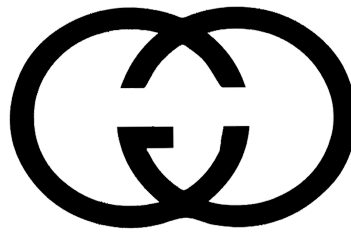


CONGO



Team 16 - Sprint 1 Planning Document

Rohith Ravindranath, Kenny Varghese, Spencer Kline, Ashwin Mahesh, Raman Kahlon

Sprint Overview

In sprint 1 we will be implementing the fixtures that will be the foundation of our website. This involves the initial setup of our environments (Firebase, Github, Frontend). It also requires that we setup our database. After we have setup the environment we will begin by creating clickable prototypes for all the pages that we will be implementing in this sprint. Once we have created the page prototypes we will start to create the basic functionality of the marketplace registering for both user and merchant accounts, login for all accounts. We will then implement more of the foundation by allowing merchants to add, edit, and remove products and also allowing users to search for those products add them to their cart and purchase them.

SCRUM Master: Rohith Ravindranath

Meeting Plan: Monday: 4:30 - 6:30 PM, Thu 4:30 - 6:30 PM

Risks and Challenges:

Some risks we will face will during this initial sprint will be about consistency and the initial developmental setup. The initial couple of days will be focused on code standards and protocol and making sure all team members will be able to deploy the website from their local computer for testing purposes. Another major challenge we will face is the consistency of the UI for this e-commerce site. We need to make sure that everyone has the same color scheme, header, and footer in order to maintain cohesiveness within the web application. Given that this will be our first instance in developing coding relating to this project, we will run into some application structure issues that will require group discussions. It is likely that we will

need many structure revisions in order for all members to be on the same track and have our code run smoothly.

Current Sprint Detail

User Story 1: As a user, I would like to like to be able to access all the necessary pages.

Task Number	Description	Time (hours)	Task Owner
1	Header/Footer CSS	1	Ashwin
2	Homepage	3	Ashwin
3	Product Page	3	Spencer
4	Cart Page	3	Rhith
5	Checkout Page	2	Rhith
6	Confirmation Page	1	Rhith
7	Search Page	3	Spencer
8	Merchant Home	3	Kenny
9	Merchant Add/Edit Product Page	2	Raman
10	Create merchant login page	2	Kenny
11	Create user login page	2	Kenny

Acceptance Criteria:

- Given that the user should see a consistent appearance of the site throughout all pages of the application, a user should see a similar color scheme, footer, header and formatting throughout all pages.
- Given that the user enters the Congo application, a user should see a homepage that shows that most popular items being sold on the site, a login button for Congo members and be able to search all the available products in the search bar.

- Given that the user searches for a product, a user should be able to see the search page with a search filter, and all of the product data split into separate pages based on how many are to be shown per page.
- Given that the user wants to see the products in their specific cart, a user should be able to see a list of products that they have selected to buy as well as the price and the estimated shipping date.
- Given that the user wants to buy the products in their specific cart, a user should be able to enter the necessary credentials to process the payment for the products they want to buy. User should also be able to select the shipping method they wish to use.
- Given that the checkout process was successful and the payment was successfully processed, the the user should be able to view their confirmation of the payment being processed as well as a receipt for all the products the user bought for that specific instance.
- Given that the user selects a product. A user should be able to see the product page with all of the product data, related products, estimated delivery date, view count, total view count, and a button to add the product to their cart.
- Given that the user logs into their merchant account, the user should be able to able to see links to their product statistics page as well as their merchant settings page.
- Given that the merchant user wants to view their products being sold, the user should be able to see a list of all their products being sold, have the capability to edit/delete and add more products.
- Given the merchant user wants to wants to access their portal, the user should be able to log in given the right credentials.
- Given the user wants to log into their Congo member portal, the user should be able to log in given the right credentials.

User Story 2: As a congo merchant, I would like to add products/tags

Task Number	Description	Time	Task Owner
1	User inputs required information for new product	2	Raman
2	Information is validated	2	Raman
3	Information is added to the database	1	Raman
4	Success message to merchant user on UI	2	Raman
5	Unit Tests - display dummy product, incorrect input, fetching from database	2	Raman

Acceptance Criteria:

- Given that the user wants to add a product on their account, the user needs to input the required information on the UI
- Given that the user inputs the information, we need to validate it and make sure it is not fake
- Given that the information is valid, we need to add this information to the database and make sure it is available to be searched by Congo users
- Given that the database responds with a success message, the merchant should receive a pop up message saying that their product is added to the database.
- Given that the unit test will test this functionality, we will make sure that we get consistent success codes when adding multiple products.

