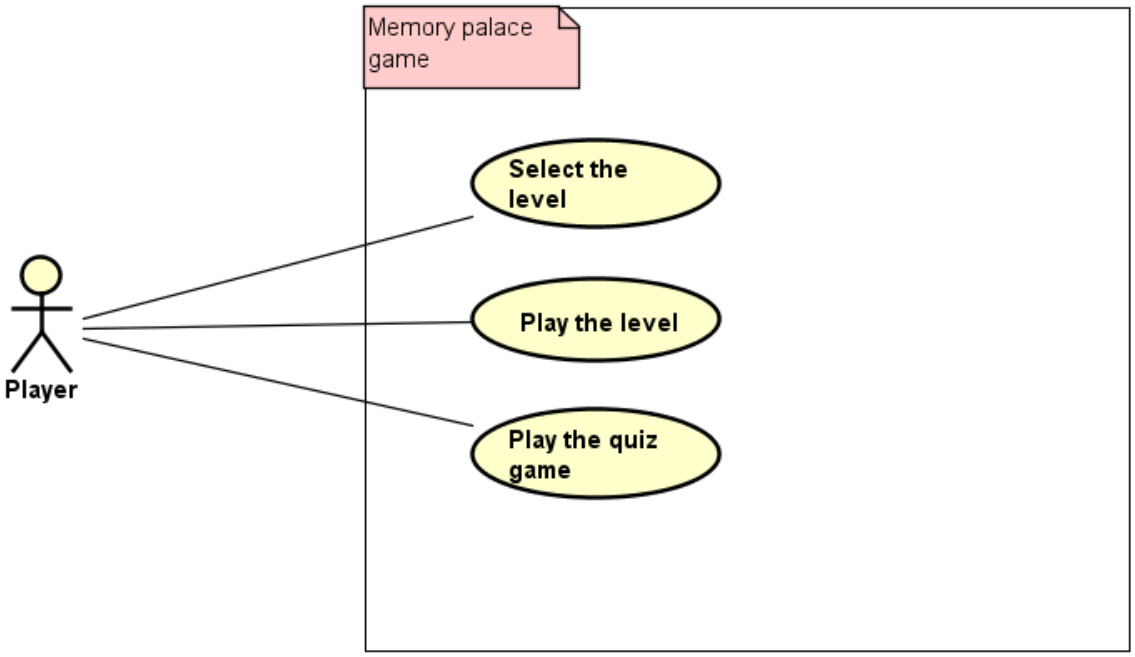
