Criterion A: Planning

# Description of Scenario

My client will be Minh Pham, a student at Inglemoor High School. Minh is highly concerned with his grades, as they have a strong effect on his future. Thus, he spends much of his time trying to calculate the minimum scores he will need on certain tests and assignments in order to obtain his desired grade. This is a very tedious task, as the process varies for each class. Different classes have different weight systems, different types of assignments, or the student may even have a different letter grade goal for each class. Minh is not alone, as many students struggle through this monotonous in order to gain insights on their grades. The product I will be developing will help streamline this process, making it extremely easy to see how a student must perform to reach certain letter grades.

The following is an excerpt from an interview I conducted with Minh, where I asked him various questions about what he would like to see in the project. At the time this interview was conducted, he had already decided he wanted a product relating to his grades in some way.

**Q: What are some of the problems you encounter with your grades?**

A: Sometimes I don’t know how my test is going to affect my grade.

**Q: What specific features are you looking for in this program?**

A: I want something where I could put in my current grades, and then figure out what I need on the next test to get an A.

**Q: What features on the current grades site would you like implemented in this new program?**

A: I’m interested in a cleaner design. I’m also interested in some sort of planner feature.

**Q: To remind you of upcoming assignments?**  
A: Yes.

**Q: What features is the current grades site lacking?**

A: I kind of want a grade calculation summary on all my classes. If I go to English, there’s no way to tell what makes up what. Like maybe give the user the ability to make categories? So, if I need to work on my essay, I can put all my essay grades under it.

# Rationale for proposed solution

I have decided to approach this problem with a Java application. I chose Java as it is compatible with a variety of operating systems. This is important to my client, who uses a variety of devices and would greatly increase convenience if he could use the product on any of his devices. I have confirmed with the client that his devices are compatible with Java applications. The IDE I will be using is NetBeans, which I chose due to my familiarity with it. In addition, NetBeans will give me the ability to design the sleek user interface that my client desires. As the product requires file input and output in order to save the user’s data, some security concerns may be raised. However, all the files will be stored on the user’s computer, so there is little concern of outside intervention.

# Success Criteria

The interview I conducted with Minh revealed the various criterion I need to accomplish with the final product for it to be considered a success. These include:

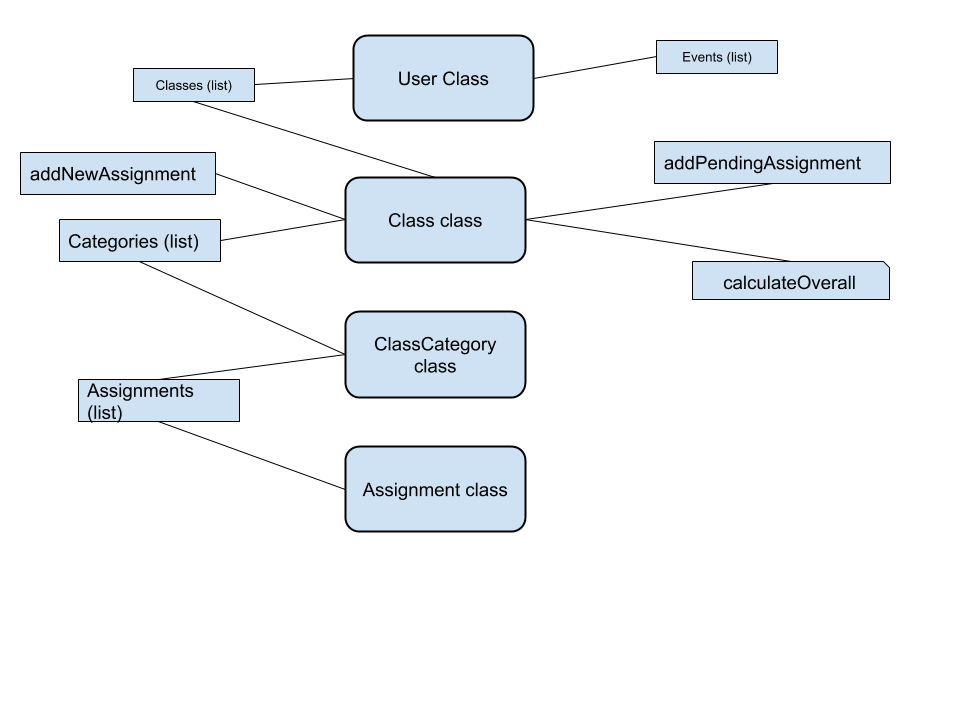
* The ability to see how assignments will affect the user’s grade
* The ability to save and store the user’s information in a file to avoid being forced to re-input data
* A calendar and planner system that will remind the user of upcoming due dates
* A clean user interface that is easy to navigate and understand

