KIT305

Assignment4

Wai Yeung Chan
476435

Contents

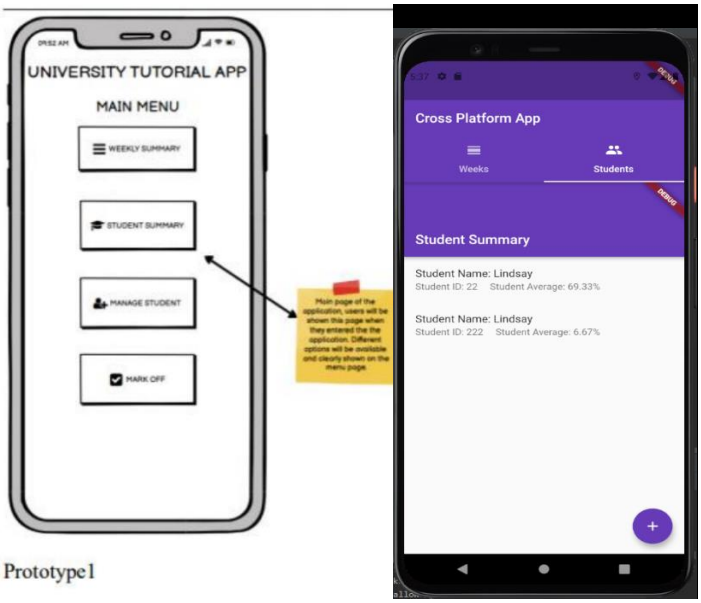
Introduction

A tutorial is one of the most important elements in university life. It helps us to make use of the knowledge we learned in the lecture and put them into practice. University encourages and monitors the attendance of the tutorial so the students will be able to consolidate their knowledge. In the ICT courses, tutorials include hands-on practical on the programming application, discussion among the students and small group consultation with the tutor. Tutorial attendance and marking are not united among the tutors, some of them are just using handwriting on a paper and some of them would have an excel recording all the information. Sometimes the tutors may even lose or forgot their attendance sheet or marking sheet that causes many troubles for the schools and students. To have a better working environment for the tutors and allow the school to record the marks that students attained in a tutorial, a united method of recording marks and tutorial attendance is necessary to reduce the possible mistake and potential problems happen in the tutorial. Nowadays there is a variety of mobile producers in the market, most of the users are using the android and IOS system. The tutor in the school is nearly not possible to be using the same operating system on their phone, the best choice for developing a tutorial marking system is by using a cross-platform system language as it is not possible for the school to provide each tutor with a mobile device. A cross-platform application will facilitate the need for different mobile operating system. In the assignment 4 criteria, we are using the flutter as the cross-platform application development tool to develop the application. Other than android and IOS system, they can also be used on the web so to give the tutors a more flexible option.

