# Game Manager with Resolver – Home

Assignment

## General

Need to build a spring boot web application that will allow users to create quiz games and play.

Each game has multiple users which are playing against each other. The goal is accumulating points by answering questions correctly. A user (player) can play in several games simultaneously.

The correct answer will be calculated based on all the players answers.

### Guidelines

1. An emphasis will be placed on clean code
2. Your code should be compiled, and all tests should pass at all times
3. Correct answer will be calculated by the logic defined under “Resolver” section
4. The application should try to resolve questions as soon as possible.
5. Notice the bonus parts (Resolver could be left as an interface without implementing it)
6. The games are not predefined with the players.
7. Don’t use a database.

### Goals:

* Working Spring Boot web application
* No bugs.
* Expose APIs according to the application APIs section.
  + Expand pojos as you see fit
* A user will not get the same question again if has already answered it

## Terms:

### Question:

Each question has a few possible answers.

### Gaining points:

By answering a question correctly, the user will earn X points.

### Game:

Each game have players competing against each other

### Leaderboard:

For each game there will be a leaderboard that represent the current rank state of the players based on the points they earned.

## Resolver:

### Interface:

A resolver is the component that set the status of the question. It will try to determine the right answer (from the given options in a question) using the answers provided by the players.

Possible results (Question status):

* Resolved (need to mark the correct answer)
* Pending (Not determine yet, require more answers)
* Unresolved (question can not be resolved)

If question is unresolved, we don’t award points and stop providing this question to players.

Implementation (majority vote) – *bonus part*

The correct answer for a question will determine based on the players vote. If an answer got more than 75% of the users, it will be determined as the correct answer.

For a question there should be a minimum of 6 players that answered the question in order to try and resolve the question.

There should also be a limit for the max player we collect an answer. If the question wasn’t resolved with 11 users, it will be marked as “unresolved”.

## Application APIs:

### Get Question

request:

* User name
* Game id Response:
* Question
  + Question text
  + Possible answers

### Answer Question

Request:

* User name
* Game id
* Answer Response:
* Answer status
* Points earned

### Get Leaderboard of a game

Request:

* Game id

### Create a game

This API should enable to create a new game.

## Assumptions:

* For simplicity, the application has no persistence layer. No need for database.
* Currently application is not running on a cluster.
* Build a fake bucket of questions in memory. Nothing fancy.