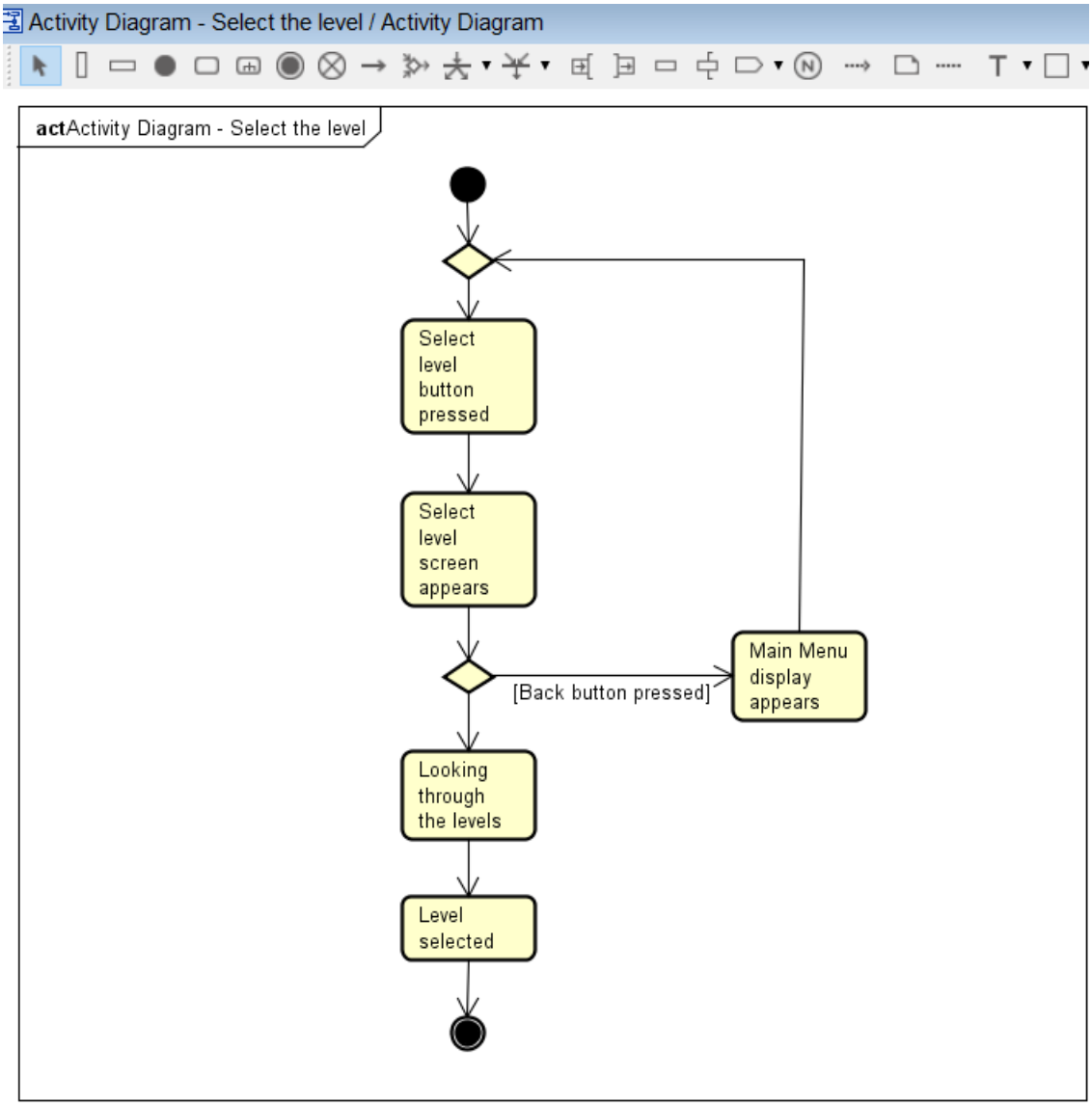
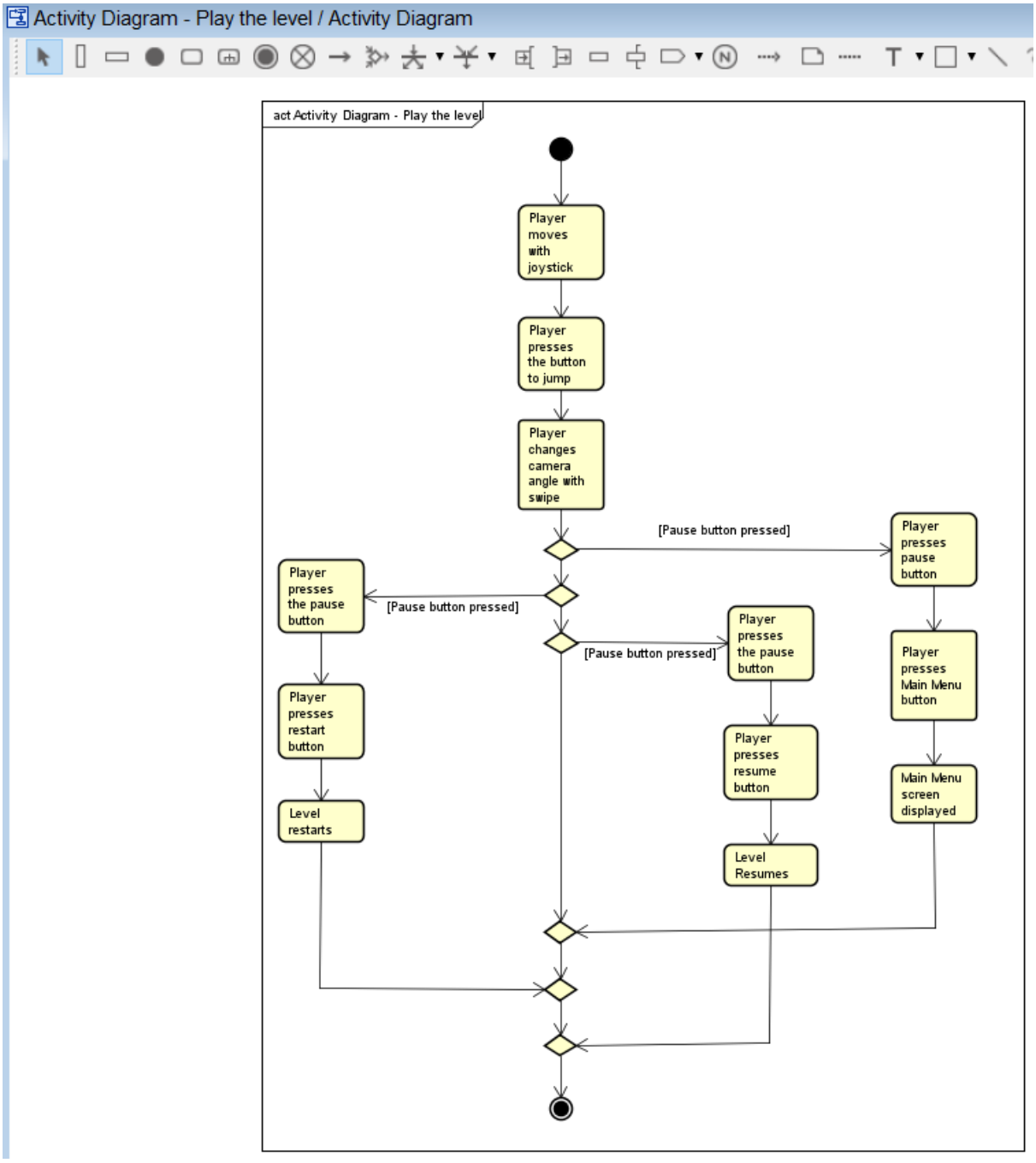
