Software Engineering Large Practical Proposal

by Liquan Chai s1226443

Overview

Animes are japanese-stype animations. It is loved by many youth around the world. During years of evolution, more people get involved in this 'anime world'. However, there are too many animes to watch, and it is hard to measure how much do people know. This application, which called 'Anime quiz!', aim to help people demonstrate their knowledge about anime and find new areas of 'the world' to entertain. The game is based on interaction between application and user. Application, on one hand, asks anime-topic-related questions. Player, on the other hand, answer these questions. This web game tests users' knowledge about anime culture, raising their confidence and interests. The ranking competition encourages potential funs, who wants to know anime but knows little, to turn into more hard core fans. Users can learn from these quizzes while entertain from challenging themselves. Moreover, playing this game also enables user to find people who have same interests by leaving comments on the discussion page.

Goals

For user:

1. User can identify how much he / she does know.
2. User can discover other topics that he / she does not know.
3. User can learn more about anime culture in general, and making friends with same interests. (Via comments)

For implementation:

In order to build this web application, the project can be split into 4 categories:

1. Develop a quiz set management sub-system (quiz set is uploaded by admin, quizzes are accessed by control system during playing, store quizzes, quiz-evaluation)
2. Develop a user management sub-system (login/out, user page display, store user information, store / modify comments, user ranking calumniation and regular ranking updates)
3. Develop a quiz-graphic-interface
4. Develop a control system of above 3 system.

Requirement specification and use cases

1. Actor involved: Administrator, user.
2. Use cases:

User visit the website and the website asking either user to login or register a new account at the main page. User may register given a user name and a password (for simplicity, no email account required). A user may login to the individual user page by entering user name and its password. At the user page, one can browse possible sets of quizzes and select a set that he / she wish to challenge. User click 'challenge' button and start answering the set of quizzes. A quiz contain text

and images and four choices are given. For each question, a countdown timer will start. If user's selection is right, the timer stops, and it will display correct answer, and add scores to total calculate accuracy, and go on for the next question. If the user get answer wrong, the application will display the correct answer and count down accuracy, and go on for the next question. If user do not response, the application will regard user's answer as wrong, and go for the next question. In the end, the user can leave comments on the quiz set discussion page, showing their contact details.

3. Functional specifications:
    a) There should be a user system (check login details, leaving contact details, and record writing comments) (likely to be implemented in SQLlite)
    b) There should be a quizzes management system to allow administrator to upload, delete questions and show to quiz sets to user.
    c) There should be a user-based ranking system calculate ranking, and detects updates from events.
    d) There should be a control system to perform the game task (display quiz, timer count down, detect selection, identify correctness)
    e) User page and comment page interface.
    f) A ranking page list top 20 users.
4. Non-functional specifications:

It is hard to think of non-functional specifications at this stage as I am not the true costumer. (I do not know the 'speed' requirements, concurrency requirements, and how it will be deployed in the real world)

Features - the ranking system specification

The form of the quizzes will based on text and pictures and all quizzes are multiple choice questions. For example, people may be asked question or question like: 'According to the picture which ainme character is this?'

The ranking scheme is mainly based two component: accumulated total score and accumulated total accuracy. When a user finished a particular set of quizzes, the score that user obtained in that set of quizzes will be added to the total score that the user already have obtained. The accuracy is calculated by the total number of correct quizzes/ total number quizzes answered. There will be 2 rankings being displayed on user page and ranking page: One is total score, another is total accuracy. The ranking system will update in a regular period of time for re-ranking. (1 hour)
(Is this ranking system ok for large scale of users?)

Technologies and high level design

For the front end, I intend use javascript and html / CSS. javascript will provide dynamic components and CSS will provide static backgrounds. javascript also have well developed templates I can use. javascript is also a weak type language , which is suitable for front end development. In addition, there are many accessible tutorials available. However, I am not really confident if I use php as my back end. The advantage of using php is that php has many mature frameworks, and template engines (e.g. smarty, this helps me to speed up my development)

However, the disadvantage, as you taught in lectures, is a weak type language. Same as above, it also have many learning sources. Another possibility is using java EE, as I am familiar with java (but the frame work... and it is not easy to learn..., also linking to SQLlite is harder than php)

Learning schedule

As I am new to web application development, a longer time for learning (e.g. php, javascript) is expected. It might take 30 hours for learning relevant ideas & model of implementations. (15 hours html / css and javascript, 6 hours php, and other 8 hours for case-study)

Estimation & Difficulties plan of work
It is hard for me to actually list the exact detailed schedule... not sure I can implement all these features, as I mentioned above since I have to learn about 2 new languages.
A core problem is that I do not know how to implement the count down timer at the back end and also show at the front end (for php). (Any special framework allow this feature?)
Another problem could be using IDEs, I planned to use dreamweaver as my front end IDE is this suitable for this quiz application? Is there any php IDE (back end) suitable for this quiz application (for a new php learner)?

Further development

These are the features that I could think about as relevant, if I have extra time, I will implement some of these features.

1. A real-time duel system, searching for other user who are interested in answering the same set of quiz. (this could be hard, it may change the design structure)
2. A life bar, if user have so many wrong answers, he / she may fail to challenge the set.
3. Game modifiers, this will make quiz more difficult (and more fun) (e.g. half the thinking time for each quiz and / or all images are displayed in black and white)
4. A graphical statistics display page after finish a quiz set, showing how scores are added.
5. A graphical statistics display of overall user information.
6. Showing recent challenge sets on the user page.
7. User may submit their own quizzes and open for evaluation.
8. Allowing non-anime quiz sets to be uploaded and building categories classifications.