CSCI 5708 Mobile Computing

Project Plan: *SmartShopper*

Feb. 18th, 2020

Group 16

Amoli, Nishant B00835717

Mandava, Abhinav B00841453

Ponangi, Sriram B00851421

Shah, Moni B00830791

Tarpara, Dhruv B00856253

Team Organization

The development of the SmartShopper application will be done by following the steps in the software development life cycle and we are planning in such a way that each member of our team will get opportunities to contribute in every stage. However, we are also assigning specific roles to each member of the team; for which they will have to take full ownership/responsibility. These are the roles we are considering at this moment:

1. **Leader:** Moni will be the leader and she will be responsible to finalize/approve the functionalities, deliverables and setting up deadlines. She will also be responsible for resolving the conflict of opinions among the team members.
2. **Code Maintainers:** Dhruv and Abhinav will be responsible for maintaining code in the GitLab, merging pull requests, resolving merge conflicts and ensuring the build status is always successful in the repository.
3. **Scrum Master:** Sriram will be responsible to ensure the agile development process is followed and also to maintain the sprint and user story dashboards in the AzureDevops tool.
4. **Quality Analyst:** Nishant will take care of the testing and reporting of the bugs found to ensure the quality of the application. For this he will perform unit testing, integration testing and UI testing.

Development Process

We will be following the agile scrum process while developing the application and for that we will be using the following terms in this context:

1. **Epic:** represents a business initiative to be accomplished and in our case we are considering having two epics where the first one is focused on minimum functionalities to develop a Minimum Viable Product (MVP) and the second on the remaining functionalities*[1]*.
2. **Feature:** represents a shippable component of software*[2]*. In our app, every screen will be considered as a feature and features are part of an epic.
3. **Product Backlog Item (Story or Bug):** represents a work item that can be used to track the progress on work*[2]*. Each feature is subdivided into stories which will be assigned to a member of the team to work on.
4. **Sprint:** represents a repeatable fixed time-box during which product backlog items are completed *[2]*. Before the beginning of each sprint we will add some stories from the product backlog items list and assign them to members of the team who will commit to complete them before the end of that sprint.

Also, we will be following the planning, designing, implementation, testing and documentation phases while working on each feature rather than following the traditional waterfall model where we proceed to a next phase only if we finish the previous phase for all the features. Therefore, having milestones after each phase in the software development lifecycle is not feasible for us so instead we are planning our milestones by features per sprint.