Use Case 3: As a congo merchant, I would like to edit products/tags

Task Number	Description	Time	Task Owner
1	User selects product to edit	2	Raman
2	User inputs the specific changes to product on UI	2	Raman
3	Input is updated to the database	1	Raman
4	Success message show on UI, that product information has been updated	2	Raman
5	Unit Tests - display dummy product, incorrect input, fetching from database	2	Raman

Acceptance Criteria:

- Given that the merchant wants to edit one of their products on the site, we need to get the specific product for which they would like to change.
- Given that the merchant chose a specific product, we need to know what changes the user wants to make to the product.
- Given that we have the specified changes, we use the changes as input to update the product on our database.
- Given that we get a success message from the database, we need to display a confirmation message on the UI along with the changes on the site as well.

Use Case 4: As a congo merchant, I would like to remove products/tags

Task Number	Description	Time	Task Owner
1	Separate product and the tags associated to it	2	Raman
2	Delete product from database	2	Raman
3	Send confirmation message and email about product	2	Raman
4	Unit Tests	2	Raman

Acceptance Criteria:

- Given that the merchant wants to remove one of their products from the site, we need to remove the product from the associated tags in the database
- Given that the product and tags are separate, we will need to delete the product from the database
- Given that we get a success code from the deletion of the product, we need to send a confirmation message to the merchant user.
- Given that the use case tests this functionality, we will make sure that our dummy merchants are able to successfully delete a product.

Use case 5: As an unregistered congo customer, I would like to browse via search bar.

Task Number	Description	Time	Task Owner
1	Setup frontend with the clickable prototype	1.5	Spencer
2	Fetch search data from database	2	Spencer
3	Display Products ordered correctly from database	3	Spencer
4	Setup search filters to filter the search results	2	Spencer
5	Unit Test - search valid and invalid products. Assert filters order correctly.	2	Spencer

- Given the user wants to search for a product via the search, allow the user to enter in a product to search for in a text box and on submit will give the frontend that search.
- Given the user searches for a product via the search, the backend will send a query to the database and return all products with matching name and tag.
- Given the user searches for a product via the search, all products with matching name and tag will be displayed originally being sorted based on rating from highest to lowest.
- Given the user selects a search filter, the products on the search page will be sorted based on that selected filter.
- Given the user searching for a product via the search is functional, as a developer we should be able to test the search input, and sorting and have all unit tests pass.

Use Case 6: As a user, I would like to see a product page

Task Number	Description	Time	Task Owner
1	Fetching product data from database	2.5	Spencer
2	Display data from database	2.5	Spencer

Acceptance Criteria:

- Given the user selects a valid product, then that product's name, description, merchant name, and images should be fetched from the database and displayed on the product page.
- Given the user selects a valid product, that product's related products should be fetched from the database and displayed on the product page.
- Given the user selects a valid product, that product's estimated delivery time, current views, and total views should be fetched from the database and displayed on the product page.

Use Case 7: As a user, I would like to register for a Congo account

Task Number	Description	Time	Task Owner
1	Get user input	1	Kenny
2	Validate user input	1	Kenny
3	Encrypt password	1	Kenny
4	Add to database	1	Kenny

Acceptance Criteria:

- Given that a user wants to become a Congo member on this site, it's imperative that the user inputs the correct information to be stored
- Given that the user inputs the information, we need to validate that input and make sure it meets all the requirements
- Given that the user input is validated, it is important that we encode the password that way their sensitive information is stored and not accessible by anyone else
- Given that all the tasks are completed successfully, we now need to store all this information in the firebase realtime database for future use.

