Final Sprint Retrospective

The following user stories were initially intended for implementation as chosen from the product backlog. They are listed below along with their respective Sprint Values, estimated durations and acceptance tests.

1. As a developer, I need to implement a feature that notifies other users that a particular user has posted an expense. Estimated duration: 2 days, Sprint Value:4
   Acceptance Tests:
   - An email is successfully sent to household members when a new expense is added.

2. As a developer, I need to implement a functionality that authenticates users and acts as a protection layer to web pages . Estimated duration: 3 days, Sprint Value:4
   Acceptance Tests:
   - Users authenticity is checked when a user logs in
   - Homepage routes have a protection layer i.e. They are to be accessed by authenticated users
   - The Web-page router is able to check if an active user is authentic to grant them access to a web-page
   - New Users are authenticated after signing up.

3. As a developer, I need to implement functionality that keeps track of sessions and authorization of users. Estimated duration: 3 days, Sprint Value:5
   Acceptance Tests:
   - User sessions are Implemented
   - A storage system for user sessions is implemented where they expire after 24 hours.
   - Cookies are assigned to sessions with a lifespan of 2 hours.
   - A User session is started when a user signs in and destroyed when a user signs out.

4. As a user, I need to be able to invite my friends to our household group via email so that we can share expenses. Estimated duration: 1 day, Sprint Value:5.
   Acceptance Tests:
   - Friends are able to receive email invitations with a group name and website link.
   - If friends are not registered users, they can sign up else they sign in.

5. As a user, I need to be able to view the household members so that I can know the friends in my group. Estimated duration: 2 days, Sprint Value:5
   Acceptance Tests:
   - Members are able to search their group using their group name
   - Members are able see the friends in their group

6. As a developer, I need to implement a functionality that splits the expenses equally so that they can be shared amongst the members. Estimated duration: 3 days, Sprint Value:5
   Acceptance Tests:
   - An expense is added on its household
   - An expense is shared equally amongst the current members of the household

- Information regarding who posted an expense is entailed on the household table

7. As a user, I need to be able to specify the group that I belong to when I post an expense, so that the expense can be posted in the correct household group. Estimated duration: 3 days, Sprint Value:4
   Acceptance Tests:
   - Users are provided with an input field to indicate their group on the expenses form when they are posting an expense.
   - The group name is captured on the expenses table stored in the database.

8. As a user, I need to be able to confirm my password when I sign up so that I can be sure of my password. Estimated duration: 1 day, Sprint Value:4
   Acceptance Tests:
   - Users are provided with an option to confirm their passwords upon signing up
   - Users are only signed up if they have confirmed passwords.

9. As a developer, I need to create an about page so that users can see a summary of the functionality offered by the web application. Estimated duration: 1 day, Sprint Value:3
   Acceptance Tests:
   - Users are able to see a summary of the functionality offered by the Splitgorithm web app upon clicking a relevant button on the welcome page.
   - Users are provided with an option to navigate back to the welcome page from the about page.

10. As a developer, I need to implement a functionality that allows a user to create a group for his/her household such that the group is stored on the database. Estimated duration: 3 days, Sprint Value:5
    Acceptance Tests:
    - A user is able to create a group for his/her household and the group is stored on the database
    - The user who created a group is labelled as a group leader
    - A table which stores all the existing groups is created and stored on a database, and a row is appended to this table whenever a new group is created
    - A user is able to join an existing group by using the group name

11. As a user, I need to manage my account so that it can stay up to date. (Epic)
12. As a user, I need to use the bill splitter app so that I can interact with other household members. (Epic)
13. As a developer, I need to implement a functionality that allows users to see what a particular member has paid for. Estimated duration: 2 days, Sprint Value:5
    Acceptance Tests:
    - Users can see expenses paid for by a particular member
14. As a user, I need to be able to delete my account so that I can be freed from bills should I decide to leave my household. Estimated duration: 2 days, Sprint Value:5.

Acceptance Tests:
- Users are able to successfully delete their accounts.
- The integrity of the database is maintained as users are deleted from all tables they exist in.

15. As a developer, I need to implement an option that allows users to see a log of all recent activity in the household. Estimated duration: 4 days, Sprint Value:5
    Acceptance Tests:
    - Users can view a log of all activity that is happening on the web app in their group.
    - The activities are described with full details of who made them and when did they make them.

16. As a developer, I need to implement a functionality that allows a user to add a display picture so that other members can view the picture. Estimated duration: 3 days, Sprint Value:2
    Acceptance Tests:
    - Users are able to add a profile picture on their profile.

