Prototype Report for Shared Grocery Ordering Feature

Introduction

The prototype was developed based on the EDR for the SSH Shared Grocery Cart feature. During implementation, some modifications were made to the original design due to time and resource constraints. Instead of using real supermarket APIs for fetching products and prices, these values were simulated by hardcoding them into a database. Realtime price updates were not implemented for the same reason. Similarly, nutrition advice was simulated by assigning a healthy score to each product, recommending replacement for less healthy items. Payment processing was not included; instead, the prototype directed users to the SSH Console Table for manual payment after confirming the order.

Despite these limitations, several key aspects were successfully implemented. These included:

- A homepage with login and signup functionality.
- A supermarket page displaying available products.
- A shared cart allowing multiple users to interact with it.
- A checkout page to confirm orders.
- An order confirmation page summarizing the order and guiding users to proceed with payment.

To assist with design planning, the team created wireframe sketches on paper and used Figma to develop high-fidelity prototypes of the application's user interface. This step helped refine the layout and user interactions before implementation.

Goals and Objectives

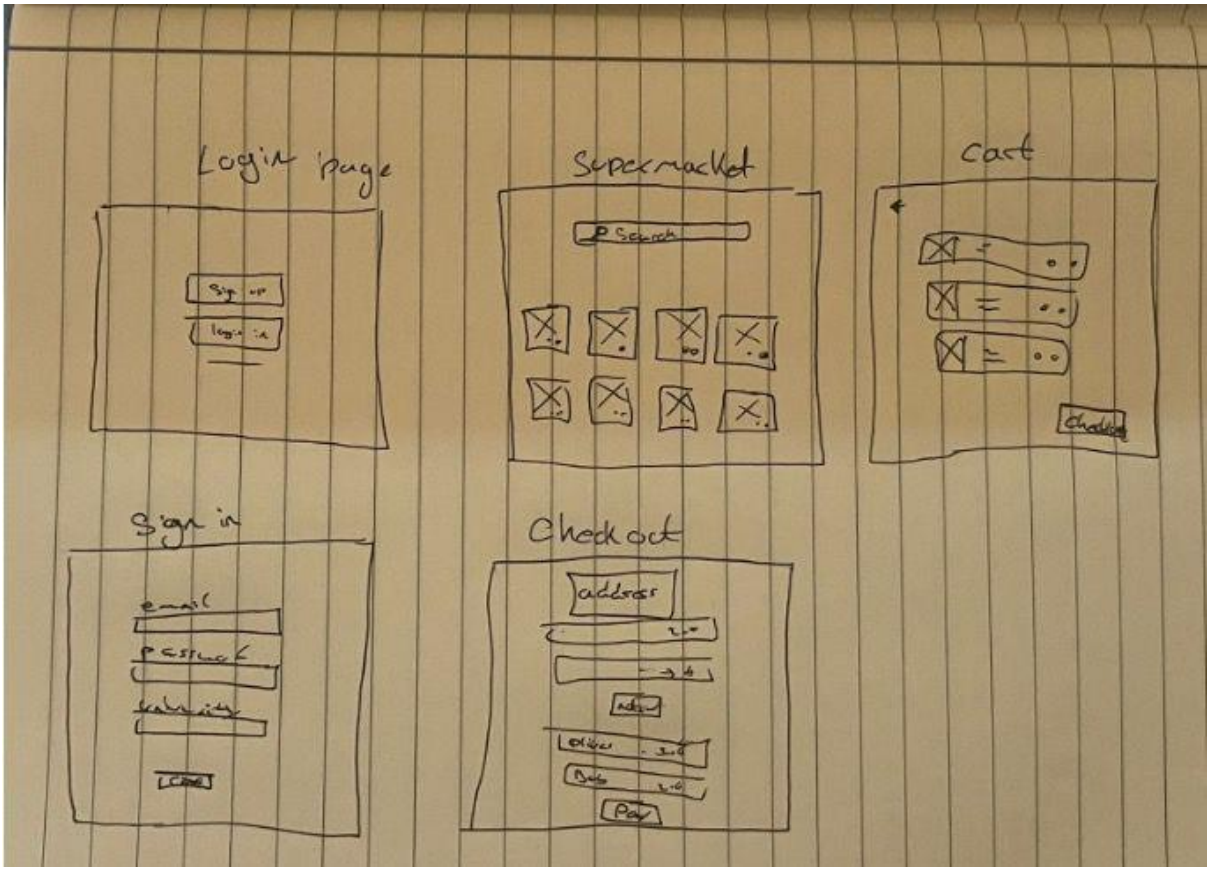
- Enable collaborative management of a shared cart.
- Provide real-time updates for item addition and cost tracking.
- Implement a functional checkout process.
- Simulate interactions with partner supermarket APIs and payment processing.

