

COP 5339 Object Oriented Software Design Project

Design Specifications

Team #6: Alejandro Reyes, Leigh Chin, and Tsz Shing Tsoi

Project Title: Exciting Trivia Game

Important Details: This offline application is a single player game that will run on Java Virtual Machine. It will be developed using Java Swing Library. Additional two players are simulated (not real players).

Application Requirements - Updated

Functional Specification with GUI¹

Upon launching the application, the user will get a splash screen with “Welcome to Exciting Trivia Game” message. Then a game menu will display a text field to indicate where the user will enter his or her name and a dropdown menu for the user to pick a difficulty level (novice, intermediate or expert). Default will be novice. The game menu will have four additional buttons: Start, View Scores, Reset and Exit. The Start, View Scores and Reset buttons will initially be deactivated until the user’s name is provided. Exit will close the application, View Scores will launch a screen with the user’s top 10 saved scores and the corresponding ranking, and Reset will erase all user data and scores. Start continues to begin the game and opens the game screen.

On the upper left corner of the game screen, a message will display in a text area, “Tom has entered the game. Jane has entered the game.” to make the user aware of the other simulated players. On the upper right corner will be a tally marking the points earned by each player in a text area. On the center of the screen will be a read only large text area that will display a question followed by some choices a, b, c or d. Under the text area, there will be six buttons. One button for each choice of the question, a next question button and an end game button. The user can end the game at any time. The next button will display a trivia question followed by the possible choices. Once the next button is clicked, the button will be disabled until someone gets the answer or everyone has answered incorrectly. Each player can only choose once. The first player to answer correctly gets the point. A message will display on the large text area on the center with the order of the answers submitted by each player and the correct answer. At this point, the next question button is enabled. Once clicked, the large text area will display the next question.

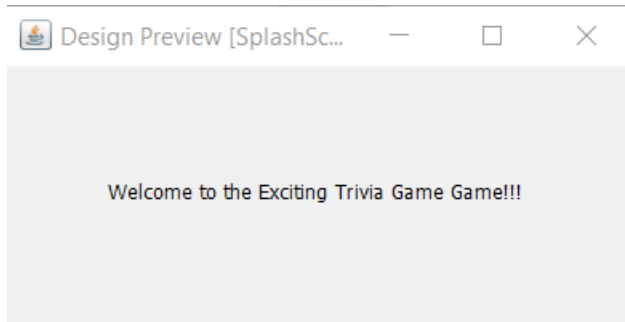
Once the user clicks the end game button, the end game screen will appear with the scores and ranking. This screen will have one Close button to close the end game screen and return back to the game menu. If the user plays long enough to answer all the questions, the game will end, and the end game screen will appear with the scores and ranking.

¹ Initial functional specifications and use cases are included in Attachment A.

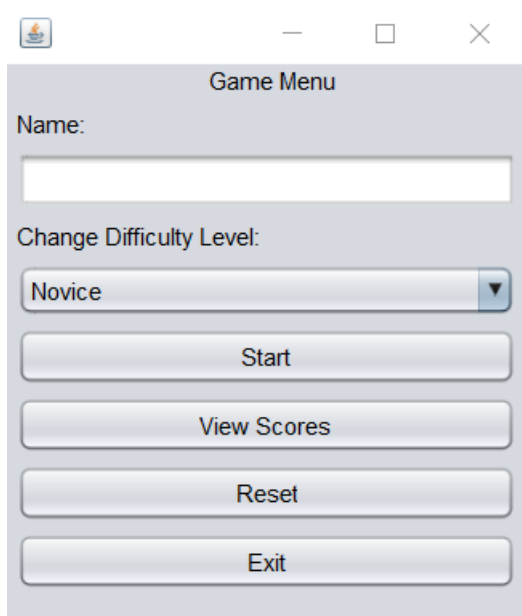
Use Cases with GUI¹

UC1. Launch Application

1. User executes the application.
2. System displays a splash screen with the following message:



3. After the splash screen ends, system displays the game menu with the following items:



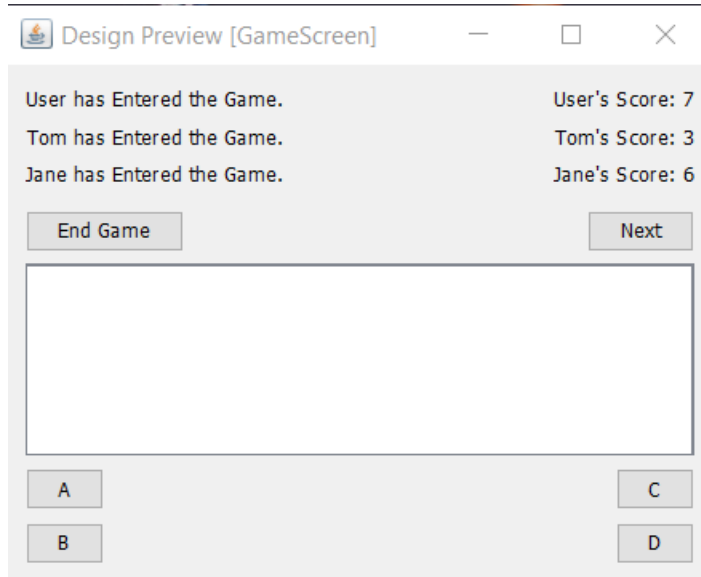
4. User enters his or her name in the text field.
5. User clicks the dropdown menu underneath the "Change Difficulty Level" label.
6. System displays the following list of options:
Novice
Intermediate
Expert
7. User selects "Novice" from the list of options.
8. All game menu buttons are activated.

UC1. Launch Application – Variation #1

- 1.1. Start at **UC1. Launch Application** Step 2.
- 1.2. System detects an existing user's name and loads the user data. Splash Screen ends.
- 1.3. Continue at **UC1. Launch Application** Step 8.

UC2. Start a New Game

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User clicks the "Start" button.
3. System displays the game screen as illustrated below:



4. System displays players joining the game in a text area in sequence on the upper left corner of the game screen.
5. System displays the current scores of each player on the upper right corner of the game screen in a text area.
6. System displays the end game button and next button above the text area in the center of the screen, and four buttons for choices a, b, c and d below the text area in the center of the screen.
7. System displays the following message in a text area in the center of the screen:

Welcome! Please click "Next" to begin the Exciting Game!

UC3. Scoring a Point

1. User carries out **UC2. Start a New Game**.
2. User clicks the "Next" button.
3. System displays a new question in a text area following by four answer choices a, b, c or d.
4. User clicks on any of the buttons {a, b, c, d} to select the answer to the question.
5. After all players select an answer or 10 seconds have elapsed, the system will display the answers in a text area picked by each player in descending order of quickest to answer along with the correct answer.
6. System will increment the score for the winner of the round and display the updated score on the upper right corner.

UC4. End the Game

1. User carries out **UC3. Scoring a Point**.
2. User clicks on the “End” button.
3. System displays the end game screen showing each player and their scores and ranking, and a “Back to Game Menu” button.
4. User clicks the “Back to Game Menu” button.
5. System displays the game menu.

UC4. End the Game – Variation #1:

- 1.1. Start at **UC4. End the Game** Step 1.
- 1.2. User clicks Next button until all questions are exhausted.
- 1.3. Continue at **UC4. End the Game** Step 3.

