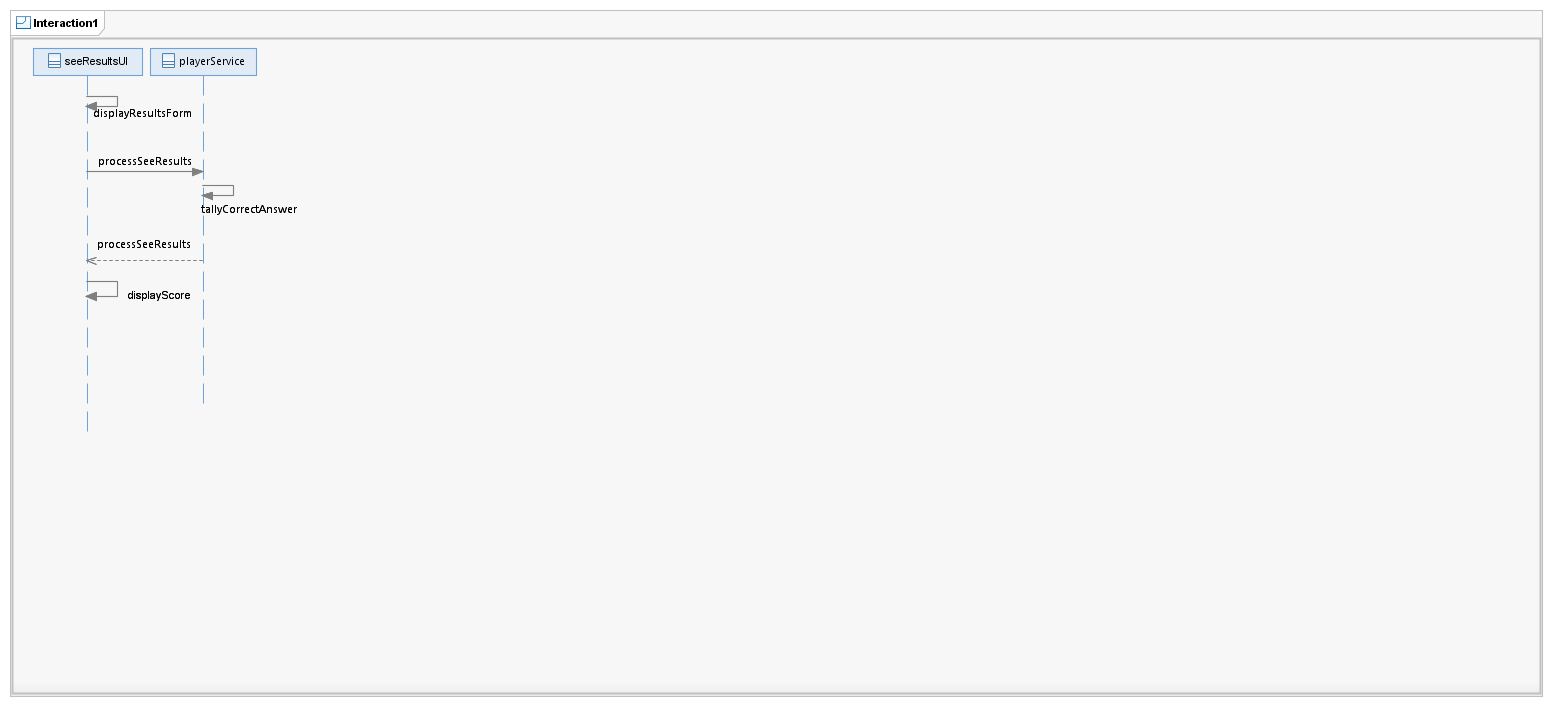


**2.2.2.1 MakeQuizByAdmin Realization Description**

From the main user interface, the user chooses to proceed through the game as an administrator, and then selects the option to make a quiz. The makeQuizByAdmin user interface is displayed, where the user will then fill in the information needed to make the quiz. Upon completing the form, the user will press a submit button, where a check will be done to ensure that the form has been filled out completely and the user is satisfied with the information they have submitted. Once the user has confirmed this, the quiz will be submitted and the quizListTable will be updated with this new quiz, and the quizQuestionTable will be updated with the new questions for this quiz. Confirmation of the creation of the quiz will then be displayed to the user.