Differences to Assignment 1 Prototype

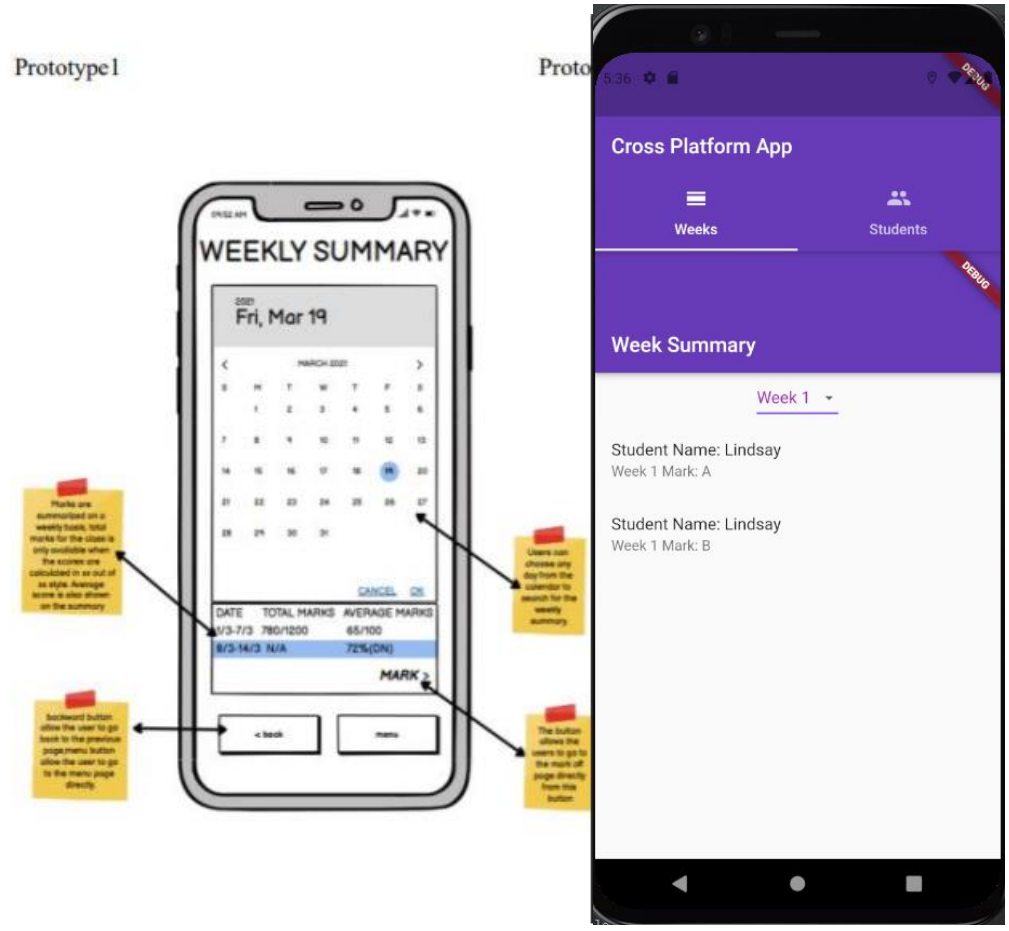
In the assignment 1 prototype, the interface of the application was divided into four, which is a weekly summary, student summary, manage student and mark-off.

The main menu page from prototype 1 is now replaced with the tap bar to increase the efficiency of the program. There are two tabs available including the student and the week which implies the student summary page and the week summary page. Each of the tab items is labelled with an icon and text to indicate the corresponding page for the user.