UC5. Reset User Data

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User clicks the “Reset” buttons.
3. System displays the following message:

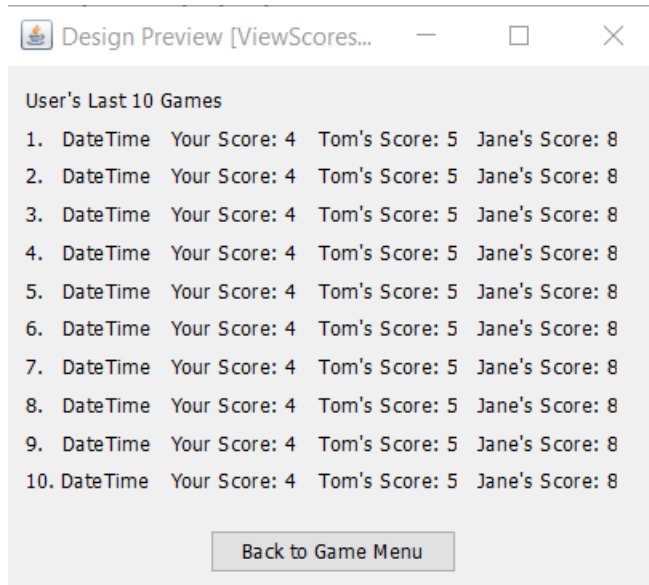
`All user data and scores will be erased.
Continue? Yes / No`
4. User clicks the “Yes” button.
5. System erases all user data and scores and displays the game menu.

UC5. Reset User Data – Variation #1

- 1.1. Start at **UC5. Reset User Data** Step 3.
- 1.2. User selects “No”.
- 1.3. System displays the game menu.

UC6. View Scores

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User clicks the “View Scores” button.
3. System displays the view score screen showing the top 10 saved scores and rankings, follow by a “Back to Game Menu” button:



4. User clicks the “Back to Game Menu” button.
5. System displays the game menu.

UC7. Change Difficulty Level

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User selects the dropdown menu underneath the “Change Difficulty Level” label.
3. System displays the following list of options:

Novice
Intermediate
Expert

4. User selects a new difficult level from the list of options.
5. The difficult level is changed.
6. System displays the game menu.

UC8. Exit the Application

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User clicks the “Exit” button.
3. The application closes.

Main Actors

1. User
2. System

Glossary

End Game Button: A button used to display scores and stop playing the game.

End Game Screen: A screen where the user can see the score after he or she finished the game and return to the Game Menu. This screen is similar to the View Scores Screen except the current scores are shown.

Exit Button: A button to exit the application.

Game Menu: A set up menu where the user can enter his or her name, view recent scores, clear the history or select a difficulty level and begin a new game.

Game Screen: A screen where the user can play the game. The user can answer a trivia question and score a point.

Next Button: A button used to display the next question in the game playing session.

Reset Button: A button to clear the users game history.

Score Points: The number of questions answered correctly.

Splash Screen: The initial screen shown to the user when the application is first launched.

Simulated Player: A system player. The system will act as a competitor and try to make the game challenging for the user.

Start Button: A button used to begin playing a new game.

System: The Exciting Trivia Game application.

User: A real player or human who is playing the Exciting Trivia Game.

View Scores Button: Displays a new screen with the user's recent scores.

View Scores Screen: The screen to display the user's recent scores.

CRC Cards

Note: Classes from Java APIs are not included. Please see Java documentation and class diagrams for details of these classes from Java APIs.

Domain Classes

Class: GameSession

Responsibilities:

Manage application states

Manages UI screens

Collaborators: GameSystem, Game, SplashScreen, GameMenu, GameScreen, ViewScoreScreen, JPanel, JFrame

Class: GameSystem

Responsibilities:

Manage application

Manage user data

Manage question list

Start a game

Collaborators: Game, Player, QuestionList, DataSaver, DataLoader

Class: Game

Responsibilities:

- Manage states of a game play
- Manage date (e.g. scores, winners) during a game play

Collaborators: Judge, Player, Timer, Question, Iterator

Class: DataSaver

Responsibilities:

- Save player data to a file

Collaborators: Player

Class: DataLoader

Responsibilities:

- Load player data from a file
- Load questions from a file

Collaborators: Player, Question

Class: QuestionList

Responsibilities:

- Manage questions

Collaborators: Question, Iterator

Class: Question

Responsibilities:

- Maintain a question and answers

Collaborators: None

UI Classes

Class: GameMenu

Responsibilities:

- Hold a JTextField to enter users name for score keeping.
- Hold a Start JButton to begin a new game.
- Hold a View score JButton to see score history.
- Hold a Reset JButton to clear the scores.
- Hold an Exit JButton to leave the game.

Collaborators: JPanel, JButton, JLabel, JTextField, JComboBox

Class: QuestionByDifficulty

Responsibilities:

- Criteria used to determine if a question should be used for the current game

Collaborators: None

Class: Player

Responsibilities:

- Manage player information (e.g. name, difficulty, scores)

Collaborators: None

Class: SimulatedPlayer

Responsibilities:

- A Player but answers are generated based on correct answer

Collaborators: None

Class: Judge

Responsibilities:

- Keep track of answers from Player
- Determine winner of the current round

Collaborators: Player

Class: GameScreen

Responsibilities:

- Display a question in a JTextArea.
- Display the results of a round in JTextArea.
- Display current scores for each player in a JLabel.
- Introduce players at start of the game in a JLabel.
- Display four JButton for each answer choice available (a, b, c, d).

Collaborators: JPanel, JButton, JLabel, JTextArea

Class: ViewScore

Responsibilities:

- Display the results of the current game in a JLabel.
- Display a history of games in a collection of JLabels.
- Hold a JButton to return the user to the MenuGame Screen.

Collaborators: JPanel, JButton, JLabel

Class: SplashScreen

Responsibilities:

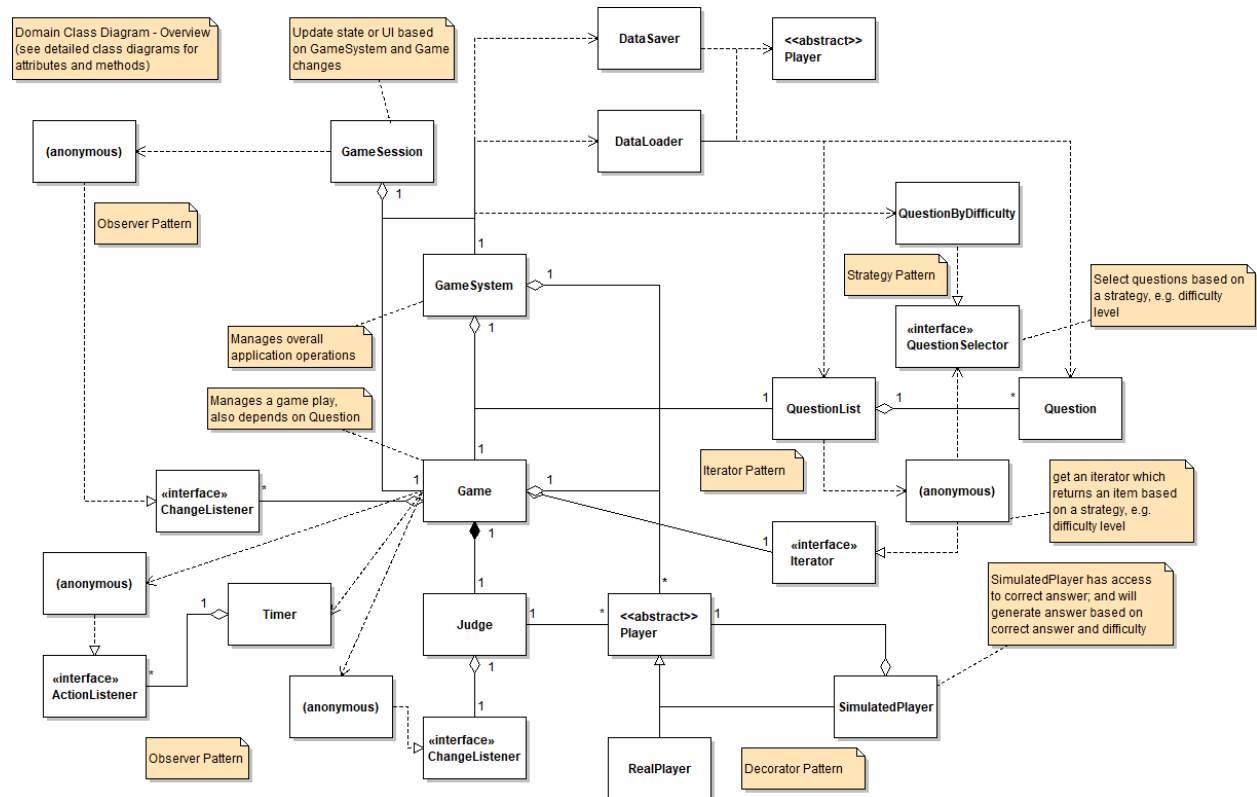
- Display a welcome message to the user using a JLabel.

