MARKETPLACE BUSSINESS PLAN DAY 6

Deployment Preparation and Staging Environment Variables

Overview

The **Foodtuck** website is a fully functional online food platform where users can browse and order a wide variety of delicious food items. The menu includes categories like **meals, burgers, sandwiches,** and **top-selling dishes**. This platform has been carefully designed to offer a seamless and secure online ordering experience, ensuring customer trust and satisfaction.Over the course of six days, the project transitioned from the initial brainstorming of ideas to successfully setting up a staging environment in preparation for deployment. Each day focused on specific tasks that played a vital role in building the platform, from conceptualization to implementing essential features and security measures. The end goal is to provide a robust and user-friendly online shopping experience, aligning with industry standards.

Day 1

Laying the Foundation of My Marketplace Journey

- In first step I selected my market place which is Q-Commerce.
- Defined my business goals and write all my business requirement in note book.
- Created data schema like schema of product, order, customer , shipment, payment etc…
- I've also defined the relationship between entities.

Conclusion: So finally I've successfully implemented my day 1 hackathon document.

Day 2

Planning the Technical Foundation

- In day 2 I've focused on technical requirement which is related to my hackathon.

- **Frontend Requirement:**

  ○ User-friendly interface for browsing data. ○ Responsive devices for mobile and desktop users. ○ Created essential pages like home, product listing, product details, cart checkout etc…

- **Sanity CMS as backend:**

  ○ Used sanity CMS to manage product data, customer details etc, sanity act as database for my marketplace.

- **Third-party API's:**

○ Integrate API's for shipment , tracking, payment gateways etc…

I've also created design system architecture. So in day 2 I've focused on frontend , backend and third party API's when we know how to do these all we can create a website.

Day 3

API Integration and Data Migration

In day 3 I've focused on API integration and data migration in my hackathon, by using provided API I've used that provided API in my hackathon.

In provided API's I've data for products clothing and I used that API's and display the products on the frontend.

Day 4

Building Dynamic Frontend Components for my Marketplace

In day 4 I focused on building dynamic frontend component and fetch data from sanity CMS.

- Implement reusable and modular component.
- Applied state management techniques.
- Responsiveness design and UI/UX best practice.
- Prepare real-world client side.

Like product listing, product details, category, search bar, tracking, shipment, gateways etc.

Day 5

Testing, Error Handling and Integration Refinement

In day 5 I've focused on testing, error handling and integration refinement

- **Functional Testing:** In functional testing I verified workflows like product listing, product details, add to cart checkout etc…
- **Performance Testing:** Use light house to analyze load time and responsiveness.
- **Security Testing:** Validate input fields, secure API keys and HTTP implementation.
- **CSV Based Testing Report:**

Test Case ID	Test Case Description	Test Steps	Expected Result	Actual Result	status	Severity Level	Assigned To	Remarks
TC001	Verify user login functionality	1.Open login page 2.Enter valid credentials 3.Click login button	User successfully logs in	User successfully logs in	Passed	Low	Ameen Alam	No Issues found
TC002	Validate login with incorrect credentials	1.Open login page 2.Enter invalid credentials 3.Click login button	Error message displayed	Error message displayed	Passed	Low	Ameen Alam	Functioning as expected
TC003	Check login button disabled for empty fields	1.Open login page 2.Leave fields Empty 3.Verify button state	Login button is disabled	Login button is disabled	Passed	Low	Ameen Alam	No issues found
TC004	Verify user session timeout	1.Login to the system 2.Stay idle for 15 mins 3.Try accessing a secure page	User is redirected to login page	User is redirected to login page	Passed	Medium	Ameen Alam	Session handling working fine

TC005	Test product search	1.Open search bar	Relevant products displayed	No products displayed	Failed	High	Ameen Alam	Search API integrati on issue
	functionality	2.Enter product name 3.Click search button						
TC006	Validate secure data transmission	1.Perform login 2. Monitor network traffic	Data is encrypted	Data is encrypted	Passed	High	Ameen Alam	SSL encrypti on working fine
TC007	Test backend integration for cart	1.Add item to cart 2.Check cart count in API response	Correct count displayed	Incorrect count displayed	Failed	High	Ameen Alam	Cart backend synchronization
TC008	Verify error handling for network failure	1.Simulate network failure 2.Perform an action requiring network	Error message displayed	Error message displayed	Passed	Medium	Ameen Alam	Error message are userfriendly
TC009	Validate responsive design on mobile	1.Open site on mobile browser 2.Check UI layout	All elements align correctly	Alignment issues in footer	Failed	Medium	Ameen Alam	Footer needs responsi ve fixes
TC010	Check loading time optimization	1.Measure page load time using tool. 2. Compare with target time.	Page loads within 2 seconds	Page loads within 1.8 seconds	Passed	Low	Ameen Alam	Performance Optimization