Design and Implementation

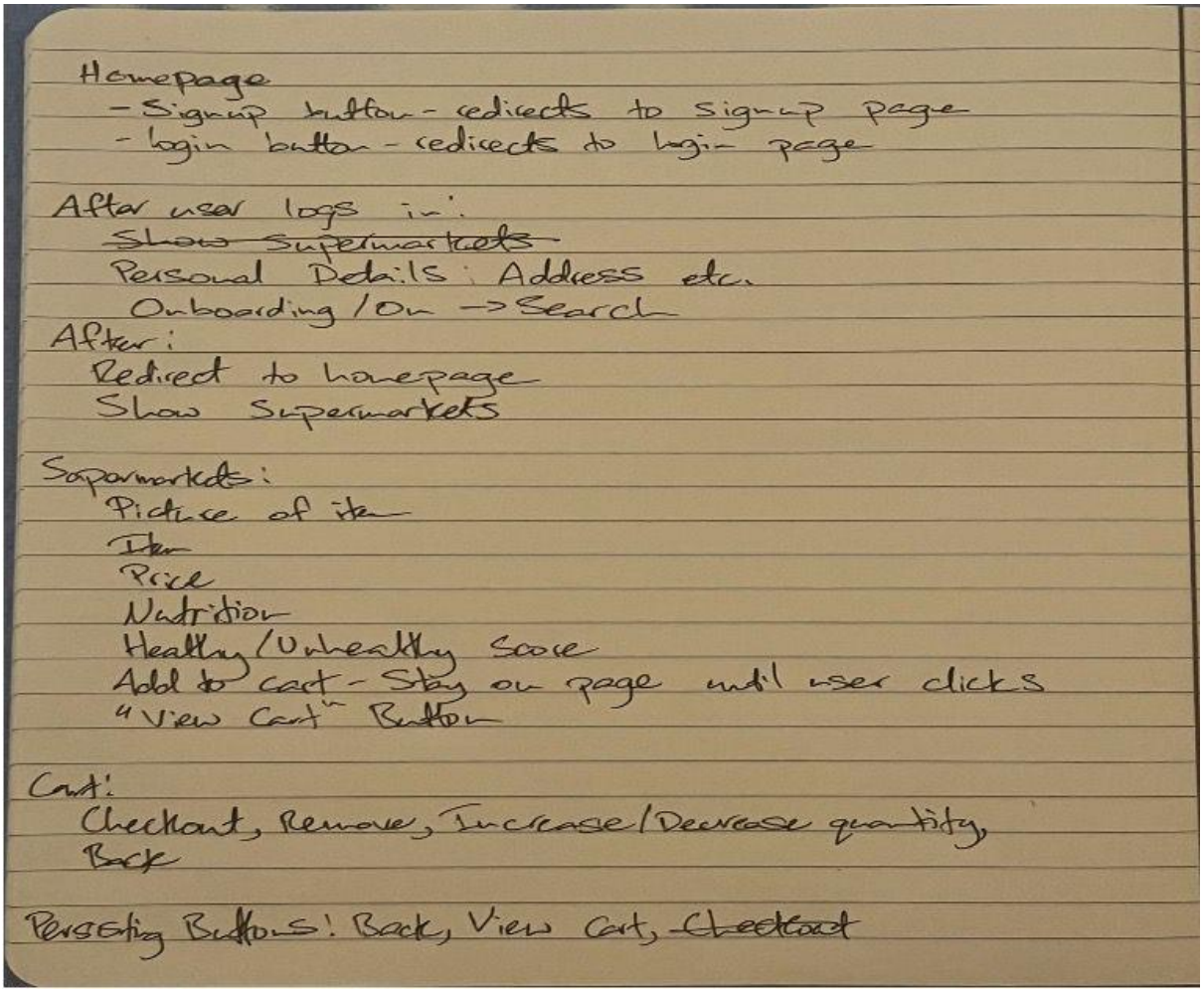
Technologies, Tools, and Techniques Used:

- Backend: Django (framework) and Python (language).
- Frontend: HTML and CSS.
- Design Tools: Wireframe sketches (on paper) and Figma for high fidelity prototypes.
- Testing: Built-in Django testing tools (extending Python's unittest).
- Observability: Used Python's logger for tracking information such as successes, errors, and other operational details.
- CI/CD: GitHub Actions was employed for continuous integration and deployment.
- Containerization: Docker was used to package the application and ensure a consistent development and deployment environment.