Collaborators: JPanel, JLabel

UML Diagrams

Class Diagrams

Domain Class Diagrams



Pattern Mapping

Observer Pattern (Game/Judge)	
Name in Design Pattern	Actual Name
Subject	Game, Judge
Observer	ChangeListener
ConcreteObserver	Anonymous class that implements the ChangeListener interface type
attach()	addChangeListener()
notify()	stateChanged()

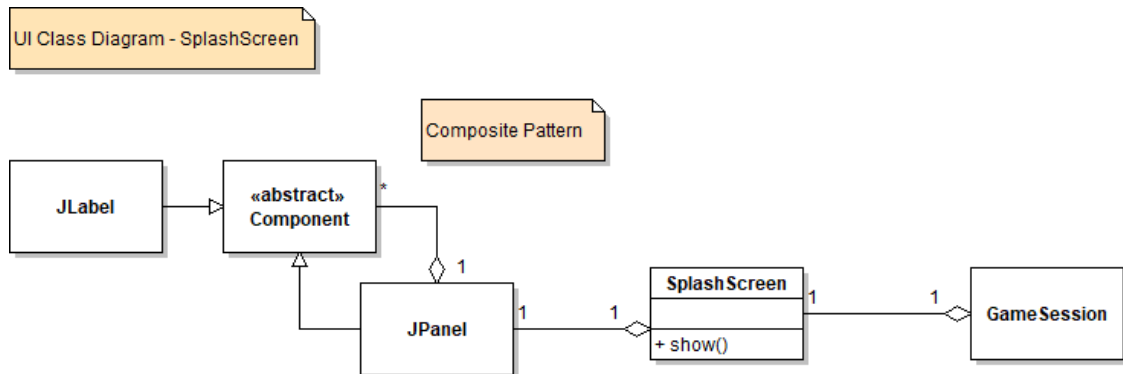
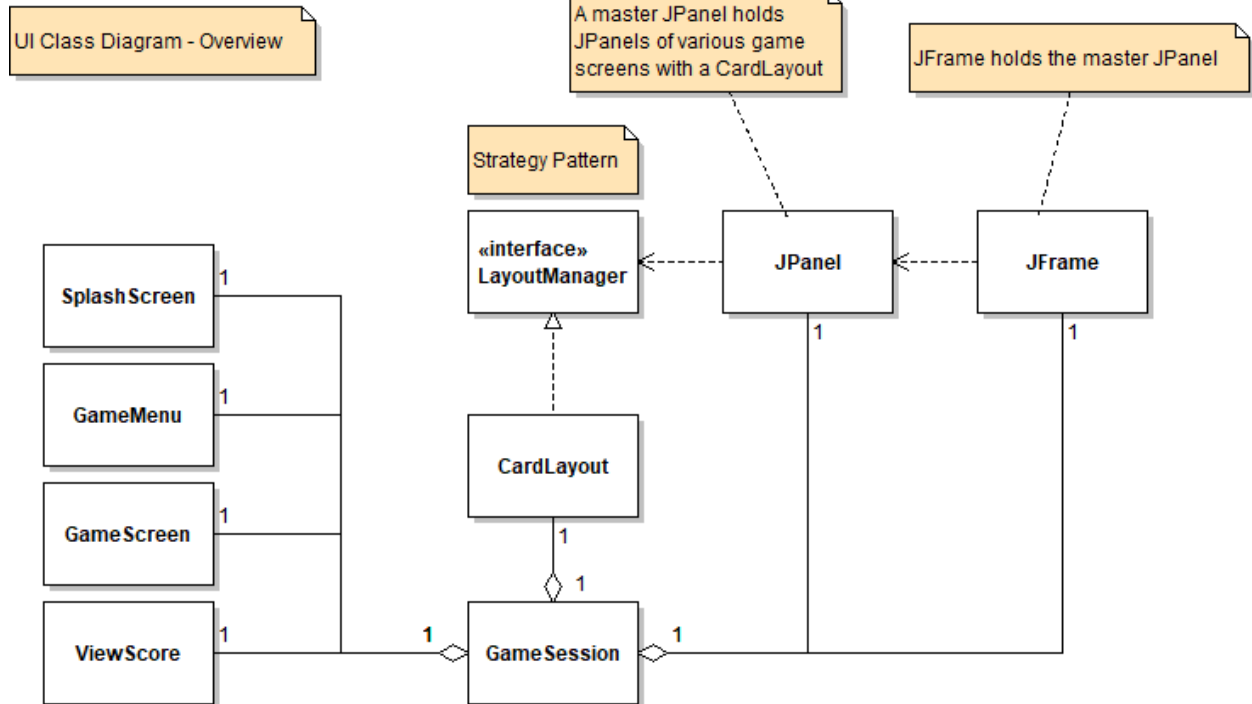
Observer Pattern (Timer)	
Name in Design Pattern	Actual Name
Subject	Timer
Observer	ActionListener
ConcreteObserver	Anonymous class that implements the ActionListener interface type
attach()	addActionListener()
notify()	actionPerformed()

Decorator Pattern (Player)	
Name in Design Pattern	Actual Name
Component	Player
ConcreteComponent	RealPlayer
Decorator	SimulatedPlayer
method()	submitAnswerToJudge(); getName(); setName(); getCurrentScore(); incrementScore(); updateScoreHistory(); getScoreHistory(); setDifficulty(); getDifficulty()