Schedule

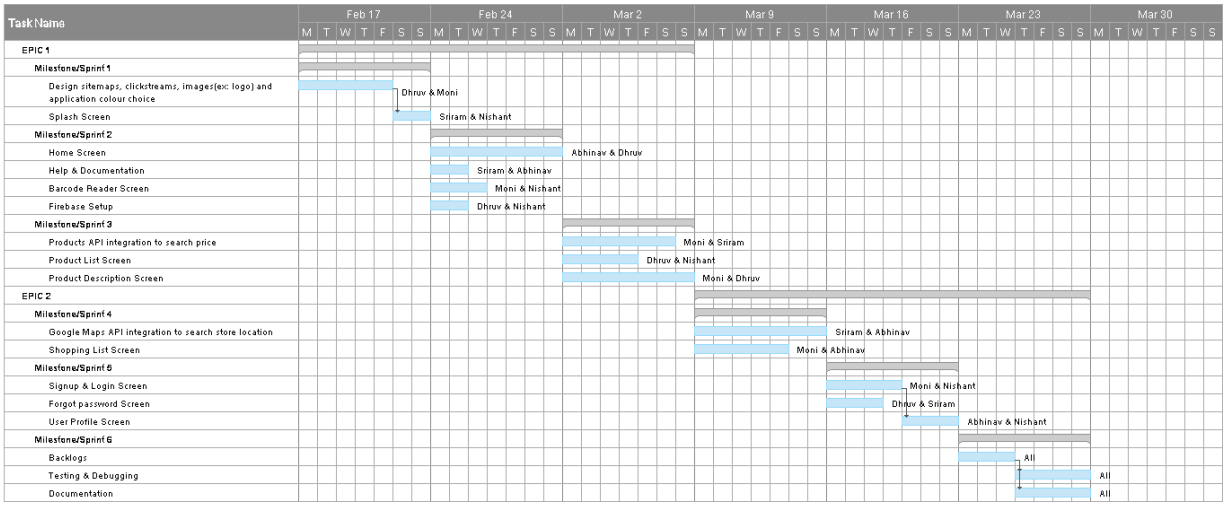

Figure-1: Gantt Chart for schedule(developed using Smartsheet*[3]*)

Features and Work Assignment

1. **Sitemaps, clickstreams and images(eg: logo) design, and application colour:** Assigned to Dhruv and Moni.
2. **Splash Screen:** will contain the logo of the app. Assigned to Sriram and Nishant.
3. **Home Screen:** is the screen that the user sees after the splash screen. Assigned to Abhinav and Dhruv.
4. **Help and Documentation Screen:** Assigned to Sriram and Abhinav.
5. **Barcode Reader Screen:** is where the users scan the barcode using the camera. Assigned to Moni and Nishant.
6. **Firebase Setup:** Assigned to Dhruv and Nishant.
7. **Products API integration to search price:** Search for APIs exposed by the stores and use them. Assigned to Moni and Sriram.
8. **Products List by Name Screen:** shows the list of price variations of the given product at different stores after search. Assigned to Dhruv and Nishant.
9. **Product Description Screen**: shows the details on the selected item. Assigned to Moni and Dhruv.
10. **Nearest stores using Google Maps API**: invokes Google Maps API to show the nearest stores. Assigned to Sriram and Abhinav.
11. **Shopping List Screen:** is a user's personalized shopping list. Assigned to Abhinav and Moni.
12. **Signup/Login Screen:** prompts the users to either signup/login or continue in guest mode. Assigned to Moni and Nishant.
13. **Forgot Password Screen:** helps users to recover their password by entering email. Assigned to Dhruv and Sriram.

14. **User Profile Screen:** is to view user account information and do the required settings. Assigned to Abhinav and Nishant.

Process Breakdown

The above features are grouped into two epics in our process. Each feature has a set of design, development and testing stories. We are considering three sprints per epic. Each sprint is a week long therefore we will have a total of 6 sprints starting from 17th February, 2020 to 29th March, 2020. The stories in epic-1 are to be completed in three sprints and the same goes for the stories in epic-2. The breakdown of the two epics and their features are shown in the table below.

Epic-1

Feature	Stories
Design sitemaps, clickstreams, images(ex: logo) and application colour choice.	1. Design sitemaps 2. Design clickstreams 3. Design logo and images
Splash Screen	1. Design the wireframe 2. Develop the screen 3. Test the functionality
Home Screen	1. Design the wireframe 2. Develop the screen 3. Design the navigation graph

Epic-2

Feature	Stories
Implement Google Maps API Screen	1. Design the wireframe 2. Develop the screen to show user's current location using GPS 3. Implement functionality to open Google Maps if the user clicks on the "Get Directions" button 4. Test the functionality
Shopping List Screen	1. Design wireframe

	4. Design and develop a custom Recycler View and populate the data 5. Test the functionality
Help and Documentation Screen	1. Design wireframe 2. Develop the screen 3. Create Help and Q&A 4. Test the functionality
Barcode Reader Screen	1. Design the wireframe 2. Develop the screen 3. Implement logic to fetch data from the barcode 4. Test the functionality
Setup Firebase	1. Create account and integrate with the application

	2. Develop the screen to store or retrieve data in local storage 3. Implement shopping list operations for list items in Firebase database 4. Test the Functionality
Signup/Login Screen	1. Design wireframe 2. Develop the screen 3. Setup Firebase Database 4. Implement the user registration/login. 5. Test the functionality
Forgot Password	1. Design wireframe 2. Develop the screen 3. Setup mail server to send mails to user 4. Test the functionality