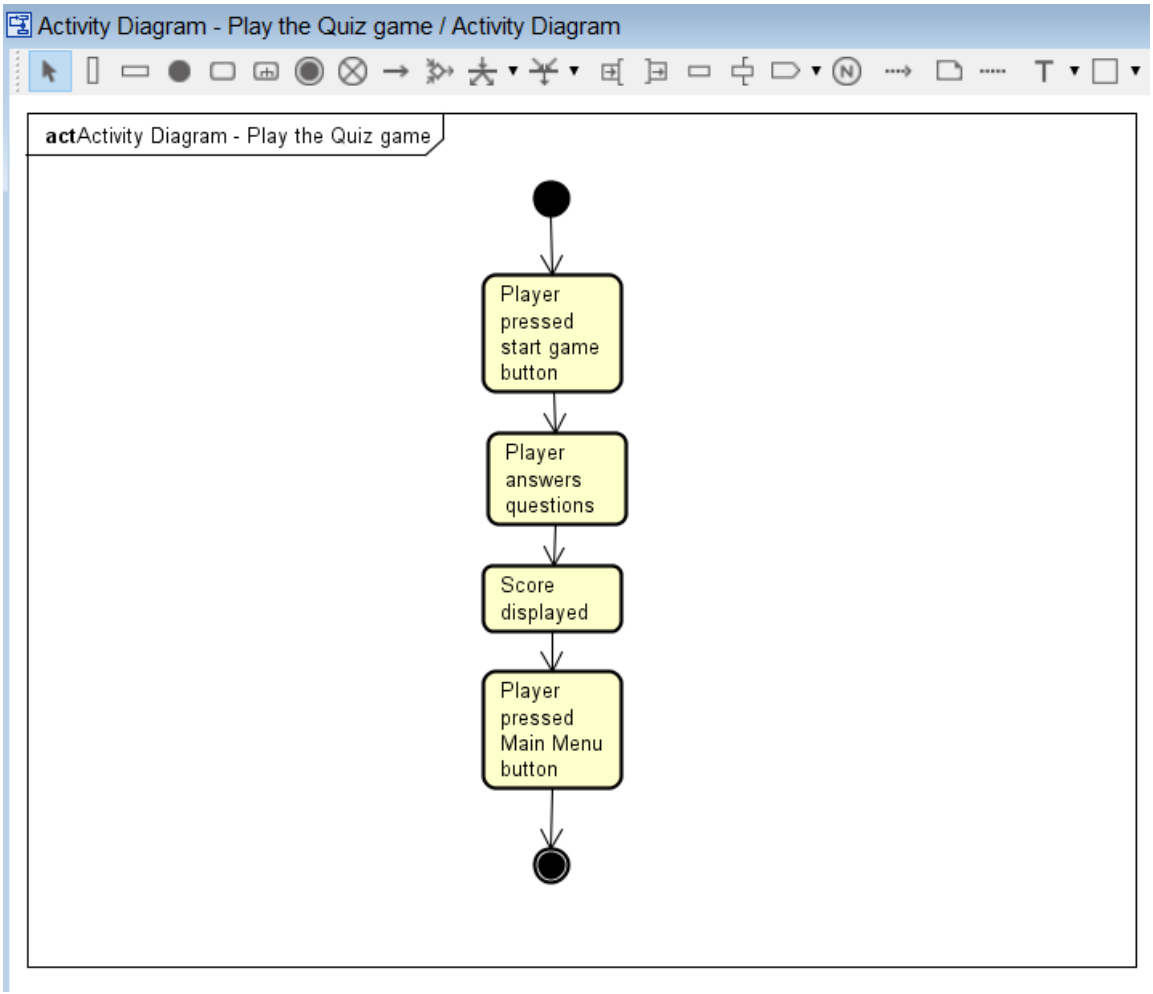