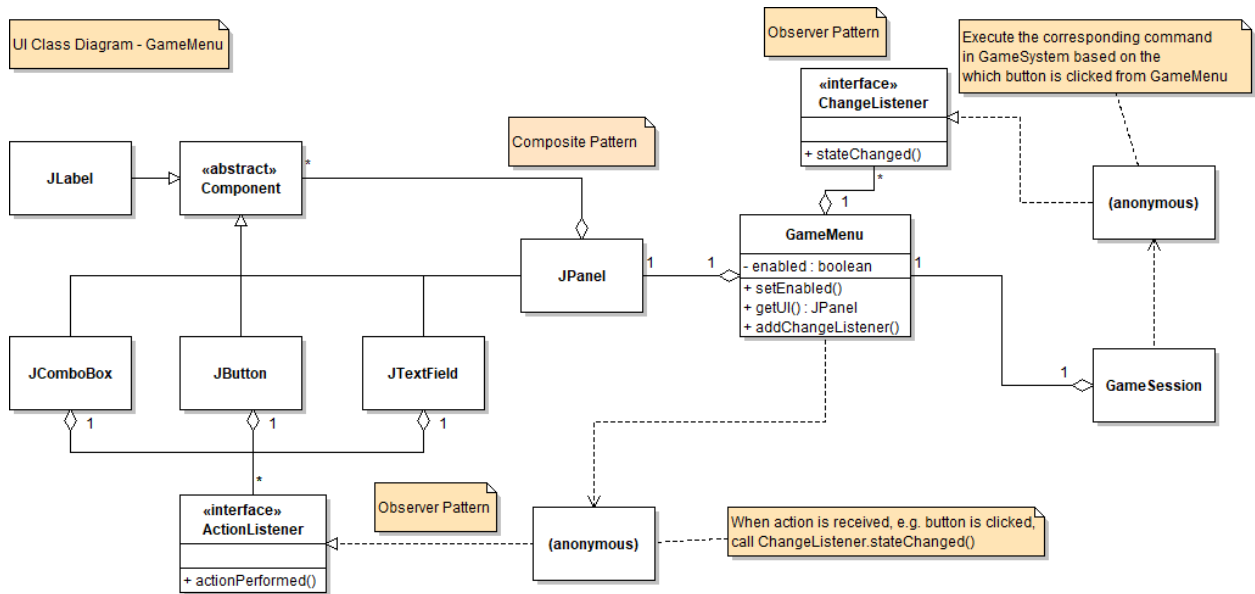
Iterator Pattern (QuestionList)	
Name in Design Pattern	Actual Name
ConcreteAggregate	QuestionList
Iterator	Iterator
ConcreteIterator	Anonymous class implementing Iterator
createIterator()	getQuestionIterator()
next()	next()
isDone()	Opposite of hasNext()
currentItem()	return value of hasNext()

Strategy Pattern (QuestionSelector)	
Name in Design Pattern	Actual Name
Context	Anonymous class implementing Iterator and using QuestionSelector to determine the next item
Strategy	QuestionSelector
ConcreteStrategy	QuestionByDifficulty
doWork()	IsValidQuestion()

