

AllinWallet

All-in-one Personal Finance App

Team 26 Sprint 2 Retrospective

Team Members

Junyi Zhang, Ray Chen, Rohith Sridharan, Shaan Hoda, Yufei Xu

What went well?	1
User Story #1	1
User Story #2	2
User Story #3	3
User Story #4	4
User Story #5	5
User Story #6	6
User Story #7	7
User Story #8	8
User Story #9	9
User Story #10	10
User Story #11	11
What did not go well?	12
How should you improve?	12

What went well?

In this Sprint, we implemented 11 out of 11 user stories completely and for each user story, we implemented a very good functionality to it. In addition, we fixed the things left in the previous Sprint and made the UI more user-friendly. In addition, we tested each of the modules we implemented manually and made sure they worked as we expected.

User Story #1

As a user, I would like the ability to permanently delete my account profile.

#	Description	Estimated Time	Owner
1	Add a button in the user profile page and popup a confirm warning	0.5 Hrs	Junyi
2	Implement delete user data in the database functionality	1 Hrs	Ray
3	Implement delete user authentication functionality	1 Hrs	Ray
4	Implement clean app data in the local device	1 Hrs	Rohith
5	Test delete functionality and make sure all the relative data delete successfully	1 Hr	Junyi

Completed: We implement a button in profile page to let the user to delete their account. Once the users click the button it will show a warning message to prevent user from clicking deleting their account accidentally. If the users confirm they do want to delete the account, the database will delete the account including the related data.

User Story #2

As a user, I would like the ability to report issues to the project developers.

#	Description	Estimated Time	Owner
1	Add a button for reporting issues and a page for the user to enter problem	0.5 Hrs	Shaan
2	Get the text user enter and implement functionality to send the report to the developer	2 Hrs	Yufei
3	Add a message box and implement functionality for the user to receive reply from the developer	1 Hrs	Yufei
4	Test the report/reply functionality works well	0.5 Hrs	Rohith

Completed: Once the user have bug when use our program, they can click the issue button in the menu and it will pop up a page for the user to type the problem. After that it will automatically generate an email that including the user's ID and will send this email to the developer.

User Story #3

As a user, I would like to set a picture as my profile avatar and change it if I wish.

#	Description	Estimated Time	Owner
1	Add user picture in multiple pages and add gui for the user to set/change profile picture	0.5 Hrs	Rohith
2	Implement data structure in the backend to store profile picture	1 Hrs	Yufei
3	Implement functionality in the front end to send profile picture	1 Hrs	Yufei
4	Implement functionality in GUI to show the user profile picture	1 Hrs	Junyi
5	Test set/change profile picture	0.5 Hr	Ray

Completed: The user is able to set their picture as their profile avatar and the picture can be displayed in multiple pages. The user also can change their profile picture. When the user logs off, and logs in anytime they want, the picture they set up before would not disappear. We saved the picture of every user in our storage.

User Story #4

As a user, I would like to see some money-saving tips while the app is on the loading screen.

#	Description	Estimated Time	Owner
1	Put together/come up with numerous money-saving tips where each one at a time will be displayed on the loading screen	3 Hrs	Yufei
2	Use a data structure or implement a method that randomizes these money-saving tips and ensures that different ones are displayed on the loading screen each time until all of them are ultimately seen/viewed	3 Hrs	Junyi
3	Implement functionality to store money-saving tips in the back end	3 Hrs	Shaan
4	Implement ability to display money-saving tips in the front end	4 Hrs	Ray
5	Test loading screen to see if money-saving tips appear	1 Hr	Rohith

Completed: Once the user login, we will show money saving tips on the loading page and each time the money saving tips will be different.

User Story #5

As a user, I would like to add location and time tags to my purchase.

#	Description	Estimated Time	Owner
1	Add to the GUI for editing existing purchases (adding visibility for locations/time tags)	3 Hrs	Ray
2	Implement functionality to modify user data (edit existing purchases) in the back end	3 Hrs	Shaan
3	Implement ability to add a location/time tag to existing purchases in the front end	2 Hrs	Rhith
4	Implement functionality to synchronize user data after modifying a purchase	2 Hrs	Yufei
5	Test purchase modification functionalities	1 Hr	Junyi

Completed: When the user wants to add the location where they buy the purchase, just click the location button, our program would find the user's location immediately and display the location on the GUI. Also, if the user wants to add the time tags to their purchase, just clicks the time button and selects the time they want to set. The purchase information along with the time would be stored in our database.

User Story #6

As a user, I would like to delete unwanted/unneeded purchases.

#	Description	Estimated Time	Owner
1	Build the GUI to edit existing purchases.	3 Hrs	Ray
2	Implement functionality to delete user data (remove an existing purchase) in the back end	3 Hrs	Shaan
3	Implement ability to remove an existing purchase and its data in the front end	2 Hrs	Rohith
4	Implement functionality to synchronize user data after removing a purchase	2 Hrs	Yufei
5	Test purchase removal functionalities	1 Hr	Junyi

Completed: This functionality has been completely implemented during Sprint 2. If the user would like, they are now able to delete particular purchases that they make. They can look through the purchases, and are able to click and delete the purchase/s that they wish to longer keep.