Memory palace game
Select the level
Play the level
Play the quiz game
Player

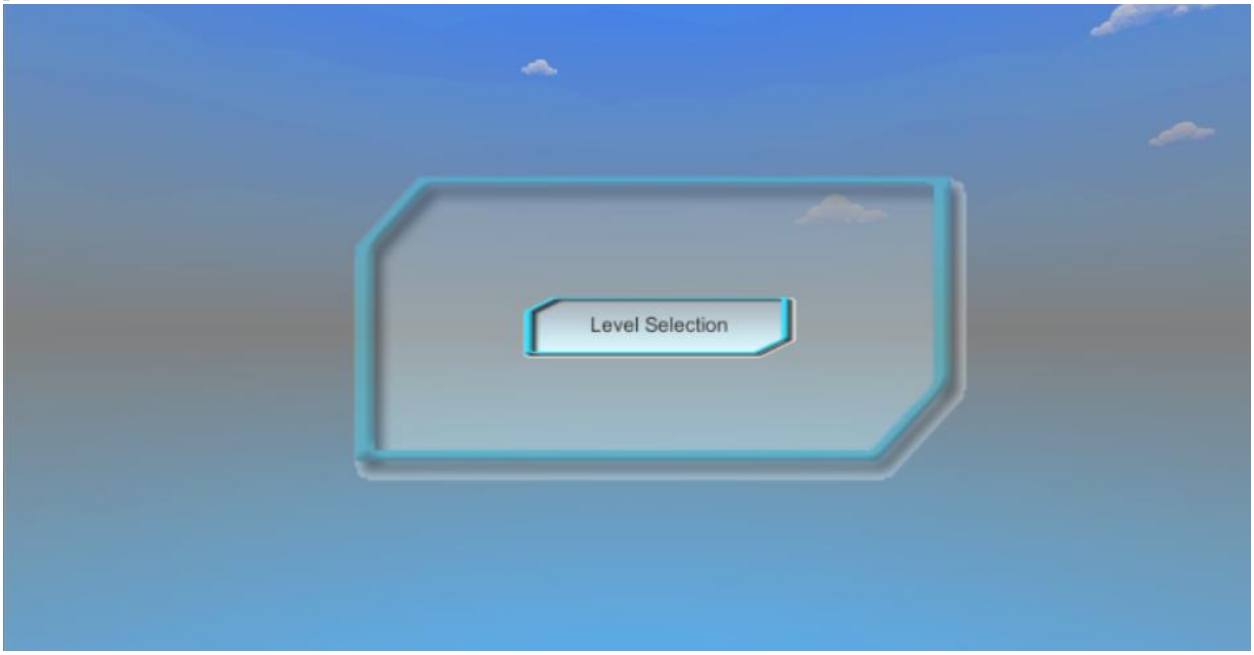
| Play the level / UseCase Description | |
|---|---|
| ITEM | VALUE |
| UseCase | Play the level |
| Summary | The player plays the level that he has perviously selected. |
| Actor | Player |
| Precondition | The player has selected the level |
| Postcondition | The screen with the Quiz Game appears. |
| Base Sequence | 1. The player can move in the game using a joystick displayed on the screen.<br>2. The player can jump by pressing the jump button.<br>3. The player can change the camera angle by swiping from left to right to ch ange the camera angle to the right, and from right to left to change the camer a engle to the left. |
| Branch Sequence | |
| Exception Sequence | Main Menu:<br>1-3 as base sequence<br>4. The player presses the pause button.<br>5. The player presses the Main Menu button.<br>6. The Main Menu screen is displayed.<br>Restart:<br>1-3 as base sequence<br>4. The player presses the pause button.<br>5. The player presses the restart button.<br>6. The level restarts.<br>Resume:<br>1-3 as base sequence<br>4. The player presses the pause button.<br>5. The player presses the resume button.<br>6. The level resumes. |
| Sub UseCase | |
| Note | |