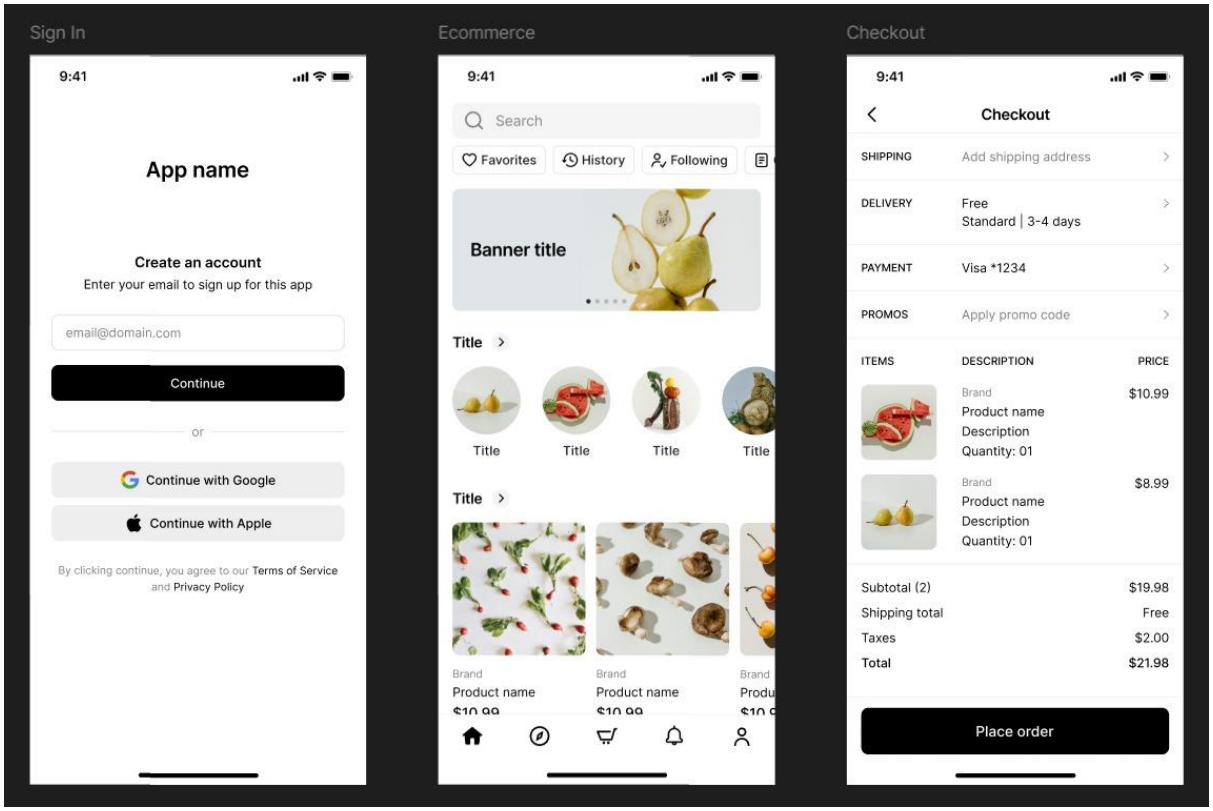
Simulating Larger System Dependencies
The prototype relied on hardcoded values in the database to simulate the behaviour of external supermarket APIs and other external systems. This approach allowed for testing and demonstrating core functionalities without requiring real-time connections to external services.



This is the wireframe used as reference for the intial design



The above screen shots shows the initial sketch of the design



These screenshots demonstrate the implemented user interface for the shared grocery ordering feature in the prototype. The design was developed using the Django framework with HTML, CSS, and Python. It provides a seamless user experience and adheres to the initial sketches and plans.

1. Sign-In Page:
Allows users to create an account or log in using their email and password.

2. Supermarket Page:
Displays a search feature and products from simulated supermarket data. Users can browse items and add them to the shared cart with detailed product information.

3. Checkout Page:
Summarizes the cart items with their total cost and shipping details. It includes checkout options and prompts users to place the final order.

Teamwork and Collaboration

The team utilized Git for version control, ensuring smooth collaboration and tracking changes effectively. Tasks were assigned based on individual preferences and expertise. Regular meetings were held to discuss issues, plan the user interface, and ensure alignment with project goals. Wireframe sketches and Figma prototypes were shared during these meetings to ensure clarity and agreement on the design approach.