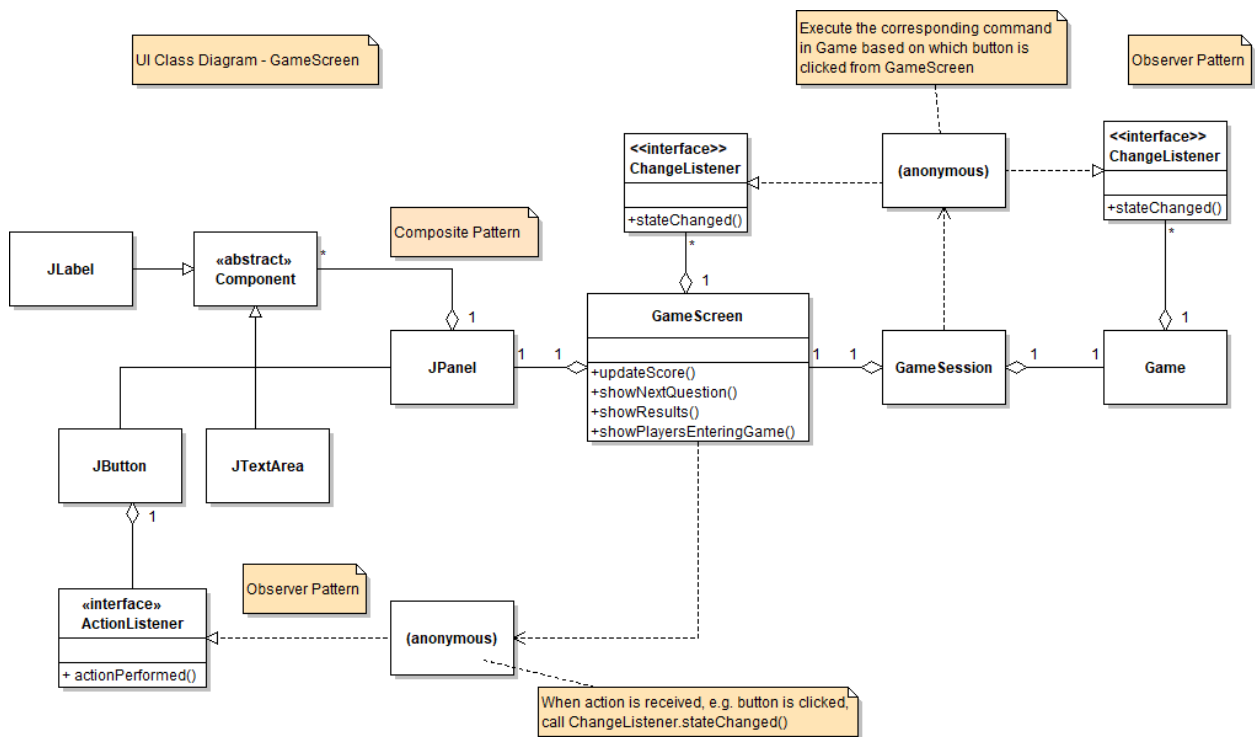
UI Class Diagrams

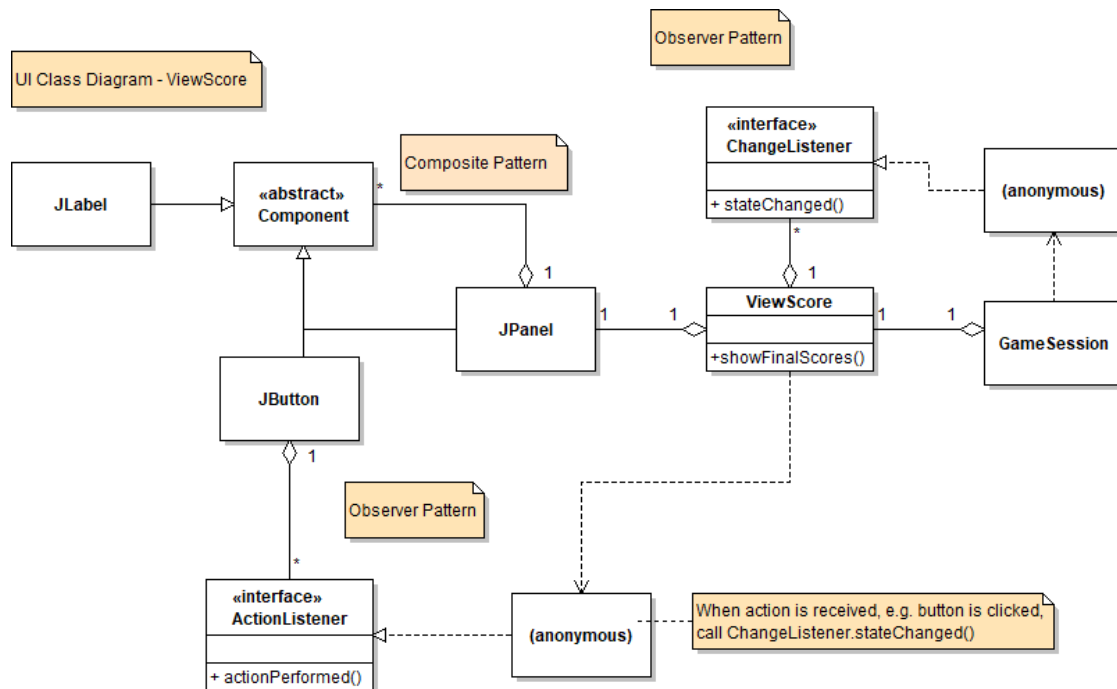


UI Class Diagram - GameMenu



UI Class Diagram - GameScreen





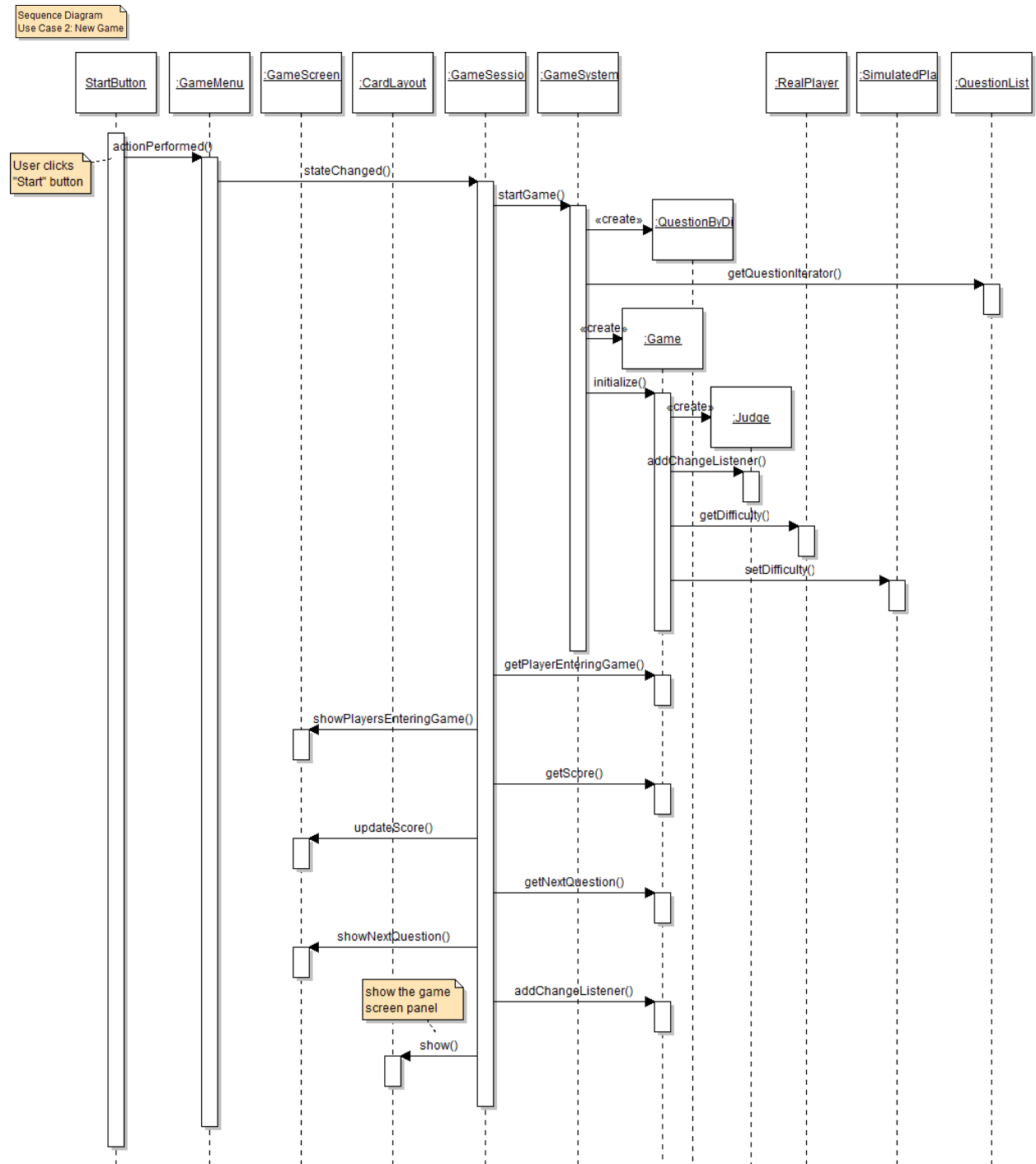
Pattern Mapping

Observer Pattern (JComponent)	
Name in Design Pattern	Actual Name
Subject	JButton, JtextField, JComboBox
Observer	ActionListener
ConcreteObserver	Anonymous class that implements the ActionListener interface type