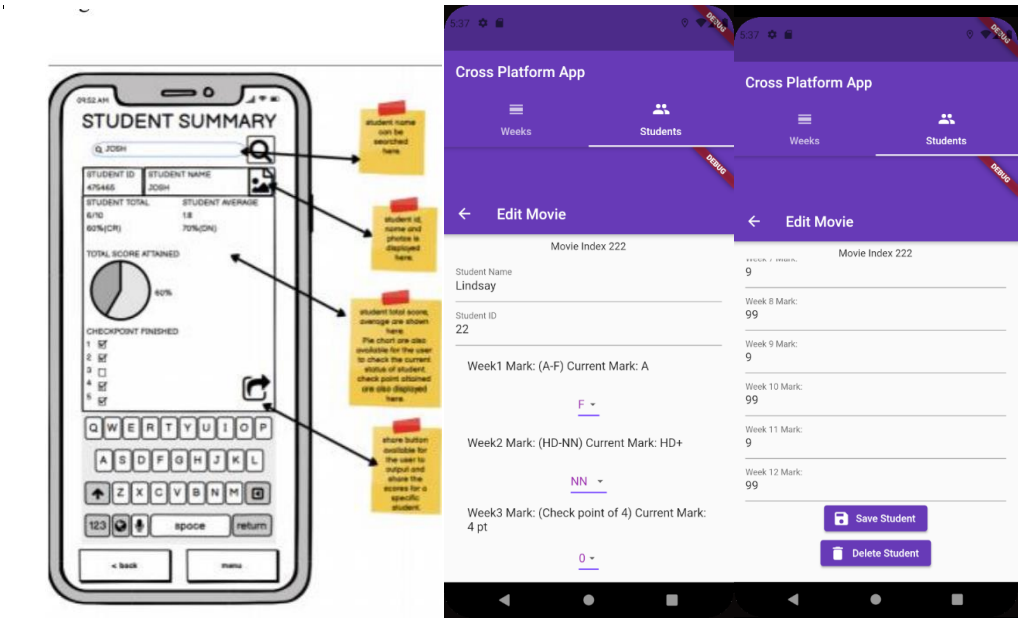
Prototype1

The weekly summary has changed from a calendar picker to a table list working with the spinner function in a flutter to select the week detail the user want and display the student's mark and the weekly average of that week. When the user selected the week the list will update immediately to show the students in the class and their mark for the corresponding week.

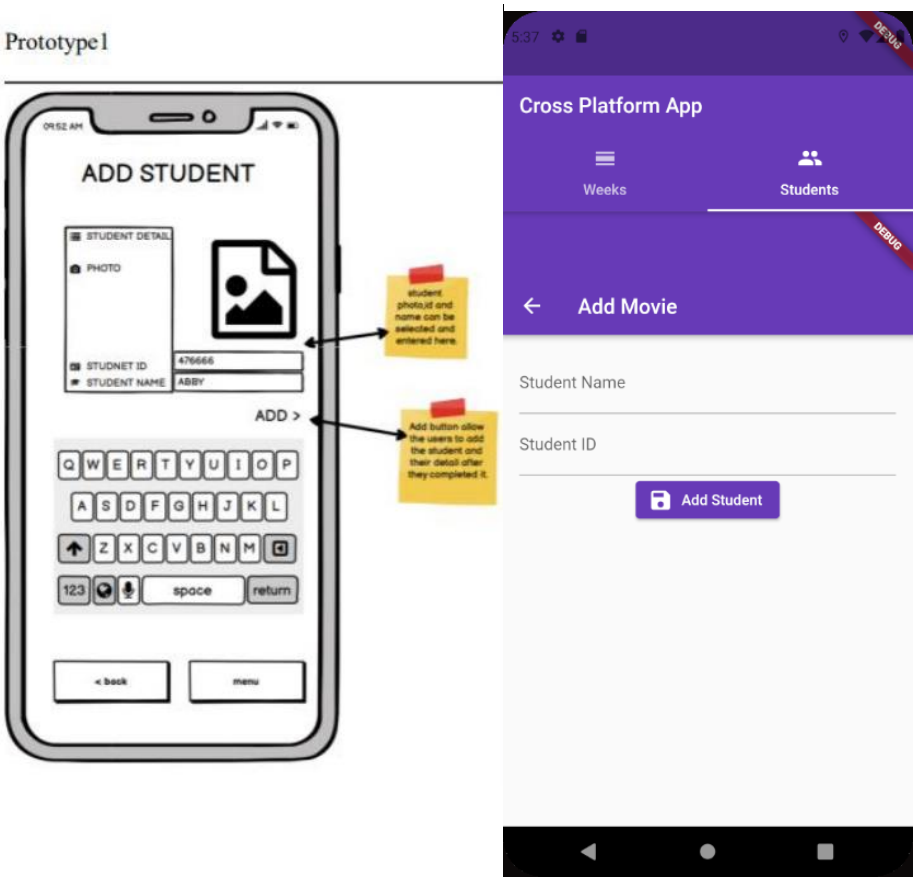


The student detail page is very similar to the prototype 1 page. They include the edition of student id and student name. The only difference is it is now combined with the weekly