**2.2.3 SeeResults Realization**

**2.2.3.1 SeeResults Realization Sequence Diagram**

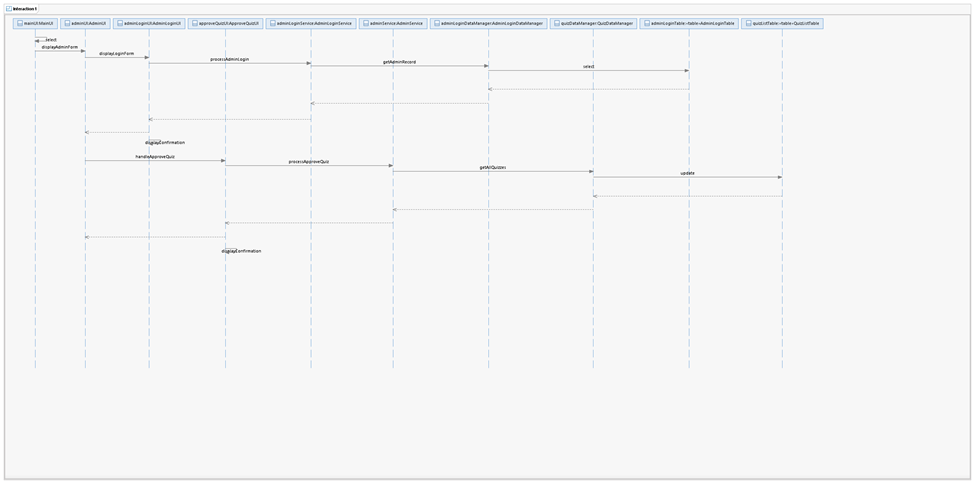
****

**2.2.3.2 SeeResults Realization Description**

Once the player has completed the quiz, the seeResults user interface will be displayed. The method processSeeResults retrieves the final number of correct answers from tallyCorrectAnswers and processSeeResults calculates the users score. The score is then displayed for the user.

**2.2.4 ApproveQuiz Realization**

**2.2.4.1 ApproveQuiz Realization Sequence Diagram**

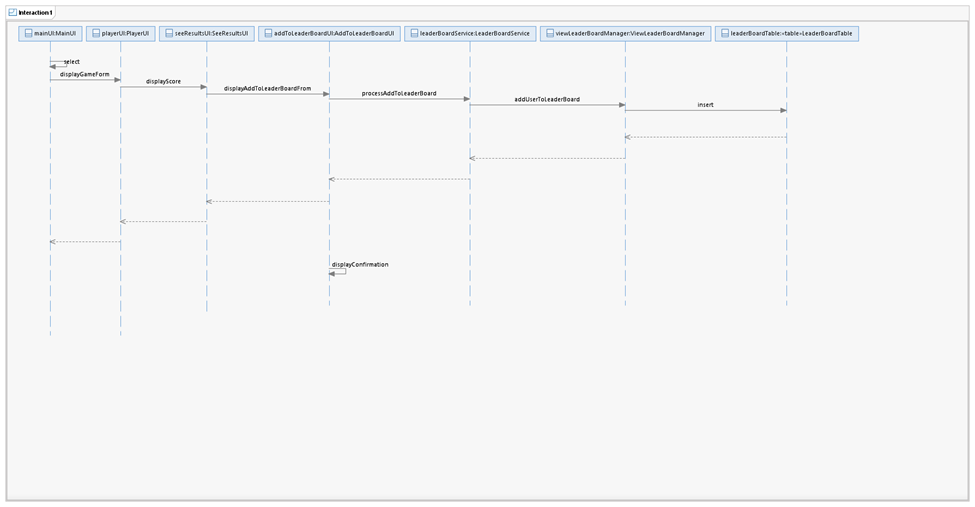
****

**2.2.4.2 ApproveQuiz Realization Description**

First the application displays a form of the main menu, the user can select to be an admin. Then an admin form and admin login form are shown. The user need to enter id and password. If the id and the password inputted by the user can be found in the AdminLoginTable, an admin login confirmation will be shown. After that, the user can select “ApproveQuiz” in the AdminUI. The list of quizzes to be approved are selected from the QuizListTable in the database. Once a quiz is approved, the QuizListTable would be updated to make the quiz available in the game playing.