Criterion B: Solution Overview

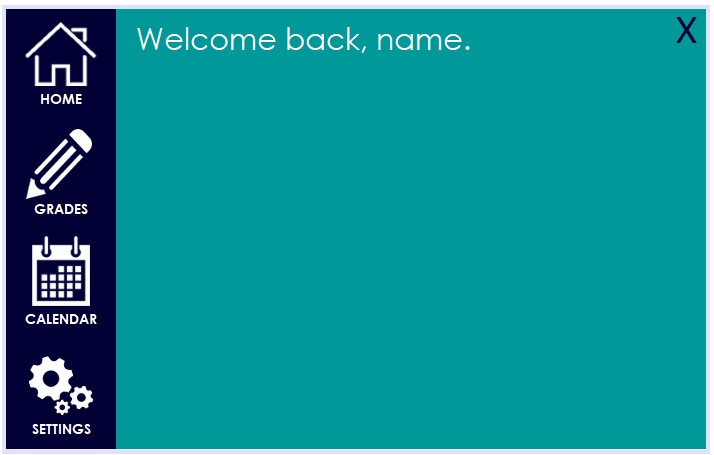
# Record of Tasks

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Action** | **Details** | **Comments/Follow Up** | **Date Completed** | **Criterion** |
| 2/11/2019 | Decided on project topic | Spoke with my client about potential project ideas. | Client is interested in a program relating to his grades in some way – calculating, planning, etc. | 2/11/2019 | A |
| 2/13/2019 | Finalized project idea, interviewed client | Interviewed my client in a more formal setting in order to gain a better idea of the product he was interested in. | I gained a good understanding of what my client wanted in the program. | 2/13/2019 | A |
| 2/14/2019 | Initial UI Creation | Messed around with various looks for the main JFrame | Created initial vision for what I want the UI to look like | 2/14/2019 | B |
| 2/16/2019 | Started login screen | Created basic layout and look for login screen | Could potentially change in the future, but I am satisfied with how it looks for now. | 2/16/2019 | C |
| 2/17/2019 | Started dashboard | Created basic layout and look for dashboard | Added main sidebar menu buttons as well as working exit button. | 2/17/2019 | C |
| 2/23/2019 | Minor changes | Started working on classes and file I/O | Initial decision is to used comma separated values to save user data | 2/23/2019 | C |
| 2/25/2019 | Started sign up page | Created basic look and layout for sign up page | May need to update later if I need more user data in constructor | 2/25/2019 | C |
| 2/25/2019 | Fixed sign up bug, worked on file i/o | Shifted from comma separated values to serialization | Also changed to storing users in an array list instead of an array | 2/25/2019 | C |
| 2/26/2019 | Dashboard buttons update | Improved the look and feel of dashboard buttons | Probably final for now. | 2/26/2019 | C |
| 2/27/2019 | Created classes for Class, ClassCategory, and Assignment | Created the main classes that are needed for the program to function. | Will work on making the program functional from now on. | 2/28/2019 | C |

Design Overviews



The following chart shows the 2 planned classes I will be creating for this project. There will almost certainly be more components added as the project proceeds with development, but these are the most essential classes, methods, and instance variables that will be required for the creation of this process.

User class objects will be created for each individual user on each device, with its information stored inside one or more text files. Specifically, the user class will contain a name instance variable, a list of events (possibly another class), as well as the classes of the user, which is the second class in this diagram. The Class class objects will have the overall grade in the class, which will be calculated with the calculate overall method, which is based on an assignments and tests list as well as the category weights list. The difference between the addNewAssignment and addPendingAssignment methods is that the naddNewAssignment will require the grade and factor it into the gradebook while addPendingAssignment will not require the user to input the grade, as the assignment will not have been completed. 

Above are screenshots of the user interface prototype I created for my client. The first is a concept I created for the login screen, where the user will select their profile by signing in. The second picture is a basic dashboard I created, which shows the four main screens of the application. I plan to have the home screen display basic user information as well as motivational messages and upcoming events. The grades section will be the bulk of content in the application, helping the user manage their grades. The calendar section will give the user a fuller planning experience that the home screen can provide, and the settings page will allow the user to adjust various features of the application.

In order to test the application, I will run a combination of tests with my client as well as tests run by myself. This will depend on the scale of the test. I will likely speak with and run tests with my client for features that are larger and more complicated, while I will test smaller features on my own. To test the sign in process, I have created a variety of accounts with varying usernames and passwords. I tested the different accounts to verify that the class serialization and deserialization would work for any account created. I also allowed my client to create his own test account to make sure that the entire process was satisfactory.