Use Case 8: As a user, I would like to get licensed for a merchant account

Task Number	Description	Time	Task Owner
1	Get user input	1	Rohith
2	Validate user input	1	Rohith
3	Encrypt password and merchant license	1	Rohith
4	Admins approve of merchant registration on admin portal	2	Rohith
5	Add to database	1	Rohith

Acceptance Criteria:

- Given that a user wants to become a merchant on this site, it's imperative that the user inputs the correct information to be stored
- Given that the user inputs the information, we need to validate that input and make sure it meets all the requirements
- Given that the user input is validated, it is important that we encode the password and the merchant license that way their sensitive information is stored and not accessible by anyone else
- Given that the all the information is correct and encrypted, it is now the admins job to either either accept or decline the registration of the new merchant, this can be done on the admin portal page.
- Given that all the tasks are completed successfully, we now need to store all this information in the firebase realtime database for future use.

Use Case 9: As a user, I would like to login into my Congo account

Task Number	Description	Time	Task Owner
1	Validate login	1	Kenny
2	Store valid login in session	0.5	Kenny
3	Unit tests - incorrect password, attempt limits	1	Kenny

Acceptance Criteria:

- Given that the user wants to log into their Congo member account, the user needs to input their login credentials and have it be validated.
- Given that the user login credentials have been validated, we need to store this instance of the time session in the Firebase Realtime Database for future reference.
- Given that the unit test makes sure this is working functionally, we will test this use case using different dummy Congo accounts

Use Case 10: As a user, I would like to login into my merchant account

Task Number	Description	Time	Task Owner
1	Validate login	1	Rhith
2	Store merchant login in session	0.5	Rhith
3	Unit tests - incorrect password, attempt limits	1	Rhith

Acceptance Criteria:

- Given that the user wants to log into their merchant member account, the user needs to input their login credentials and have it be validated.
- Given that the user login credentials have been validated, we need to store this instance of the time session in the Firebase Realtime Database for future reference.
- Given that the unit test makes sure this is working functionally, we will test this use case using different dummy Congo accounts

Use Case 11: As a user, I would like to add products to my cart

Task Number	Description	Time	Task Owner
1	Product is added to the user's cart within database	2	Rohith
2	Products in cart appear in Cart Page by calling the database	2	Rohith
3	Unit Test - Test to see if products are correctly in the database and extracting it	2	Rohith

Acceptance Criteria:

- Given that user should be able to view their cart on different computers, we need to add the product within the user's cart within the database, this will allow the user to see their cart on any computer
- Given that the user wants to view their cart, the products in the user's cart should be pulled from the database and populate the Cart Page.
- Given that a unit test has been created to test this functionality, the user's cart should be seen on multiple computers and on the product page,

Use Case 12: As a user, I would like a checkout my products in my cart.

Task Number	Description	Time	Task Owner
1	Getting user shipping information from form	1	Ashwin
2	Validating Information	1	Ashwin
3	Unit Test	1	Ashwin

Acceptance Criteria:

- Given that a user is ready to enter their information, they should be able to do so easily
- Given that a user has inputted their shipping information, it should be validated to check that all inputs are acceptable
- Given that the user input is valid, it should be stored in session to pass to the payment page
- Given that the shipping appears functional , run unit tests to ensure it is fully functional

Use Case 13: As a user, I would like to complete the payment necessary to the products in my cart.

Task Number	Description	Time	Task Owner
1	Get payment info from user	0.5	Ashwin
2	Send info to Stripe	1	Ashwin
3	If valid, complete and record transaction in database	1	Ashwin
4	Else, report error to user	0.5	Ashwin
5	Unit Test	0.5	Ashwin

Acceptance Criteria:

- Given that the user is ready to pay, they should be able to easily and clearly input their payment info
- Given that the user has inputted their card information, it should be successfully sent to Stripe with the charge amount
- Given that the transaction was successful, report to the user that their order was successfully placed, and create a new Order entity in the database
- Given that the transaction was unsuccessful, report to the user what the error was without reloading the page, eg. Invalid card number, Insufficient Funds
- Given that the handling of payment appears functional, run unit tests to ensure it is actually fully functionally

User Case 14: As a developer, I would like to set up a developer workflow

Task Number	Description	Time	Task Owner
1	All members clone git	2.5 hours (0.5 hour per person)	Ashwin, Rohith, Spencer, Raman, Kenny
2	Each member makes separate branch	2.5 hours (0.5 hour per person)	Ashwin, Rohith, Spencer, Raman, Kenny
3	Commit to branches with explanation of the changes	2.5 hours (0.5 hour per person)	Ashwin, Rohith, Spencer, Raman, Kenny
4	Merge meeting to fix bugs	5 hours (1 hour per person)	Ashwin, Rohith, Spencer, Raman, Kenny

Acceptance Criteria:

- Given that we all need the same source code, all of the team members must clone the initial repository
- Given that we want to avoid conflicts with merging, every member will make their own branch
- Given that it is important to know what has changed with each commit, each member will give a clear explanation in their commit messages
- Given that merging can be a difficult process, we will meet in person to collaborate and corroborate on this process.