**2.2.5 AddToLeaderBoard Realization**

**2.2.5.1 AddToLeaderBoard Realization Sequence Diagram**

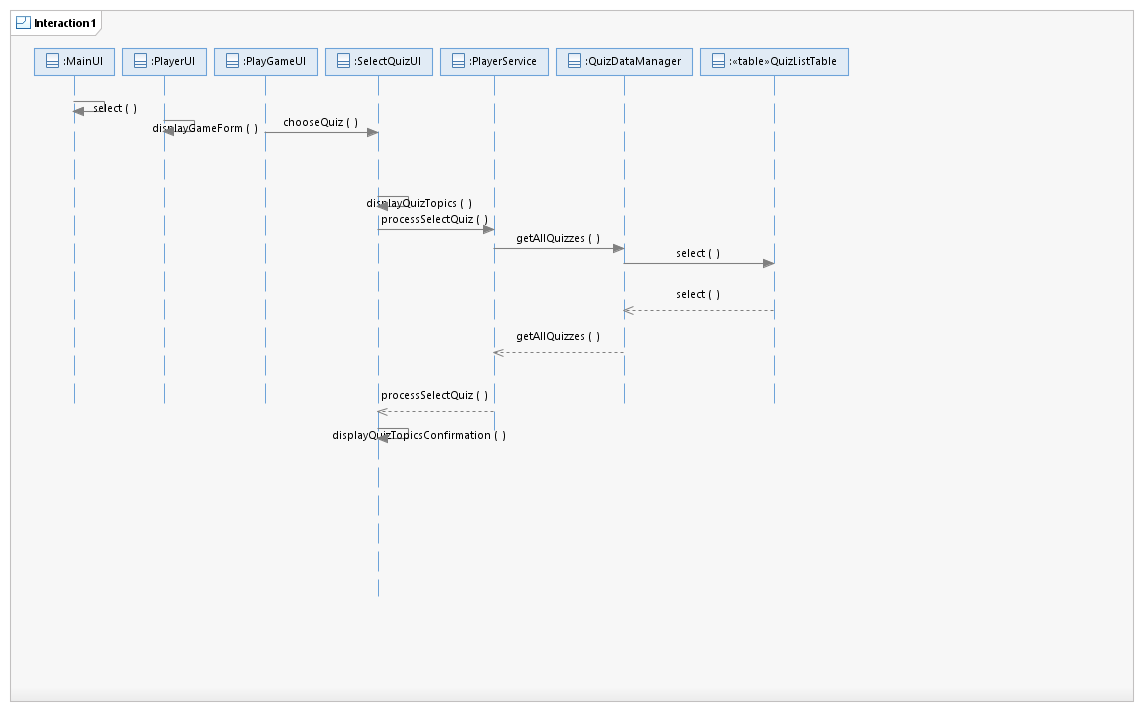
****

**2.2.5.2 AddToLeaderBoard Realization Description**

An user as a normal player can choose to play game in the main menu, after playing the game, the application will show his/her result. The user will be asked whether add the result to leaderboard or not, if yes, a form will be shown for the user to enter his/her name. Then the name and the result will be inserted in the LeaderBoardTable.