○ **CSV Content:**

```
Test Case ID,Test Case Description,Test Steps,Expected Result,Actual
Result,Status,Severity Level,Assigned To,Remarks TC001,Verify user login functionality,1. Open
login page 2. Enter valid credentials 3. Click login button,User successfully logs in,User
successfully logs in,Passed,Low,Bameen Alam,No issues found. TC002,Validate login with
incorrect credentials,1. Open login page 2. Enter invalid credentials 3. Click login button,Error
message displayed,Error message displayed,Passed,Low,Bameen Alam,Functioning as expected.
TC003,Check login button disabled for empty fields,1. Open login page 2. Leave fields empty 3.
Verify button state,Login button is disabled,Login button is disabled,Passed,Low,Bameen
Alam,No issues found. TC004,Verify user session timeout,1. Login to the system 2. Stay idle for
15 mins 3. Try accessing a secure page,User is redirected to login page,User is redirected to
login page,Passed,Medium,Bameen Alam,Session handling working fine. TC005,Test product
search functionality,1. Open search bar 2. Enter product name 3. Click search button,Relevant
products displayed,No products displayed,Failed,High,Bameen Alam,Search API integration
issue. TC006,Validate secure data transmission,1. Perform login 2. Monitor network traffic,Data
is encrypted,Data is encrypted,Passed,High,Bameen Alam,SSL encryption working fine.
TC007,Test backend integration for cart,1. Add item to cart 2. Check cart count in API
response,Correct count displayed,Incorrect count displayed,Failed,High,Bameen Alam,Cart
backend synchronization issue. TC008,Verify error handling for network failures,1. Simulate
network failure 2. Perform an action requiring network,Error message displayed,Error message
displayed,Passed,Medium,Bameen Alam,Error messages are user-friendly. TC009,Validate
responsive design on mobile,1. Open site on mobile browser
2. Check UI layout,All elements align correctly,Alignment issues in
footer,Failed,Medium,Bameen Alam,Footer needs responsive fixes. TC010,Check loading time
optimization,1. Measure page load time using tool 2. Compare with target time,Page loads
within 2 seconds, Page loads within 1.8 seconds,Passed,Low,Bameen Alam, Performance
optimized.
```

Day 6

Deployment Preparation and Staging Environment Variable

Day 6 focused on deployment by setting up staging environment,

Key Outcomes

1. Set up and configure a staging environment for my marketplace. This includes; ▪ Selected a hosting platform like vercel.
    - Connected GITHUB repository with to the platform.

- Setting up environment variables securely within the hosting platform.
- Validated the functionality in a production like environment.
- Configure sensitive variables (e.g., API keys) in .env and uploaded them securely on vercel.

1. **Staging Environment Set up** ○ Deployed a staging build to validate functionality in a production like environment

   ○ For Example : .env.local File;

```
   NEXT_PUBLIC_SANITY_PROJECT_ID=" project id"
NEXT_PUBLIC_SANITY_DATASET="data set"
NEXT_PUBLIC_SANITY_API_TOKEN="token"
```

2. Staging Environment Variables

- **Functional Testing:** Verified workflows like product listing and checkout.
- **Performance Testing:** Used GTmetrix for analyzing speed and performance.
- **Security Testing:** Validated HTTPS, input handling and secure API calls.

3. Documentation

- Created READ.ME file and summarized the project structure and deployment steps.
- Organized the GitHub Repository with folder for src / , public / and documents.

Conclusion :

Over the six days, the clothing market place project progressed from concept to deployment, integrating robust features and ensuring a seamless user experience. With a Well-structured GitHub repository, dynamic components, and comprehensive testing, the project is now ready for live deployment in a production environment.