attach()	addActionListener()
notify()	actionPerformed()

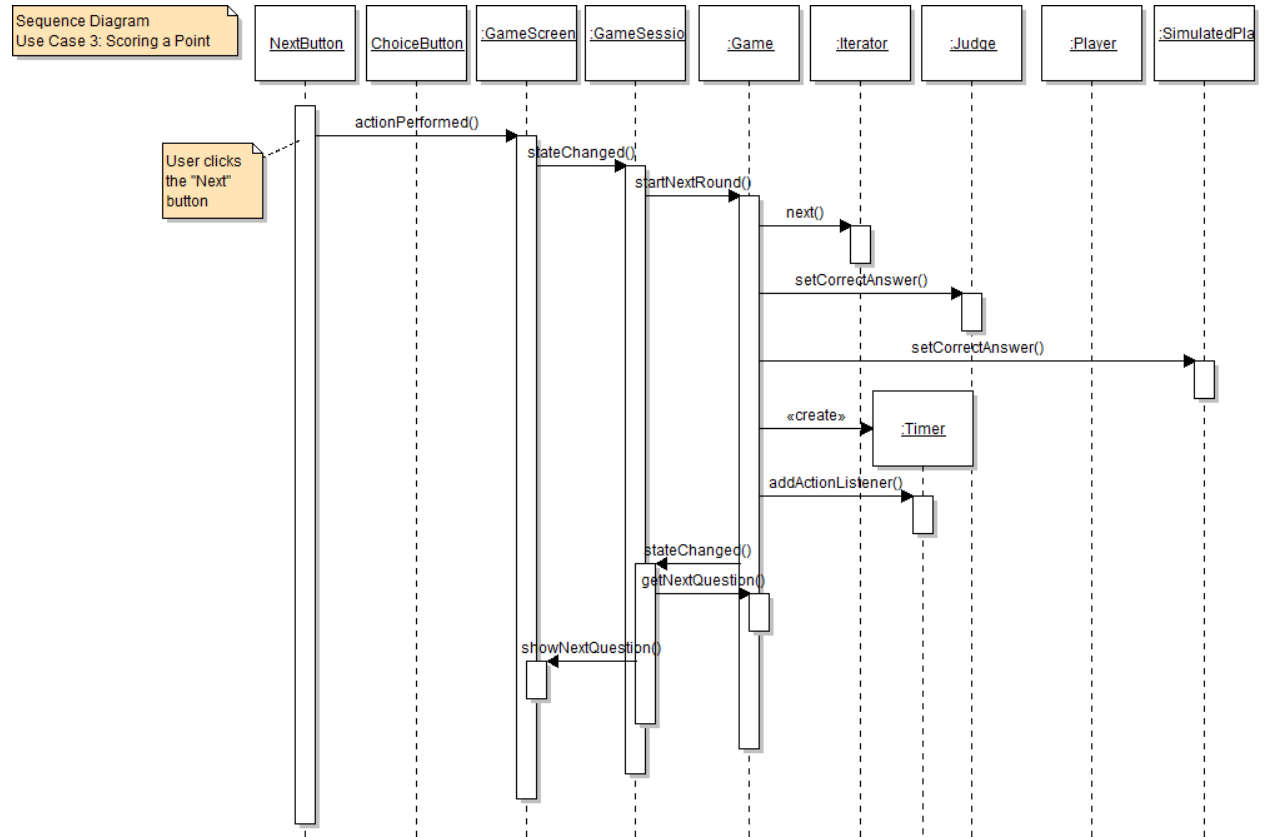
Observer Pattern (UI Classes)	
Name in Design Pattern	Actual Name
Subject	SplashScreen, GameMenu, GameScreen, ViewScore
Observer	ChangeListener
ConcreteObserver	Anonymous class that implements the ChangeListener interface type
attach()	addChangeListener()
notify()	stateChanged()