17. As a user, I need to use the bill splitter app so that I can manage expenses. (Epic)
18. As a user, I need to be able to post expenses so that the website can divide them equally amongst members. (Epic)
19. As a user, I need to visit the website so that I can create an account. (Epic)
20. As a developer, I need to implement a functionality that allows users to see debts of a particular member. Estimated duration: 4 days, Sprint Value:3
    Acceptance Tests:
    - Users are able to see the debts of a particular member.
21. As a user, I need to receive notifications about all activities in the bill splitter app so that I can stay up to date. (Epic)
22. As a developer, I need to implement functionality to allow users to view debts so that they can make payments. Estimated duration: 2 days, Sprint Value:5
23. As a developer, I need to create a payment option so that users can settle their expenses. Estimated duration: 3 days, Sprint Value:5
    Acceptance Tests:
    - Users are able to settle a certain expense that they owe in their group.

24. As a user, I need to be able to see my balance in the household at any point in time. Estimated duration: 3 days, Sprint Value:5
    Acceptance Tests:
    - Users are able to see their current balance in a specific group that they are in.
    - The balance values are formatted conveniently to two decimal places.

During the final Sprint, the development team had planned to implement a lot of functionality to enhance the UI, general look and security of the web app. However, the team continued to struggle with time management, general understanding of some of the APIs used electricity issues as well as internet connectivity issues.

SQL Queries

Making SQL queries proved to be a huge problem for the team, with most of the members starting to really understand towards the end of the Sprint. This meant that, as the complexity of the queries which needed to be made increased, the progress of the team became even slower.

User sessions

The team struggled to understand the express-session API documentation, as a result the functionality achieved from using this API is very minimal. To elaborate, in the end it was used to improve the security of the website by making sure that users needed to be signed in, in order to be able to access the different pages of the website. The member assigned to users sessions related stories, struggled to understand the passport API which could have helped with tracking which user is signed in, on what page. This would have greatly enhanced the functionality of the website, as there would be no need to manually perform user authentication by asking the user to identify themselves (and the group they belong to) for every action they want to perform on the website.

Internet Issues and Electricity Issues

The different members had access to different levels of internet access. Some members were privileged enough to be able to make MS Teams calls during the day, some members could only afford to be on MS Teams after midnight. This level of internet access fluctuated throughout the project, it made it difficult to have a consistent Daily Scrum. This greatly hampered the progress of the team, as members were not always available for meetings as a result of various reasons ranging from data deficiencies to poor internet connectivity. Meetings were sometimes held via Whatsapp.

As a result of the challenges, some of the issues could not be implemented. These are listed below:
- As a developer, I need to implement a functionality that allows a user to add a display picture so that other members can view the picture
- As a user, I need to be able to delete my account so that I can be freed from bills should I decide to leave my household.

Unfortunately since there was no other Sprint, these issues could not be attempted. However, since basic functionality was achieved, this is deemed acceptable by the team.

Roles and responsibilities

Member name	Role(s)
Thabiso Msimango	Scrum Master (all 3 Sprints),member of development team, documentation
Hlanganani Jiyane	Member of development team, documentation

Lungisani Luvuno	Member of development team, documentation
Moshekwa Malatji	Member of development team, documentation

Positives
- The web application has basic functionality.
- Nodemailer is being used for consistent email notifications.
- The web app uses persistent data storage (SQL).

Problems
- The web application uses manual user authentication for every action the user wants to perform.
- The team was unable to successfully use passport in combination with user sessions to deduce which user is signed in on each page.
- The team struggled with using the SQL database, making it difficult to carry out some stories.

Improvements
- The functionality of the web app can be greatly improved through automatic user authentication on all the pages of the web app.
- The styling of the pages could also be better.
- Front-end pages can be tested using pertinent frameworks such as Puppeteer and Cyprus.

Sprints Summary

Sprint	Duration	Velocity
First	**7 days**	**21/33**
Second	**9 days**	**26/45**
Final	**8 days**	**80/87**
Overall		**127/165**

Final Remarks

The Sprint velocity of the team fluctuated throughout the project, going from a relatively high value to a low value in the second Sprint as a result of the inclusion of numerous new APIs. The velocity then went back up in the last Sprint as the team started having a better understanding of the different APIs being used. Communication initially lacked as a consistent Daily Scrum could not be established. However, this improved from the second Sprint going into the last one. Documentation was done consistently throughout the project.