**2.2.6 SelectQuiz Realization**

**2.2.6.1 SelectQuiz Realization Sequence Diagram**

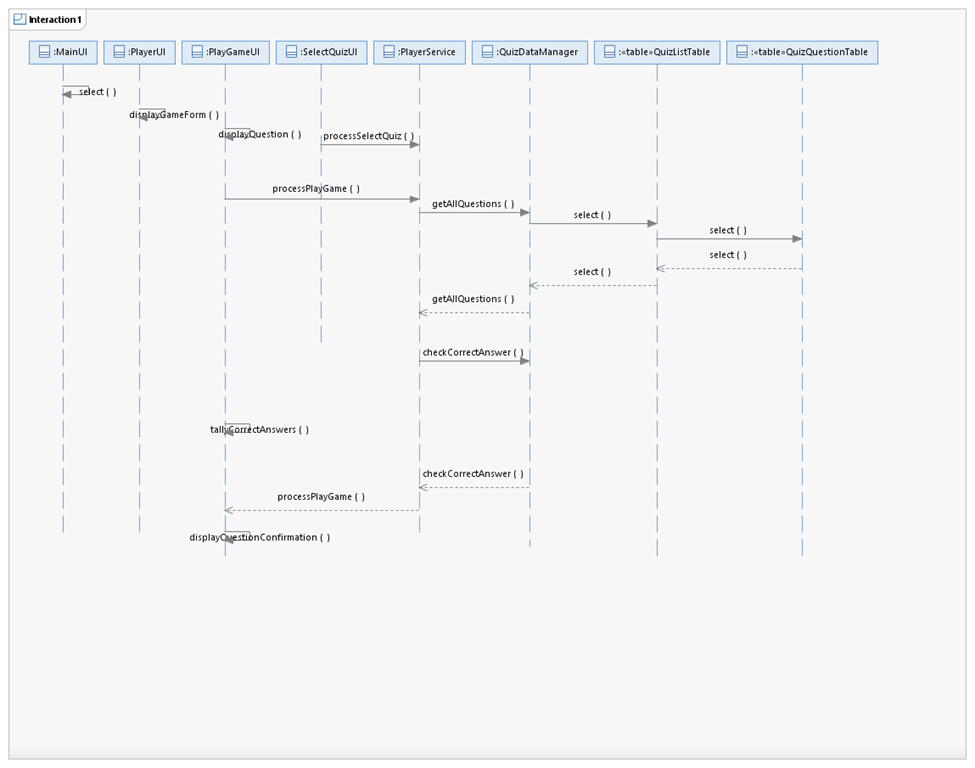


**2.2.6.2 SelectQuiz Realization Description**

In the select quiz sequence diagram we start from the main menu, we select the option to play a game and then we have to select a quiz. From the playerUI when then choose displayGameForm, from here we need to choose which quiz we want to take so we have to displayQuizTopics, this needs us to access the database to get all the quizzes that are in the database so we can choose one. We then need to process this call and after that we have to call the getAllQuizzes method which will return a list of all the quizzes from the database. The getAllQuizzes method selects the quizzes from the QuizListTable.

**2.2.7 PlayGame Realization**

**2.2.7.1 PlayGame Realization Sequence Diagram**

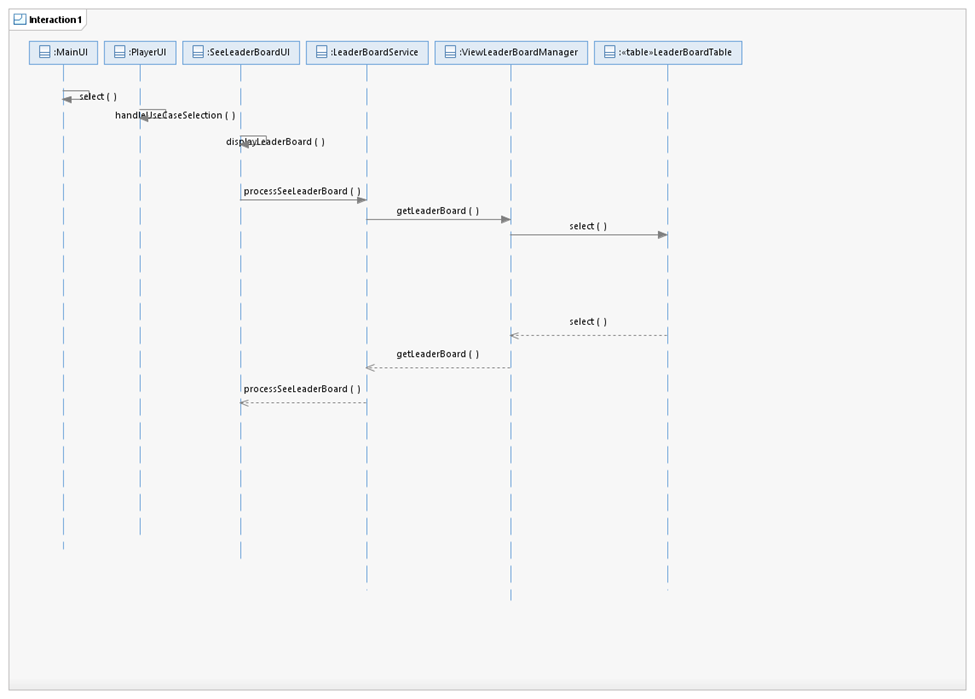


**2.2.7.2 PlayGame Realization Description**

In play game we start from the main menu and select to be a player and to play a game. From the PlayerUI we are displayed the game form, which in turn we need to display a question to answer. Display question needs to get a question from the database. We use the getAllQuestions method, which would get all the questions for the selected quiz that we want to complete. We would then have a for-loop to answer the questions one by one as we play the game. The getAllQuestions method selects the quiz that was chosen from the QuizListTable and then selects the questions for that table from the QuizQuestionTable. Also while we are playing the game we need to checkCorrectAnswers, which once the user locks in their answer for the selected question this method would be used to see if we got that question right. Finally we have another method that tallys the number of answers that the user got correct, then once all the questions are answered we display a confirmation message that the quiz has been completed.

**2.2.8 SeeLeaderBoard Realization**

**2.2.8.1 SeeLeaderBoard Realization Sequence Diagram**



**2.2.8.2 SeeLeaderBoard Realization Description**

In SeeLeaderBoard you start at the main menu and you would select to see the leaderboard, this would then take you to the PlayerUI. From the PlayerUI you would then use the method handleUseCaseSelection, which would then ask the system to display the leaderboard. To display the leaderboard we have to process that call and select the leaderboard from the database. We use the method getLeaderBoard to select the leaderboard from the database and show the leaderboard to the user in the seeLeaderBoardUI.