

DUNGEON CODER

Team 23 - Sprint 2 Retrospective

Ryan Teo Yii Wen	0028791734	Zhuohang Li	0028283889
Ley Yen Choo	0028729283	Devon Lee	0028067918

What went well?

In Sprint 2, we were able to establish most of the user stories we mentioned in our Sprint 2 planning document. Things were going more smoothly now since the setup of the new technologies involved in the project has been completed. The entire team is still working on learning the API of the libGDX framework, hence there is slower progress in that section of Sprint 2. The progress of server and website went pretty smoothly as we were familiar with the API at this point of the semester.

Following are the tasks that were successfully completed in Sprint 2.

- Unfinished Tasks from Sprint 1
 - We have completed several tasks that were not completed from Sprint 1.
 - We have implemented a code evaluator which will check if the code submitted by the user can be compiled and ran in the game.
 - The user is able to manage their account information such as username, password and email.
 - The user is able to assign grades for each student.
 - The user is able to leave comments for each tasks for each student.
 - The user is able to view an alert when an achievement was reached.
- As a teacher, I would like to assign tags to certain students that require more attention.
 - The user is able to set a priority tag to a certain student that require more attention.
 - The database is able to save the priority tag.
- As a teacher, I would like to see statistics on the class progress as a whole.
 - The user is able to view the overall class statistics on each task when the user clicks on a task.
 - The statistics will change accurately as the students make progress in the task.

- As a teacher, I would like to be able to control the access of stages of the game.
 - The user is able to set a lock status on a certain student to prevent the student from accessing other modes of the game.
 - The database is able to save the lock status.
- As a teacher, I would like to save some or all of my student's attempted codes.
 - The user is able to download files for the code submitted by the students.
- As a teacher, I would like to be able to create announcements for the class.
 - The user is able to add announcements on the website.
 - When the user adds the announcement, the student is able to view those announcements in the student side of the website.
 - The database is able to save all the announcements created by the user.
- Unit testing
 - Unit testing has been implemented for several tasks.
 - The unit testing will test if the implementation of several functionalities works as intended.

What did not go well?

Although we were able to implement most of the user stories, still there were a few tasks which remained unsuccessful. We did not have enough time to complete these task as some of the tasks appears to be more difficult to implement than expected. We ran into a lot of bugs and problems with implementing the features that we wanted. Most of these tasks were not completed

- As a student or player, I would like to design my own movements, skills and equipments.
 - Most of this task is completed however, there were some parts of the task that has not been completed.
 - The content of the stage has not been decided yet as we have encountered various bugs and implementation problems for the functionalities in this task.
 - The music for the gameplay has not been implemented yet as a result of the same issue stated before.
- As a teacher, I would like to be able to review my students' attempted codes.
 - The file storage system and student-task specific file names is implemented on the server side.
 - The website file retrieval functionality is not implemented yet hence, we only have a test file to test this functionality.

- As a teacher, I would like to be able to show a good example of codes after the deadline is over.
 - The user is able to view the code that is submitted by students on the website.
 - However, the functionality to retrieve code saved in the database has not been implemented yet, hence the code cannot be viewed. Currently we only have a test file to test this functionality.

How should you improve?

During Sprint 3, there are a few different ways we will try to improve individually and as a team. We realized that the current features that we are trying to implement would take significantly more effort. We would be working on these aspects to hopefully improve in Sprint 3.

- Better time management
 - Sprint 2 took a lot longer than we anticipated as we struggled steadily progress on the game client specifically. Therefore, we will be doing our best to set more time early on the weeks of Sprint 3 to work on the project to ensure that if we face any problems, we would have time to try and fix it.
- Better testing
 - In Sprint 2, we made progress on creating and running unit tests for some of the features in the game. However, there is still significant progress to be made on unit testing to make sure that the features implemented are working as intended.
- Improving team communication
 - In Sprint 2, we realized that communication became a problem as work from other classes piled up. Also, the fact that we didn't attend classes anymore made it worse. In Sprint 3, we will be working on improving our communication outside of classes and make more effort in meeting after classes.