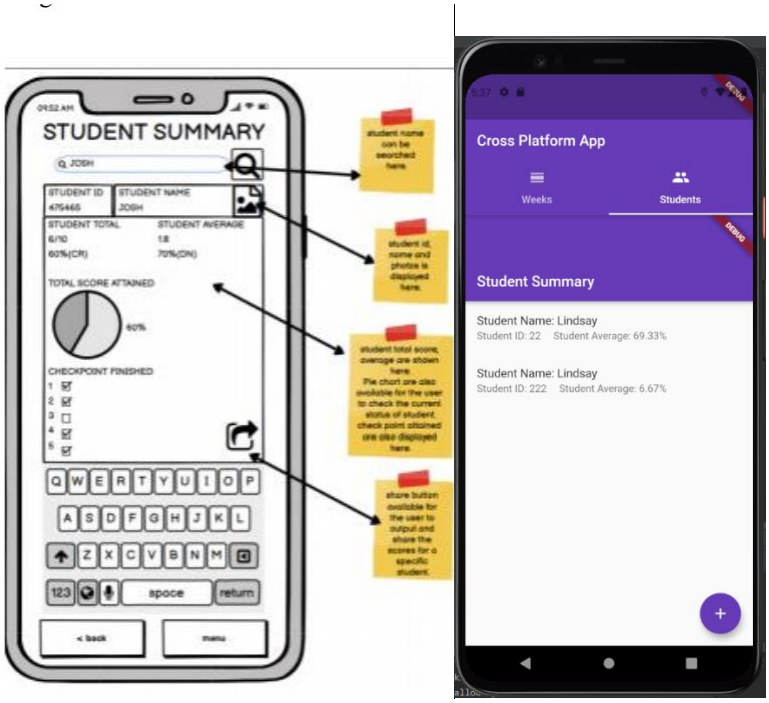
Markoff page to increase efficiency and reduce the need for directing to a different page. The save button and delete button is also available on the student detail page. There are 5 marking scheme in total including Grade marking A-Z, HD-NN, checkpoints, attendance and mark out of 100 for the tutors.



The add student page is the same as the prototype one. Users are required to type the student's name, student id to add the student. An add student button is placed under the student detail after the tutor has entered the student detail for adding the student.



The student summary page is showing the summary of the student, including their student id, student name and the student average score throughout the semester. User can also tap on the name of the student to enter the corresponding student detail page to see and edit more details about the student.



For the Mark-Off page, it is combined with the student detail page like prototype 3 on the IOS system to increase the efficiency of using the application.

Usability Goals and Design Principles Self-Critique

According to the Kit305 week5 lecture, there are 6 key aspects to a usable system which is called usability goal. They are Learnable on first use, memorable on repeat uses, efficient, failure-resistant, forgiving and satisfying. Prototype 3 aims to provide a working and usable application ready for the tutor to use. All the aspect needs to be fulfilled to provide a usable application to the user.

To make the system learnable on first use, the options available for the users are displayed with the description with text to show the users all the possible actions. For example, save, delete, and add student functions. It allows the user to understand the function of each button without testing. All the buttons are provided with icons that allow the user to have a clearer and precise understanding of the purpose of the buttons.

To make the system memorable for repeat uses, the function of the application is now combined into a small amount of storyboard so that they can remember it easily. We have combined the student summary and week summary into one page using the tab function like the prototype 3 assignment on the IOS system. The marking function of the application is also combined with the student detail page to reduce the number of view and redirection needed so it is easily memorable. Operations are visible, clear and simple to allow the users to remember them easily.

To make the system efficient, we have achieved the first two aspects of a usable system which is learnable on first use and memorable on repeat uses. With the two achievements,

the system has reduced the operation required from the user and the simpler operation required. We have also reduced the use of redirection and combined the functions into the pages to increase efficiency. Instead of memorizing the corresponding week marking scheme, other than the mark out of 100, the other marking scheme is using the spinner for marking to allow the mark to enter quickly.

To make the system failure-resistant, the backward button is shown on each page other than the main tab page so the user can go back to the previous page easily. The use of a spinner ensures the marks to be entered correctly. For the mark out of 100 and the student id editing, we have restricted the keyboard to a number pad to reduce the possibility of an error occurring.