	2. Design the database 3. Test the functionality
Product API integration for search price	1. Research the available APIs that display product details 2. Create Service classes in application 3. Test the functionality
Products List by Name Screen (Search Results)	1. Design wireframe 2. Develop the screen 3. Implement logic to fetch data from APIs 4. Implement a feature that allows users to filter the list based on different options 5. Connect the Recyclerview adapter and data fetched from API

User Profile Screen	1. Design wireframe 2. Develop the screen 3. Implement the feature that allows users to modify password and email and save 4. Test the functionality

	6. Test the functionality	
Product Description Screen - Images, details about the product.	1. Design wireframe 2. Develop the screen 3. Implement Google Maps API to fetch user location to get directions 4. Implement Mapview to show store locations 5. Test the functionality	

Risk Identification, Analysis And Management

In the initial phase, several risks have been identified and at the end of the first epic, all the risks will be reviewed. So far we have identified the following potential risks that might hinder our progress. After analysing these risks, on the basis of severity, each risk has been provided with a priority from low to high. Along with the risk, the probability of the occurrence, severity, mitigating action and contingency plan have been mentioned in the table below.

Risk Description	**Probability of the risk**	**Severity**	**Mitigating Action**	**Contingency Plan**
Group members dropping the course or members leaving the group	LOW	HIGH	It has been decided that two members will be assigned to each task/role so that there won't be any dependency on one member. Also, at the end of the week, every	If a member decides to drop the course, another member who was working on the same task/role will continue working on it. In the worst case, if both of them decide

			group member will go through the entire work and if there's any doubt, it will be broached up in the second meeting.	to drop the course, the other members will decide among themselves and take it from there.
Laptops get stolen or stop working	MEDIUM	HIGH	All the contents related to the project such as code, documentations, e.t.c will be simultaneously committed to GitLab.	Any other computer can be used to log into GitLab and the updated work can be pulled onto the new device.
Any member is sick	MEDIUM	MEDIUM	Two members will be working on the same task so that there won't be any dependency.	If a member is ill and is not able to work at all, another member who was assigned the same task can continue working on that task.
New technology - Kotlin	HIGH	MEDIUM	All the group members will be giving special emphasis on Kotlin programming language.	Use of Java where required as Kotlin has interoperability with Java.
New technology - Android Studio	HIGH	MEDIUM	Focusing on lab work will mitigate this risk to the greatest extent.	Consulting with teaching assistants or the professor.
Limited API integration - Companies like Walmart might not provide APIs to give access to their databases or even if they do, it might be very limited. Also, they might decide to stop giving access anytime	MEDIUM	HIGH	Along with finding the required APIs, a dummy database will be created.	In the worst case, if we are not able to find the required APIs, the dummy database will be used.
Contentions between the group	LOW	HIGH	Moni has been chosen as the chair and will be	Leader's decision has to be followed.

members			coordinating everyone's work and roles. She will also be making important decisions.	

Tools

Our application development process will make use of numerous project management, tracking and development tools.

1. Draw.io – for creating use case diagrams and clickstreams
2. DYNO Mapper – for designing sitemaps
3. Mockflow – for designing wireframes
4. Justinmind – for prototyping the screens
5. Android Studio – IDE for developing the application
6. GitLab – for code maintenance
7. AzureDevops – for tracking the agile process

References

[1] "Epic - Scaled Agile Framework", *Scaled Agile Framework*, 2020. [Online]. Available: https://www.scaledagileframework.com/epic/. [Accessed: 18- Feb- 2020].

[2] *Scrumguides.org*, 2020. [Online]. Available: https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf. [Accessed: 18- Feb- 2020].

[3] "Smartsheet", *App.smartsheet.com*, 2020. [Online]. Available: https://app.smartsheet.com. [Accessed: 18- Feb- 2020]