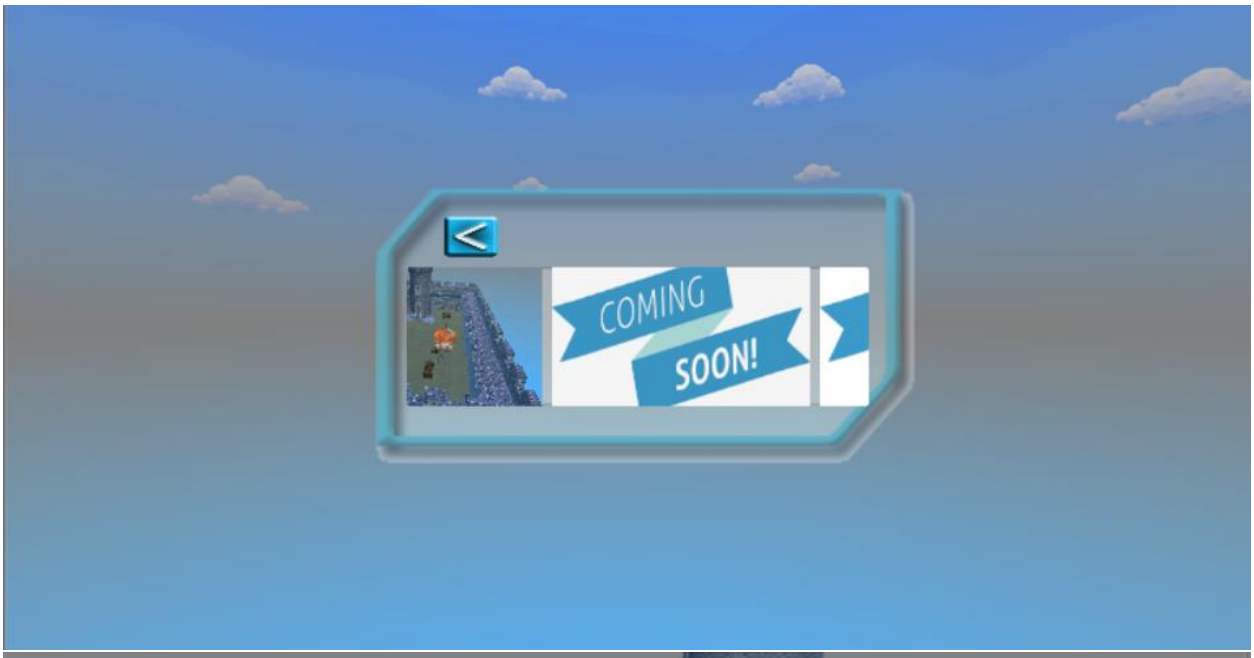
| Play the quiz game / UseCase Description | |
|---|---|
| ITEM | VALUE |
| UseCase | Play the quiz game |
| Summary | The player enters the Qiuz game section and plays the Quiz game by answeri ng the questions. |
| Actor | Player |
| Precondition | The 60 seconds in the level section have passed and the Quiz game screen is displayed |
| Postcondition | The player returns to the Main Menu and the Main Menu screen is displayed. |
| Base Sequence | 1. The player presses the button to start the game.<br>2. The player answers the questions by pressing the button coresponding to t he answer that he wihses to pick.<br>3. The player receives his score displayed on the screen<br>4. The player presses the button coresponding to the Main Menu |
| Branch Sequence | |
| Exception Sequence | |
| Sub UseCase | |
| Note | |


Activity Diagram - Select the level / Activity Diagram
actActivity Diagram - Select the level
Select level button pressed
Select level screen appears
[Back button pressed]
Main Menu display appears
Looking through the levels
Level selected


Activity Diagram - Play the level / Activity Diagram
act Activity Diagram - Play the level
Player moves with joystick
Player presses the button to jump
Player changes camera angle with swipe
[Pause button pressed]
Player presses pause button
Player presses Main Menu button
Main Menu screen displayed
[Pause button pressed]
Player presses the pause button
[Pause button pressed]
Player presses the pause button
Player presses resume button
Level Resumes
Player presses restart button
Level restarts


Activity Diagram - Play the Quiz game / Activity Diagram
actActivity Diagram - Play the Quiz game
Player pressed start game button
Player answers questions
Score displayed
Player pressed Main Menu button


Level Selection


COMING
SOON!