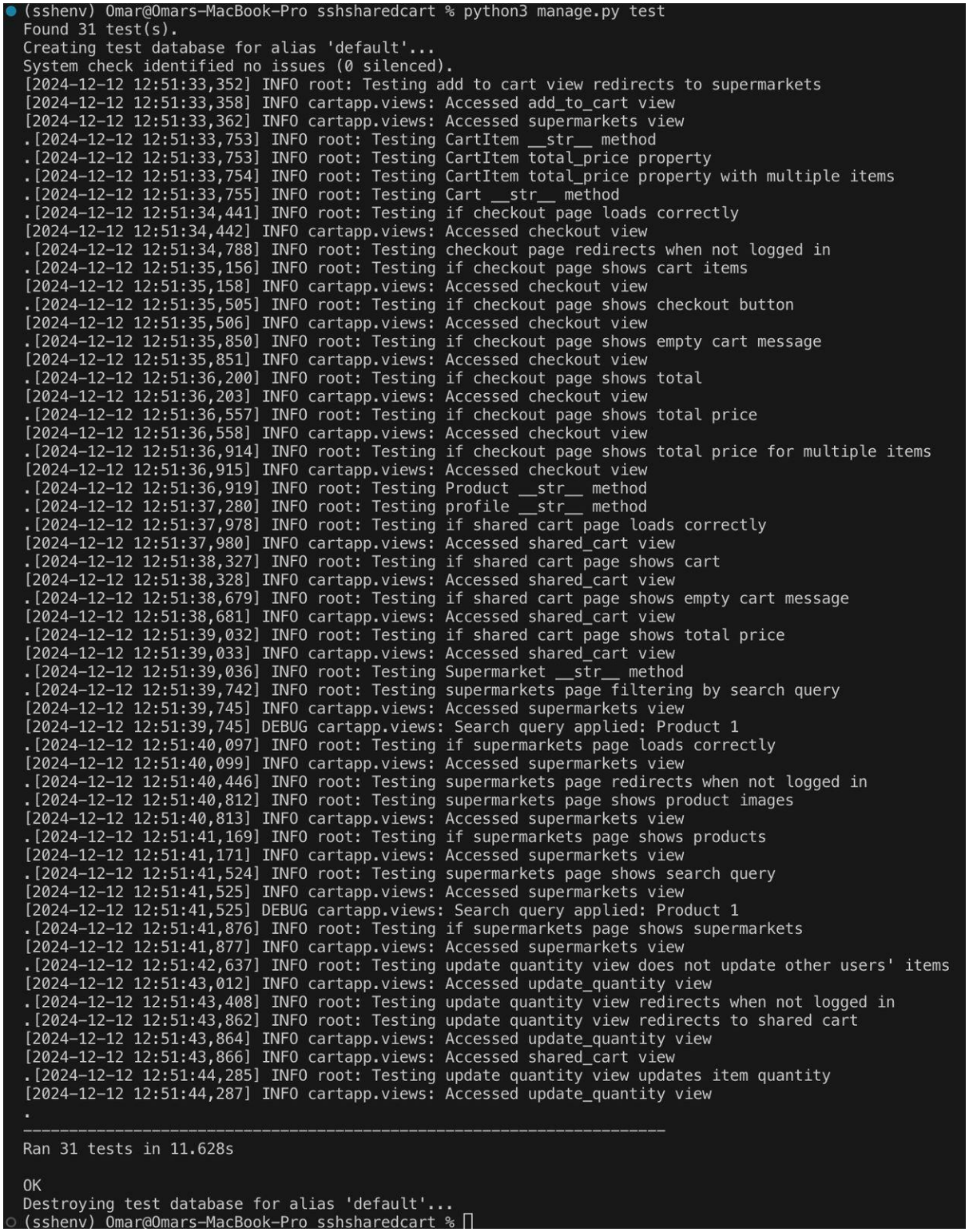
Challenges Encountered and Resolutions

- External API Integration: As API data was unavailable, hardcoded values were used to simulate data from external systems.
- Debugging: Several bugs arose during development, which required extensive testing and troubleshooting to resolve.
- Repository Issues: A temporary issue prevented committing or pushing code to the repository, requiring a restart of some work.
- Time Management: The team faced overlapping deadlines with other assignments and tests. To overcome this, tasks were prioritized based on importance and estimated completion time.

Testing Strategies

Testing involved creating unit tests using Django's testing framework. These tests ensured the following:

- Correct functionality of user authentication.
- Proper interaction with the shared cart (e.g., adding/removing items).
- Validation of order confirmation logic.
- Rendering of all frontend pages.



The above screenshot shows the results of running 31 test cases using Django's built-in testing framework. All tests passed successfully, confirming the functionality and stability of the prototype's core features, including the shared cart interface and simulated API interactions. This testing step ensures that the system meets the outlined requirements and performs as expected.