User Story #7

As a user, I would like to be able to sort the purchases in my spend analysis report by time/category/title/cost.

#	Description	Estimated Time	Owner
1	Add the sort button which can sort the purchases by time/category/title/cost	0.5 Hrs	Junyi
2	Implement functionality to sort the purchases in the back end	1 Hrs	Ray
3	Implement ability to show the sorted purchase in the front end	1 Hrs	Ray
4	Implement functionality to synchronize user data after modifying a purchase	1 Hrs	Rohith
5	Test sort purchase functionalities	1 Hr	Junyi

Completed: The functionality for the user to sort their purchases from their analysis report is available. Additionally the user is now able to sort via varying sorting options such as by, time, category, title and cost. Once the user views their spending analysis and purchases, they can click on a type of sort to see their purchases to be sorted in that specific order.

User Story #8

As a user, I would like to see what category I have spent money on the most.

#	Description	Estimated Time	Owner
1	Add the categorize by money button, once pressed will show which category I spent the most money on	0.5 Hrs	Junyi
2	Implement delete user data in the database functionality	1 Hrs	Ray
3	Implement categorization functionality by money	1.5 Hrs	Rohith
4	Test categorize by money	1 Hr	Junyi

Completed: A category activity was implemented, where when the user clicks on the “Categories” button, they are able to visualize exactly how much money they are spending in each particular category. Initially, the numbers were displayed in normal text format. To make this even more user-friendly and aesthetically-pleasing however, there is now a pie chart in this activity that displays these numbers.

User Story #9

As a user, I would like the ability to search a specific purchase by title.

#	Description	Estimated Time	Owner
1	Add the button which can search a specific purchase by title	0.5 Hrs	Rohith
2	Implement functionality to search a specific purchase by title in the back end	1 Hrs	Yufei
3	Implement ability to show the specific purchase in the front end	1 Hrs	Yufei
4	Test search a specific purchase by title functionalities	0.5 Hr	Ray

Completed: If the user clicks on the “Search” button, they are essentially able to look for a particular purchase of interest by entering in the title in the query. The result or results will populate based on the title that the user types into the search box. This feature can become very useful if the user has a significant amount of purchases stored in the application, and does not wish to/has trouble scrolling through every single one to find what they are looking for.

User Story #10

As a user, I would like to know how much money I was under my budget for a giving period of time.

#	Description	Estimated Time	Owner
1	Add the remaining budget display to the start screen/landing page	3 Hrs	Ray
2	Implement functionality to choose a weekly, monthly, or annual budget type	4 Hrs	Ray
3	Implement the budget type functionality in the back end	1 Hr	Yufei
4	Improve the GUI related to budget	3 Hrs	Shaan
5	Test the budget functionalities	1 Hr	Rohith

Completed: We have chosen to display the remaining budget in the eport screen instead of the start screen. The report screen displays the type of budgets the user has set in their profile. The functionality of setting a weekly, monthly, or annual budget is located in the profile screen. The database API in the backend correctly handles different types of budget the user has set up. The GUIs related to the budget functionalities are implemented completely and tested, and they work as expected.

User Story #11

As a user, I would like the start screen/landing page of the app to display as much relevant information (e.g. budget, income, spending, etc) as possible so that I don't have to navigate to other places for these information.

#	Description	Estimated Time	Owner
1	Implement the functionality to display personal income in the start screen/landing page	3 Hrs	Yufei
2	Implement a ListView or similar Android widget in the start screen to display user purchases	4 Hrs	Ray
3	Improve the interactions between the start screen and add purchase screen.	3 Hrs	Ray
4	General GUI improvements of the the start screen/landing page	3 Hrs	Ray
5	Test various functionalities in the start screen	1 Hr	Rohith

Completed: The start screen displays the income the user has set in their profile. The functionality of setting a monthly income is located in the profile screen. We have implemented the purchase list display in the start, report, and search screens using the RecyclerView class of the Android API. The interaction between the start screen and the add purchase screen has been improved as the user can now tap a purchase item in the start screen and enter the add purchase screen to edit it. The GUI of the start screen has been improved because we are now using the Android Material Theme in the start screen.

What did not go well?

1. For the delete account part, we only delete the data that storing in our database. We want the delete account functionality also delete the data in the user's local storage.
2. Regarding User Story #2, when the user has the ability report their feedback, we did not yet implement the ability to store that feed back into the database. Moving forward we can save the feedback so we can understand what feedback we are given.
3. For the changing profile picture, now we only submit the picture that user choose, but for this part, we want to edit the picture before the picture is submitting to our database.
4. Regarding the money-saving tips, while we successfully implemented this user story and its functionality works, we could have added even more tips. Fortunately, this should not be an issue as it is quite simple to garner these tips and store them, but we will surely consider this in the next Sprint.

How should you improve?

- We need to make the GUI look more good-looking instead of using just one or two colors which looks too monotonous.
- Cosmetics will be one of the main focuses