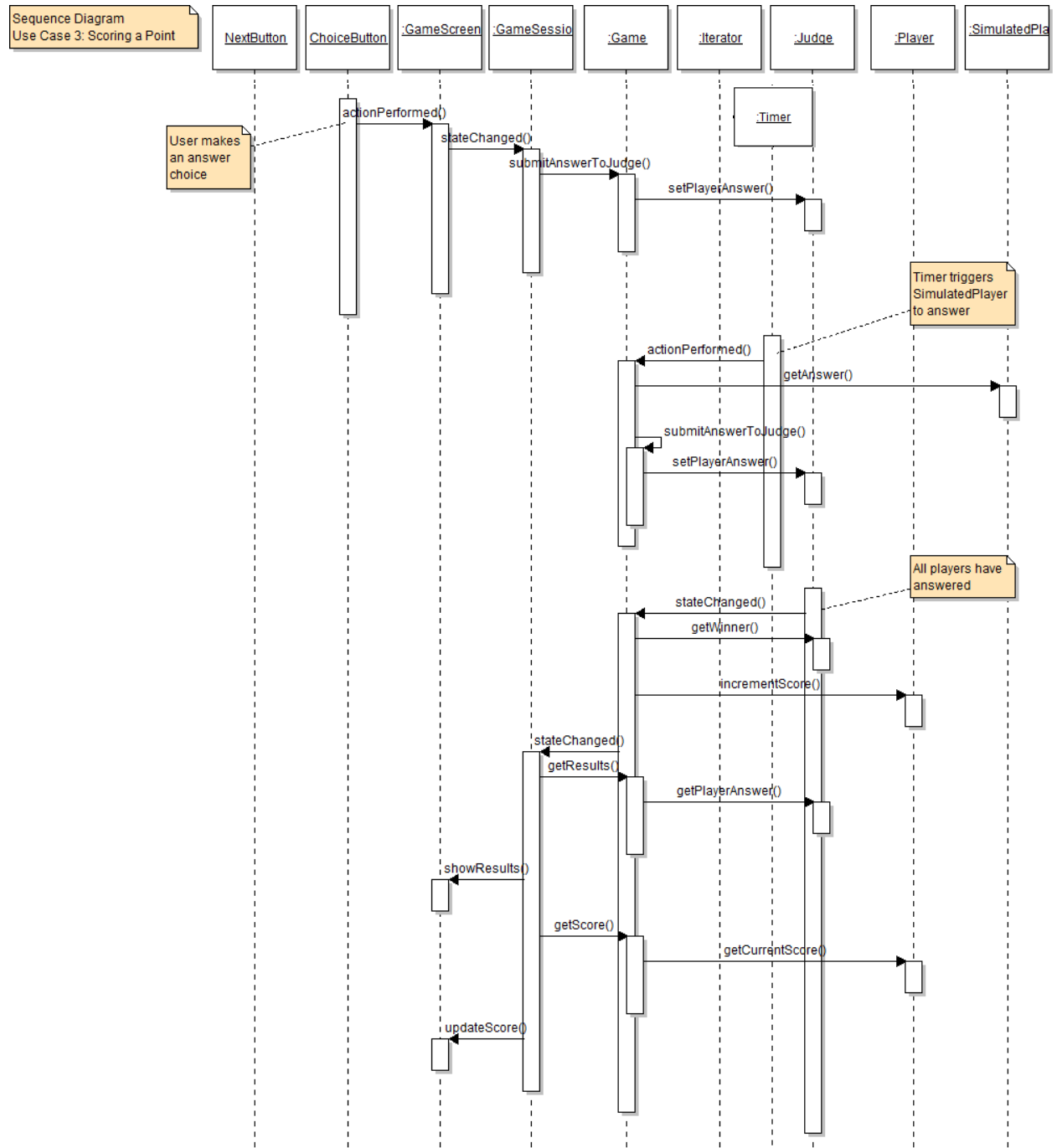
Sequence Diagrams

UC2. Start a New Game

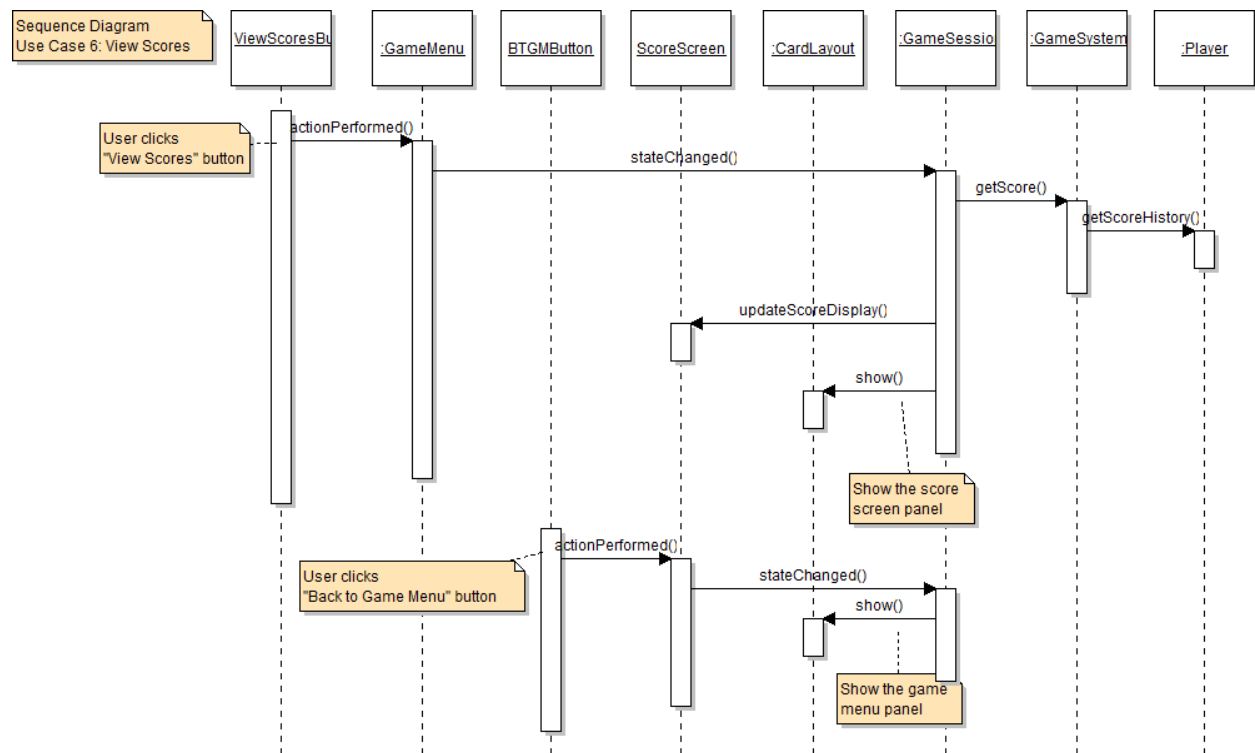


UC3. Scoring a Point

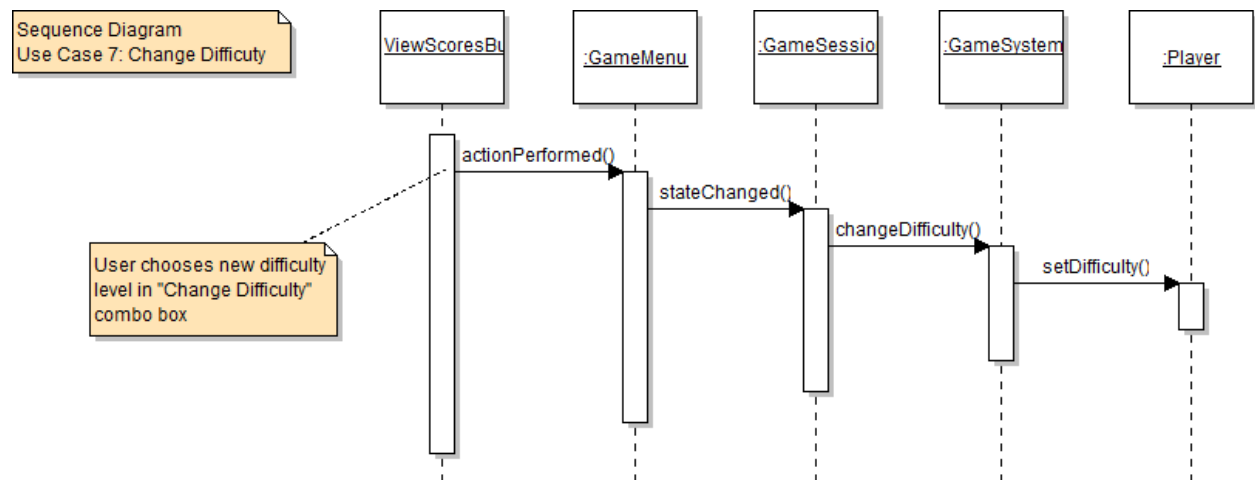




UC6. View Scores

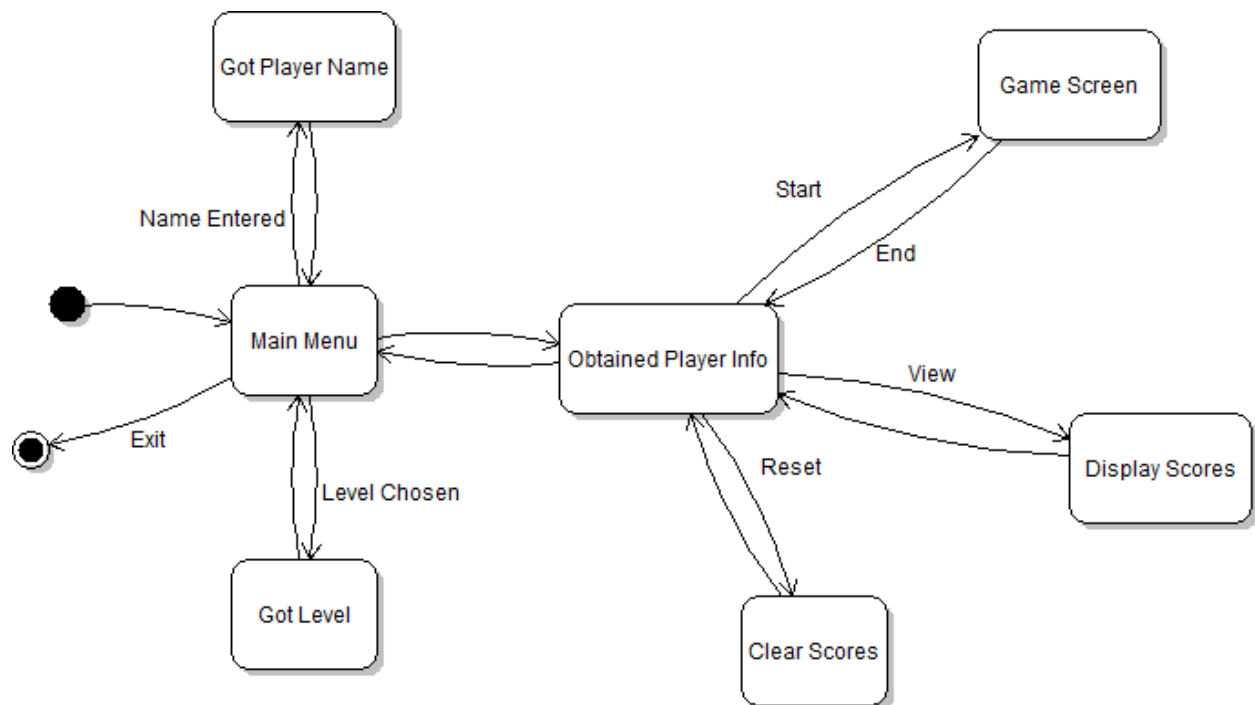


UC7. Change Difficulty Level

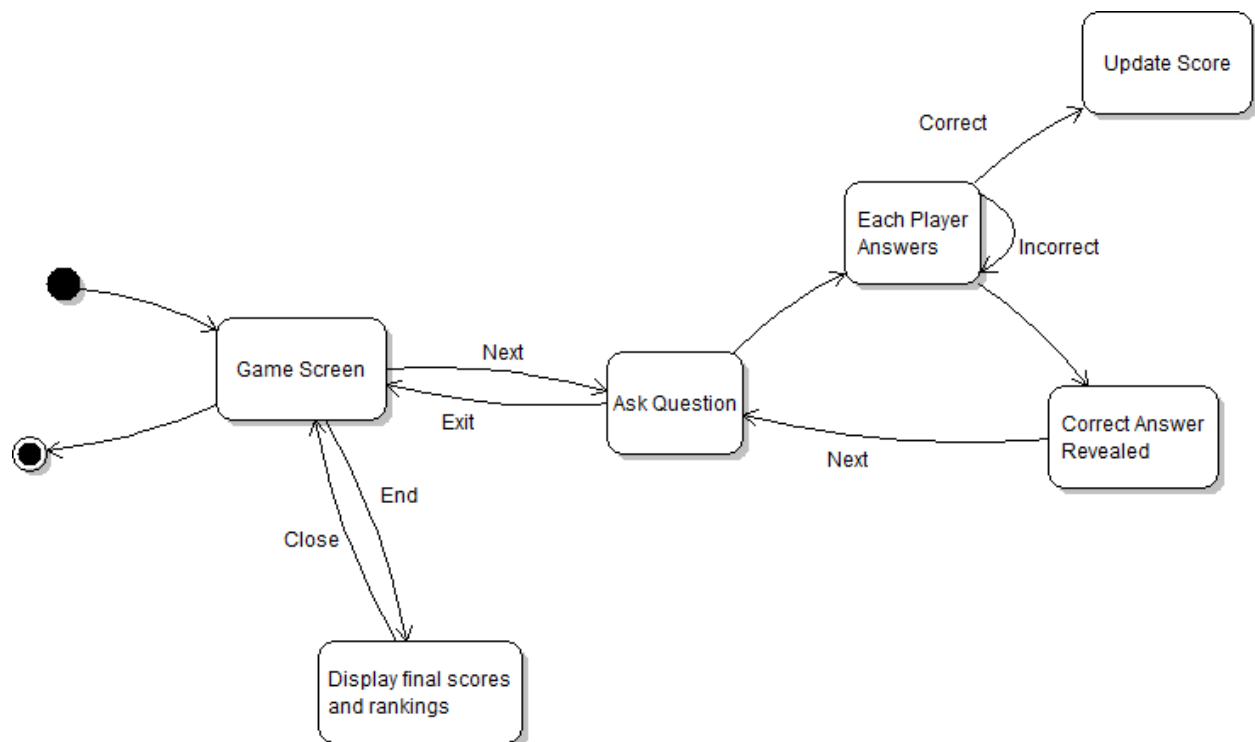


State Diagrams

GameSession



Game



Revision History

Version	Date	Description	Revised By:
0.1	1/31/2020	Initial draft, functional specification, UCs 2, 3, 4, 8	AR
0.2	2/3/2020	Added UCs 1, 5, edited UC2	TST
0.3	2/7/2020	Added UCs 6, 7	LC
0.4	2/11/2020	Edited document for consistency	TST
1.0	2/12/2020	Formatted document for review and submission	TST
1.1	3/10/2020	Modified according to professor's comments	AR
1.2	3/10/2020	Revised requirements with GUI	TST
1.3	3/15/2020	Renamed document, added section placeholders, adding pattern mapping, inserted initial application requirements as attachment, added glossary items	TST
1.4	3/17/2020	Added sequence diagrams and state diagrams	LC
1.5	3/19/2020	Inserted GUI from AR, revised use cases, added domain CRC, class diagrams, revised sequence diagrams	TST
2.0	3/19/2020	Inserted UI class diagrams and UI CRC from AR, made final revisions for submission	TST

Appendix A: Initial Application Requirements

Functional Specification

This will be a single player application which runs offline. Upon launching the application, the user will get a splash screen with “Welcome to Exciting Trivia Game” message. Then a game menu will display where the user will enter his or her name and picks a difficulty level (novice, intermediate or expert). Default will be novice. The game menu will have four additional buttons: Start, View Scores, Reset and Exit. The Start, View Scores and Reset buttons will initially be deactivated until the user’s name is provided. Exit will close the application, View Scores will launch a screen with the user’s top 10 saved scores and the corresponding ranking, and Reset will erase all user data and scores. Start continues to begin the game and opens the game screen.

On the upper left corner of the game screen, a message will display, “Tom has entered the game. Jane has entered the game.” to make the user aware of the other simulated players. On the upper right corner will be a tally marking the points earned by each player. On the center of the screen will be a read only large text box that will display a question followed by some choices a, b, c or d. Under the text box, there will be six buttons. One button for each choice of the question, a next question button and an end game button. The user can end the game at any time. The next button will display a trivia question followed by the possible choices. Once the next button is clicked, the button will be disabled until someone gets the answer or everyone has answered incorrectly. Each player can only choose once. The first player to answer correctly gets the point. A message will display on the large text box on the center with the order of the answers submitted by each player and the correct answer. At this point, the next question button is enabled. Once clicked, the large text box will display the next question.

Once the user clicks end game, the end game screen will appear with the scores and ranking. This screen will have one Close button to close the end game screen and return back to the game menu. If the user plays long enough to answer all the questions, the game will end, and the end game screen will appear with the scores and ranking.

Use Cases

UC1. Launch Application

1. User executes the application.