To make the system forgiving, failure-resistant is used to prevent the user from improper use. Also, every single page of the system is providing a cancel or back button except the main page so that the user can go back to the previous page whenever they want so to prevent improper use.

For Don Norman's Design Principles, there are also six main aspects for designing an application, including visibility, feedback, affordance, mapping, constraints, and consistency. To improve the usability of the application, the application is designed to achieve most of the principles.

To improve visibility, the system is designed with all the buttons shown with description so that the potential operation will be displayed to the user to avoid misunderstanding. The two main pages which is the weekly summary and student summary page are displayed with the tab function on top to make the options more visible. The font colour and background colour of the application is different to make the detail of the application more visible.

To improve the feedback performance, when the user changes the selected week in week summary, an update will automatically appear with the student list so the user will know what action has been done. When the user added a student or deleted the student from the student list, the list will update immediately to the user to provide feedback.

To improve affordance, we made our function the same as the perceived image of an object. For example, add student button is provided with an add button on the student summary list with a “+” icon, which implies adding. The save button has got a floppy disk icon on it and the delete student button has got a rubbish bin icon on it, these perceived images of an application will improve the affordance.

To improve the constraints, the marking scheme from week 1 to week 4 which is the grading and attendance field on the student detail page is using the spinner for the user to choose the score to prevent typo. The student id filed and marked out of 100 fields is using a number pad instead of a normal keyboard to prevent the user to enter syntax other than numbers.

To improve the consistency of the program, the layout and style of the whole program are all the same with a purple background and white font. All the data from our database is the same and displayed the same to make sure the persistence of the database data and consistency of the program.

Code Documentation

The above cross-platform application is built and designed for the large screen device and tested with android pixel 4. It is also tested for use of web application. However, it should be also available for IOS devices but not yet been tested. The application mainly consists of 4 classes for the whole program to run smoothly.

1: main.dart , this class contains the student summary class, the main class, the week summary class and also the other classes for the use of the above three class. The main class contains a tab function with the student summary and week summary class. Users can also access the student detail page and student add page from this class.

2: movie.dart, this class contains the database structure for the use of the google firestore firebase. It contains the field required and specifies their data type so that it assists the whole program to run smoothly.

3: movie_detail.dart, this class provides the function of updating and deleting database function to the database. It shows the student details including their name, id and mark detail of the student.

4: StudentAdd.dart, this class provide the function of adding to the database. Users can add the student with their student's name and student id from this class.

Most of the function in the application is constructed with the code from KIT305 lecture and tutorial slides, The spinner function is constructed with the codes from the codegrepper.com. (Sable, 2020). The tab bar function in the application is produced with the aid of pub.dev.com. (fluttercandies.com, 2020)

Conclusion

In the progress of working on assignment 4, the way of interacting with firebase on flutter seems to be a bit direct than the IOS Xcode and the android studio. The flutter program is a bit more complicated as we cannot directly interact with the layout. I am proud of my achievement of making an efficient program for the use of tutorial in UTAS. However, there is more I can achieve which is sharing the function of the application and the function of changing the making scheme for a specific week, the photo function for the flutter application is also not implemented due to the huge workload and upcoming exam. All the thing I learnt from this assignment is very useful and hopefully will be used in my future career. In my future career, I hope I can make good use of my time and the other plugins available on the android, iOS and flutter platform to produce a better looking and fully functioned application for the use of the public.

Reference

Shy Sable, 2020. *spinner in flutter Code Example*. [online] Codegrepper.com. Available at: <https://www.codegrepper.com/code-examples/whatever/spinner+in+flutter> [Accessed 12 June 2021].

fluttercandies.com, 2020. *extended_tabs | Flutter Package*. [online] Dart packages. Available at: <https://pub.dev/packages/extended_tabs> [Accessed 12 June 2021].