**Criterion B: Record of Tasks**

| Task No. | Planned Action | Planned Outcome | Time Esti-mated | Target Completion Date | Crite-ria |
| --- | --- | --- | --- | --- | --- |
| 1 | Find client issue | I have a problem that can be solved using my computer science skills for a client. | 1 week | 7/5/2019 | A |
| 2 | Initial interview with client | Understand the client’s issue in detail. Learn about the client’s current system and how my solution can resolve any issues my client currently faces. | 2 hrs | 8/5/2019 | A |
| 3 | Define success criteria for product based on first interview | Decide how the product’s quality can be measured. Use the success criteria to guide development. | 2 hrs | 9/5/2019 | A |
| 4 | Go through success criteria with client and refine it. | Make sure the client agrees with the success criteria and make any final edits. | 1 hr | 11/5/2019 | A |
| 5 | Complete documentation of Criterion A | Complete first draft of planning for feedback from teacher. | 2 hrs | 13/5/2019 | A |
| 6 | Create initial screen designs | Try and think of how the final product would look. | 2 hrs | 21/5/2019 | B |
| 7 | Get client feedback on screen designs | Understand what needs to be changed and improved to make the screens more personalised to my client and closer to what they want. | 1 hr | 25/5/2019 | B |
| 8 | Create final screen designs | Update the screen designs based on client feedback. | 2 hrs | 30/5/2019 | B |
| 9 | Create structure diagram | Figure out how the program will flow. | 1 hr | 3/6/2019 | B |
| 10 | Create system and program flowcharts | Work out how the algorithms in the program would work. | 1 hr | 5/6/2019 | B |
| 11 | Create UML class diagrams | Figure out what methods and classes are required for the product to function and how all the components of the code are linked. | 1 hr | 6/6/2019 | B |
| 12 | Create a test plan | A detailed plan of testing with test data and expected outcomes is created based on the success criteria. | 2 hrs | 10/6/2019 | B |
| 13 | Experiment with Swift, Xcode, Firebase, and Alamofire | Understand the language, how to use it within the IDE, and how to implement Firebase and Alamofire protocols. | 2 months | 18/8/2019 | C |
| 14 | Plan the structure of the database | A detailed structure of the database which is denormalized and facilitates uploading receipts and querying data | 5 hrs | 21/8/2019 | B |
| 15 | Complete documentation of Criterion B | Complete first draft of solution overview for teacher feedback. | 1 hr | 27/8/2019 | B |
| 16 | Create database | Database is ready to fully use and access from program. | 1 day | 8/9/2019 | C |
| 17 | Establish database connection from program and test it. | The code can connect to and modify the contents of the Firebase database. A parent class for all the screens has been created in which there are appropriate methods which help to perform the Firebase queries. | 3 days | 22/9/2019 | C |
| 18 | Plan view controller segues | The graphical user interface should have logical transitions | 3 days | 4/10/2019 | C |
| 19 | Create login screen and authentication | Login required to start using the app | 1 day | 14/10/2019 | C |
| 20 | Complete camera view controller | User is able to choose an old photo to upload or take a photo | 2 days | 25/10/2019 | C |
| 21 | Create POST and GET functions to Tabscanner | Receipts should be uploaded and the OCR data is received on request | 2 days | 8/11/2019 | C |
| 22 | Create NetworkingClient abstract class | NetworkingClient class has static functions which can be called to handle the uploading and downloading from the API | 2 days | 14/11/2019 | C |
| 23 | Create Receipt class | Receipt objects can be instantiated with the OCR data off the receipt | 1 day | 17/11/2019 | C |
| 24 | Display OCR data on verification screen | User can view the data after uploading the receipt to Tabscanner. User can edit the data to verify the OCR conversion | 1 day | 24/11/2019 | C |
| 25 | Create expenditure view controller | The UI is easy to read and will display the total expenditure as well as the expenditure for each category | 1 day | 25/11/2019 | C |
| 26 | Create manage view controller | The user is able to view receipts from different months and categories | 2 days | 26/11/2019 | C |
| 27 | Create FirebaseHandler abstract class | FirebaseHandler has static functions that handle the uploading and downloading from the Firebase Realtime Database | 3 days | 8/12/2019 | C |
| 28 | Use FirebaseHandler static functions to upload receipts | When receipts are verified, they are uploaded to the database and the totals are recalculated | 1 day | 10/12/2019 | C |
| 29 | Use FirebaseHandler to view expenditure | Total expenditure for different months is calculated and then queried from the database and displayed to the user | 1 day | 16/12/2019 | C |
| 30 | Use FirebaseHandler to view receipts in different months and categories | By selecting a category and month, the user can view all the relevant receipts, and their exact data/purchases | 2 days | 23/12/2019 | C |
| 31 | In the manage receipts view controller the user can delete unwanted receipts | Using a tableview the receipts can be deleted and this change is reflected in the database as well as the totals | 1 day | 4/1/2020 | C |
| 32 | Contact client to show program for final improvements. | Make changes to program for it to suit what the client wants. | 1 day | 9/1/2020 | C |
| 33 | Outline the techniques used to create this application | Techniques explained in depth with annotated code and screenshots | 5 days | 20/1/2020 | C |
| 34 | Complete documentation of Criterion C | Complete first draft of development for feedback from the teacher. | 2 days | 24/1/2020 | C |
| 35 | Update Criterion C based on teacher feedback | Criterion C completed after feedback from first draft. | 1 day | 4/2/2020 | C |
| 36 | Complete test plan based on final product | The actual outcomes are added to the test plan which is included in an appendix for Criterion E. | 3 hr | 7/2/2020 | E |
| 37 | Final client interview to evaluate program | Client gives feedback on all the success criteria and evaluates the program and a whole. | 2 hrs | 18/2/2020 | E |
| 38 | Write an evaluation of the product based on client feedback | The product is evaluated thoroughly with client comments paraphrased and thoroughly referred to. | 2 hrs | 22/2/2020 | E |
| 39 | Write recommendations for further development | How the program can be improved is reflected upon, noted down, and justified for Criterion E. | 2 hrs | 24/2/2020 | E |
| 40 | Complete documentation of Criterion E | Complete first draft of evaluation for feedback from teacher. | 2 hrs | 27/2/2020 | E |
| 41 | Record a video showing the functionality of the product | Video shows full outcome of project and is ready for feedback from teacher. | 1 day | 4/3/2020 | D |
| 42 | Edit the video | Video is not longer than 7 minutes and all slow tasks have been sped up such as data entry. Subtitles are added to make referenced to success criteria explicit. | 4 hrs | 6/3/2020 | D |
| 43 | Update Criteria D and E based on teacher feedback | Last 2 criteria updated and finished after acting on feedback received from teacher. | 3 hrs | 7/3/2020 | D, E |
| 44 | Edit full documentation of project | Documentation is edited, error free, and completed to the best of my ability | 3 hrs | 9/3/2020 | All |
| 45 | Complete cover page for project | Computer Science Internal Assessment is complete and ready for submission | 1 hr | 10/3/2020 | All |