2. System displays a splash screen with the following message:

`Welcome to Exciting Trivia Game`
3. After the splash screen ends, system displays the game menu with the following items:

`Your Name: (a text box)`
`Change Difficulty Level:`
`Novice (default)`
`Intermediate`
`Expert`
`Start (initially deactivated)`
`View Scores (initially deactivated)`
`Reset (initially deactivated)`
`Exit`
4. User enters his or her name in the textbox and selects "Novice".
5. All game menu items are activated.

UC1. Launch Application – Variation #1

- 1.1. Start at **UC1. Launch Application** Step 2.
- 1.2. System detects an existing user's name and loads the user data. Splash Screen ends.
- 1.3. Continue with **UC1. Launch Application** Step 5.

UC2. Start a New Game

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User select "Start".
3. System displays the game screen.
4. System displays the following messages in sequence on the upper left corner of the game screen:

`<username> has entered the game...`
`Tom has entered the game...`
`Jane has entered the game...`
5. System displays the current scores (initially zero) of each player on the upper right corner of the game screen.
6. System displays the following message in the center of the screen:

`Welcome! Please click "Next" to begin the Exciting Game!`
7. System displays the following six options on the bottom of the game screen:

`A B C D Next End`

UC3. Scoring a Point

1. User carries out **UC2. Start a New Game**.
2. User clicks the "Next" button.
3. System displays a new question following by four answer choices a, b, c or d.
4. User clicks on any of the buttons {a, b, c, d} to select the answer to the question.
5. After all players select an answer or 10 seconds have elapsed, the system will display the answers picked by each player in descending order of quickest to answer along with the correct answer.
6. System will increment the score for the winner of the round and display the updated score on the upper right corner.

UC4. End the Game

1. User carries out **UC3. Scoring a Point**.
2. User clicks on the "End" button.
3. System displays the end game screen showing each player and their scores and ranking, and a "Close" button.
4. User clicks the "Close" button.
5. System displays the game menu.

UC4. End the Game – Variation #1:

- 1.1. Start at **UC4. End the Game** Step 1.
- 1.2. All questions are exhausted in the current game.
- 1.3. Continue with **UC4. End the Game** Step 3.

UC5. Reset User Data

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User selects "Reset".
3. System displays the following message:

`All user data and scores will be erased.
Continue? Yes / No`
4. User selects "Yes".
5. System erases all user data and scores and displays the game menu.

UC5. Reset User Data – Variation #1

- 1.1. Start at **UC5. Reset User Data** Step 3.
- 1.2. User selects "No".
- 1.3. System displays the game menu.

UC6. View Scores

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User selects "View Scores".
3. System displays the top 10 saved scores and rankings.
4. User is prompted to:

Press Enter

5. System displays the game menu.

UC7. Change Difficulty Level

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User selects "Change Difficulty Level".
3. User is prompted to enter new level:

Difficulty Level:
Novice
Intermediate
Expert

4. The difficult level is changed.
5. System displays the game menu.

UC8. Exit the Application

1. User carries out **UC1. Launch Application** or other use cases which return to the game menu.
2. User clicks the "Exit" button.
3. The application closes.