63
Jump
You have 60 seconds
to memorize the place

```
public void Question1Correct()
{
        playerScore = playerScore + 10;
        scoreDisplayText.text = "Score: " + playerScore;

        question1Display.SetActive(false);
        question2Display.SetActive(true);
}
```

```
public void ReturnToMenu()
{
    if (playerScore > highScore)
    {
        highScore = playerScore;
        highScoreText.text = highScore.ToString();

        PlayerPrefs.SetInt("score", highScore);
    }

    SceneManager.LoadScene("MainMenu");
}
```

```
void Start()
{
    playerScore = 0;

    highScore = PlayerPrefs.GetInt("score");
    highScoreText.text = "" + highScore;

}
```

```
public void ToMenu()
{
    Time.timeScale = 1;
    SceneManager.LoadScene("MainMenu");
}
```

```
0 references
public void RestartLevel()
{
    Time.timeScale = 1;
    SceneManager.LoadScene(SceneManager.GetActiveScene().name);
}
```

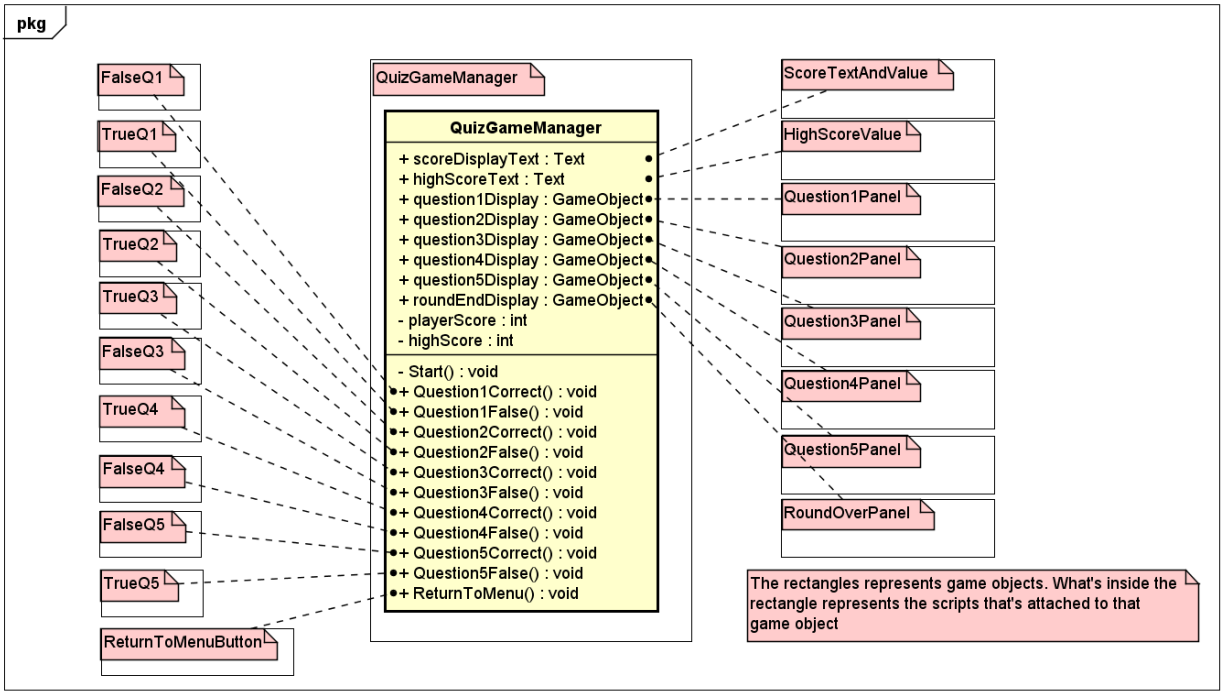
```
0 references
private void Update()
{
    timerText.text = ("" + timeLeft);

    if (timeLeft <= 0)
    {
        StopCoroutine("LoseTime");
        timerText.text = ("Time is up!");


        SceneManager.LoadScene("quizMenu");
    }
}
```