Results and Evaluation

The prototype successfully demonstrates the following:
Collaborative cart management and user contribution tracking.
Simulated API functionality for price and product data.
Functional checkout process.

Conclusion

The grocery ordering prototype effectively illustrates the practicality of the design outlined in the EDR. By concentrating on essential features, the project highlights a workable system for managing groceries collaboratively. This prototype lays a solid groundwork for continued development.

Suggestions for Future Productivity

- Better Time Management: Allocate time specifically for the project in advance to avoid overlap with other assignments.
- Enhanced Communication: Increase the frequency of brief progress updates to ensure all team members stay aligned.
- Early Issue Identification: Address repository or tooling issues immediately by seeking help or restarting work earlier.
- More Collaborative Debugging: Pair programming or joint debugging sessions could help resolve technical challenges faster.

Evaluations of Group Members

Umair’s reflection:

Reflection on Omar (Backend):
Omar demonstrated exceptional leadership qualities throughout the project. His efficiency in handling the backend code ensured that the core functionality of the prototype was implemented seamlessly. He was always available to assist the team whenever help was needed and addressed doubts with patience and clarity. Omar also contributed significantly by proposing the EDR we chose, which aligned perfectly with the project's objectives. His ability to meet deadlines and maintain a positive team dynamic made him an invaluable member of the group. To enhance his already impressive skills, Omar could further explore innovative backend optimization techniques for future projects, which would elevate the quality of his work even further.

Reflection on Hamad (Frontend):
Hamad excelled in designing and implementing the frontend of the prototype, creating a user interface that was both intuitive and visually appealing. His contributions to the design and implementation sections of the report showcased his clear understanding of the system’s functionality. Hamad's assistance in crafting the report was instrumental in ensuring its comprehensiveness and professionalism. His phenomenal UI design skills provided a strong foundation for user interactions within the prototype. Moving forward, Hamad could focus on incorporating advanced design frameworks or tools to further refine his work and enhance his development speed.

Omar’s Reflection:

Reflections on Hamad (Frontend):
Hamad demonstrated strong creativity in designing an intuitive and visually appealing frontend. They effectively implemented responsive pages, including the homepage, supermarkets page, and checkout interface. Hamad also contributed significantly to the wireframe sketches and Figma prototypes, ensuring the design was consistent with the project’s objectives. However, Hamad could improve their productivity by adhering to tighter deadlines and ensuring their code is consistently documented for easier collaboration.

Reflections on Umair (Report)
Umair contributed significantly to the report, organizing the content clearly and ensuring it covered all required details. They also coordinated with other members to gather screenshots and pictures of the prototypes and tests. However, Umair could improve by proactively seeking feedback earlier in the drafting process to minimize last-minute revisions.

Hamad’s Reflection:

Reflection on Omar (Backend):

Working with Omar was overall a positive experience, as they demonstrated a strong work ethic and technical expertise throughout the project. Their ability to focus on tasks and deliver high-quality outputs on time was a significant asset to the team. Omar contributed extensively to the backend development, particularly in structuring the models and implementing complex views. Their technical skills ensured that the project met its functional requirements.

However, there were occasional challenges in team cohesion and communication. Omar tended to work independently, often preferring to tackle problems alone rather than collaboratively. While this demonstrated their problem-solving abilities, it sometimes led to duplicate efforts or misaligned priorities within the team. For example, there were instances where code changes were made without clear communication, resulting in merge conflicts and delays.

To improve collaboration in future projects, I would suggest that Omar focus on increasing their engagement during team discussions. Proactively sharing updates and seeking input from team members could help align efforts and enhance overall team productivity. Additionally, participating more actively in collaborative decision-making would create a stronger sense of unity within the group

Reflection on Umair (Report):

Umair played a vital role in producing a clear and comprehensive project report, ensuring that our work was well-documented and presented professionally. Their strong writing skills and attention to detail were evident throughout the process, as they consistently delivered high-quality content for the report. Umair’s ability to synthesize technical details into concise and understandable language was particularly valuable, as it made the report accessible to both technical and non-technical stakeholders.

However, there were occasional challenges in aligning their progress with the rest of the team. At times, Umair would work independently on sections of the report without fully integrating feedback from team discussions. This occasionally led to discrepancies between the report’s content and the actual implementation of the project. For instance, certain technical details were either oversimplified or missed due to limited consultation with the development team.

To address this, I would recommend that Umair increase their involvement in collaborative discussions during the early stages of the project. Regularly checking in with the team to ensure accuracy and consistency in the report’s content would help align their efforts with the broader goals of the project. Additionally, incorporating feedback iteratively rather than at the end of the writing process could streamline revisions and reduce last-minute edits.