User Case 15: As a developer, I would like to create meaningful documentation

Task Number	Description	Time	Task Owner
1	Weekly code reviews and coding standards	10 hours (2 hours per person)	Ashwin, Rohith, Spencer, Raman, Kenny
2	Thorough documentation	10 hours (2 hours per person)	Ashwin, Rohith, Spencer, Raman, Kenny
3	Create planning documents for the next week	10 hours (2 hours per person)	Ashwin, Rohith, Spencer, Raman, Kenny

Acceptance Criteria:

- Given that all team members will be coding and committing their changes, the team members should have weekly meetings to ensure all code is up to standard and easily readable.
- Given that all team members will be using each others' code, it is necessary that all files, functions, and global variables are documented and have a thorough explanation to make it easier for other members to understand.
- Given that all team members complete their given tasks for this sprint, it is crucial that we are thinking forward and investigate the tasks and use cases that needs to be complete for the next sprint.

Remaining Product Backlog

Unregistered Congo Customer

- ~~1. As an unregistered congo customer , I would like to register for a Congo account.~~
- ~~2. As an unregistered congo customer , I would like to browse the products.~~
- ~~3. As an unregistered congo customer , I would like to browse products by category.~~

Registered Congo Customer

- ~~4. As a congo customer member, I would like to have all the abilities that an unregistered congo customer have.~~
- ~~5. As a congo customer member , I would like to login to my account~~
- ~~6. As a congo customer member , I would like to add a product to my shopping cart.~~
- ~~7. As a congo customer member , I would like to remove an item/s in my shopping cart.~~
8. As a congo customer member , I would like leave reviews on products I have purchased.
9. As a congo customer member , I would like to view my recommended products.
- ~~10. As a congo customer member , I would like to purchase products in my cart with Stripe API.~~
- ~~11. As a congo customer member , I would like to refund products with Stripe API.~~
- ~~12. As a congo customer member , I would like to view my items in my shopping cart.~~
13. As a congo customer member , I would like to view my purchase history.
14. As a congo customer member , I would like to view how many products are remaining on a product I am viewing.
15. As a congo customer member , I would like to view how many people are viewing this product.
16. As a congo customer member, I would like to purchase a Congo gift card.
17. As a congo customer member , I would like to redeem a Congo gift card.
- ~~18. As a congo customer member , I would like to receive email confirmation after a successful order placement.~~
- ~~19. As a congo customer member , I would like to save my payment information.~~
- ~~20. As a congo customer member , I would like to save my shipping information.~~
- ~~21. As a congo customer member , I would like to save my billing information.~~
- ~~22. As a congo customer member , I would like to reset my account password through my email.~~

Congo Stream Member

- 23. As a congo customer Stream member, I would like all the same features as a regular member
- 24. As a congo customer Stream member , I would like to receive expedited shipping
- 25. As a congo customer Stream member , I would like to receive discounts on select products.

Congo Merchant

- ~~26. As a congo merchant, I would like to be verified.~~
- ~~27. As a congo merchant, I would like to add products~~
- ~~28. As a congo merchant, I would like to edit products~~
- ~~29. As a congo merchant, I would like to remove products~~
- ~~30. As a congo merchant, I would like to be able to login in to a merchant page~~
- 31. As a congo merchant, I would like an merchant page with my statistics
- ~~32. As a congo merchant, I would like to be able to add tags to my products~~
- ~~33. As a congo merchant, I would like to be able to remove tags to my products~~
- 34. As a congo merchant, I would like to pay to promote my products

Congo Admin

- 35. As an congo admin, I would like to approve merchants
- 36. As an congo admin, I would like to reject merchants
- 37. As an congo admin, I would like to approve products
- 38. As an congo admin, I would like to reject products
- 39. As a congo admin, I would like to be able to login to a site admin page.
- 40. As an congo admin, I would like to see overall site statistics