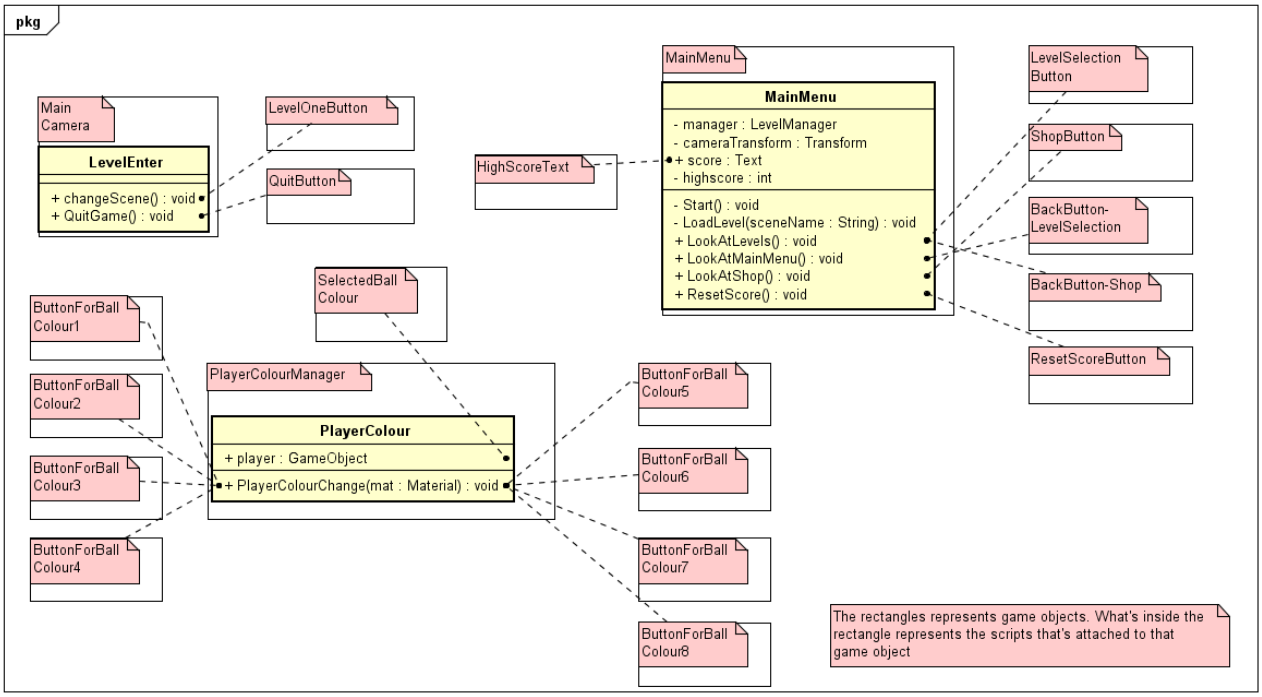
```
0 references
private void Update()
{
    timerText.text = ("" + timeLeft);

    if (timeLeft <= 0)
    {
        StopCoroutine("LoseTime");
        timerText.text = ("Time is up!");


        SceneManager.LoadScene("quizMenu");
    }
}
```

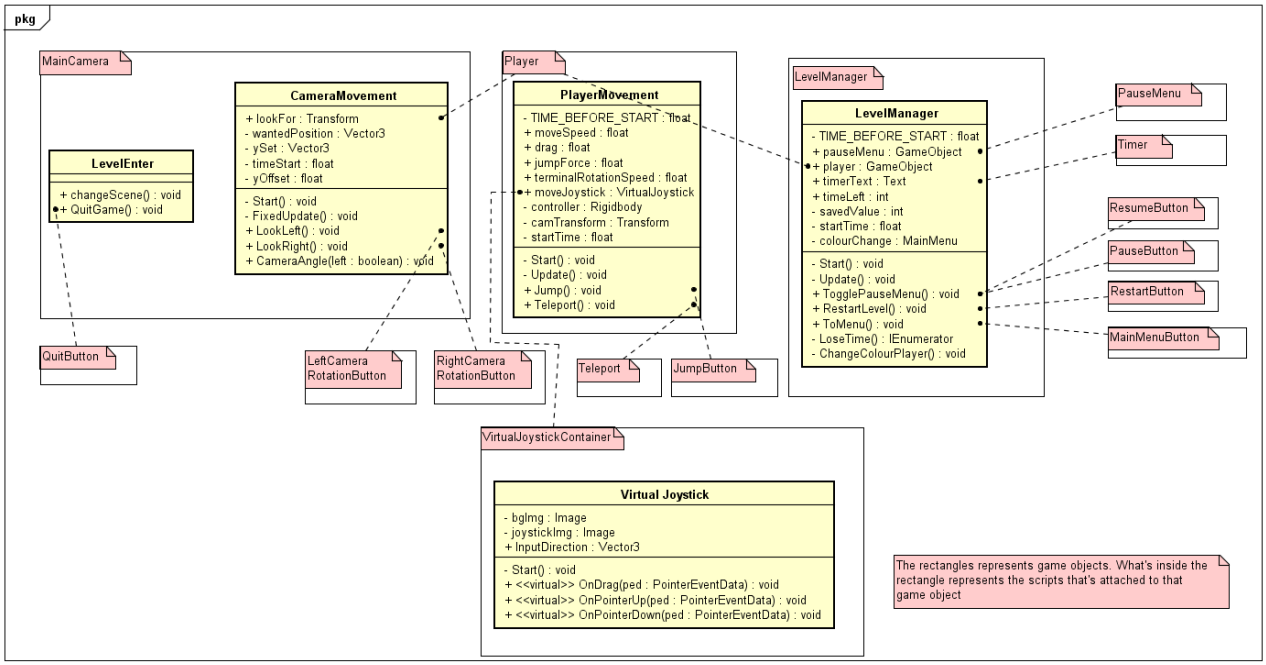
```
1 reference
private void ChangeColorPlayer()
{
    if (savedValue == 6)
    {
        Material material1 = Resources.Load<Material>("Materials/mat_6");

        player.GetComponent<Renderer>().material = material1;
    }

   else if (savedValue == 1)
    {
        Material material1 = Resources.Load<Material>("Materials/Ground_Texture_Material");

        player.GetComponent<Renderer>().material = material1;
    }

    else if (savedValue == 2)
    {
        Material material1 = Resources.Load<Material>("Materials/House_color");

        player.GetComponent<Renderer>().material = material1;
    }
```

```
0 references
IEnumerator LoseTime()
{
    while (true)
    {
        if (timeLeft > 60)
        {
            yield return new WaitForSeconds(5);
            timeLeft = timeLeft - 5;
        }
        else
        {
            yield return new WaitForSeconds(1);
            timeLeft--;
        }


    }
}
```

```
0 references
private void Start()
{
    pauseMenu.SetActive(false);
    startTime = Time.time;
    savedValue = PlayerPrefs.GetInt("ballColor");

    StartCoroutine("LoseTime");

    ChangeColorPlayer();
}
```


