

Mage Games Case Study

In this study, you are expected to create a simple, mobile game called “Trivia Game”. Implementation should be done in Unity and C#

Reference video: <https://www.youtube.com/watch?v=0Yk-XHyPm-0>

GUI

- At first, the user should see the play and leaderboard button on the main menu screen.
- When the player clicks the leaderboard button, it should make an API call to an url.
- Url: localhost:8080/leaderboard?page=0
- When the API call is completed, a popup should be shown which includes a list of players with score & rank.
- The list is implemented with pagination.
- The leaderboard popup should have a close button.
- *Optional: Popup show and close animations.*
- Leaderboard json data:
https://magegamessite.web.app/case1/leaderboard_page_0.json
- https://magegamessite.web.app/case1/leaderboard_page_1.json



Gameplay

- When the player clicks the play button at the main scene, the game scene should be opened.
- You are expected to implement a trivia game such as “Trivia Quest” above.
- There are 10 questions in the example JSON file. You can download the JSON file from here. <https://magegamessite.web.app/case1/questions.json>
- Each question has four choices.
- Points: Correct: 10 | Wrong: -5 | Does not reply in time: -3 (Configurable)
- The player has 20 seconds to answer.
- The game should show the “correct/wrong answer” after the player selects an option. If it is correct the option is colored with green, otherwise it will be red. Then, the game should continue to the next question with reset the remaining time and update the user score.
- You can display these transitions with fancy animations. For example, when the player answered correctly points may increase with incremental animation.

Notes

- It is important to implement the game with object oriented principles with relevant design patterns and it is a major evaluation criteria.
You should check these before the implementation:
<https://gameprogrammingpatterns.com/contents.html>
<https://unity.com/how-to/unity-ui-optimization-tips>
- You can use any tween library for animations like dotween etc. library for character movement.
- The game should run all portrait screen ratios properly.
- You are free to use any kind of asset.