pkg
FalseQ1
TrueQ1
FalseQ2
TrueQ2
TrueQ3
FalseQ3
TrueQ4
FalseQ4
FalseQ5
TrueQ5
ReturnToMenuButton
QuizGameManager
QuizGameManager
+ scoreDisplayText : Text
+ highScoreText : Text
+ question1Display : GameObject
+ question2Display : GameObject
+ question3Display : GameObject
+ question4Display : GameObject
+ question5Display : GameObject
+ roundEndDisplay : GameObject
- playerScore : int
- highScore : int
- Start() : void
+ Question1Correct() : void
+ Question1False() : void
+ Question2Correct() : void
+ Question2False() : void
+ Question3Correct() : void
+ Question3False() : void
+ Question4Correct() : void
+ Question4False() : void
+ Question5Correct() : void
+ Question5False() : void
+ ReturnToMenu() : void
ScoreTextAndValue
HighScoreValue
Question1Panel
Question2Panel
Question3Panel
Question4Panel
Question5Panel
RoundOverPanel
The rectangles represents game objects. What's inside the rectangle represents the scripts that's attached to that game object


pkg
Main Camera
LevelEnter
+ changeScene() : void
+ QuitGame() : void
LevelOneButton
QuitButton
HighScoreText
MainMenu
MainMenu
- manager : LevelManager
- cameraTransform : Transform
+ score : Text
- highscore : int
- Start() : void
- LoadLevel(sceneName : String) : void
+ LookAtLevels() : void
+ LookAtMainMenu() : void
+ LookAtShop() : void
+ ResetScore() : void
LevelSelection Button
ShopButton
BackButton-LevelSelection
BackButton-Shop
ResetScoreButton
SelectedBall Colour
ButtonForBall Colour1
ButtonForBall Colour2
ButtonForBall Colour3
ButtonForBall Colour4
PlayerColourManager
PlayerColour
+ player : GameObject
+ PlayerColourChange(mat : Material) : void
ButtonForBall Colour5
ButtonForBall Colour6
ButtonForBall Colour7
ButtonForBall Colour8
The rectangles represents game objects. What's inside the rectangle represents the scripts that's attached to that game object




## Select the level / UseCase Description

| ITEM | VALUE |
|---|---|
| UseCase | Select the level |
| Summary | User selects the level that he will play. |
| Actor | Player |
| Precondition | The game is loaded and it's in the main menu. |
| Postcondition | The player enters the level. |
| Base Sequence | 1. The player presses the button for selecting the level<br>2. The screen for selecting the level appears.<br>3. The player scrolls through the levels using a finger to drag the screen on a horizontal line from left to right and from roght to left on the area where the l evels are displayed<br>4. The player selects the level by tapping once on the level that he wishes to p lay. |
| Branch Sequence | |
| Exception Sequence | Main menu:<br>1-2 as base sequence.<br>3. The player presses the back button and the screen with the main menu app ears. |
| Sub UseCase | |
| Note | |