Project Lifecycle for Online Food Delivery System

The lifecycle method chosen for the execution of the project is Agile methodology. The reasons for this choice are:

- Agile supports incremental development and continuous feedback, which is necessary for the evolving requirements of an online food delivery system.
- It allows flexibility to integrate new features as customer needs change.
- Continuous improvement through sprint retrospectives helps refine the delivery process.
- Enables frequent collaboration through scrum meetings, essential for coordinating between restaurants, delivery personnel, and customers.
- Facilitates better team collaboration and communication across different aspects of the system.

Plan for implementation using Agile:

1. **Project Initiation**
    - Identify all the features needed for the project with reference to the SRS document and create a product backlog.
    - Prioritize features based on business value and dependencies.
2. **Sprint Execution**
    - Hold meetings to discuss progress, plan, and address blockers.
    - Develop features incrementally and continuously test them.
    - Sprints will be of 1-week duration.
3. **Reviews and Retrospection**
    - At the end of each sprint, demonstrate the completed work.
    - Get a set of reviewers (including potential users, restaurant partners, and delivery personnel) to try the interface and provide feedback.
    - Reflect on the sprint to discuss what went well, what challenges were faced, and how the next sprint can be improved.
4. **Backlog Refinement and Iterative Deployment**
    - Based on feedback, update the product backlog.
    - Reprioritize new features or enhancements.

    - Continue working on the project and aim for a potential finished product at the end of each sprint.

5. **Final release and continuous improvement**
    - Once core functionalities are stable and tested, release the system.
    - Facilitate post-release maintenance and updates.

Tools Selection

- Planning Tool: Jira
- Design Tool: Figma
- Version Control: GitHub
- Bug Tracking: Jira
- Development Tool: VSCode (primary IDE)
- Testing Tools: Postman (for API testing), Selenium (for web testing), Jest (for Unit testing)

Deliverables Categorization

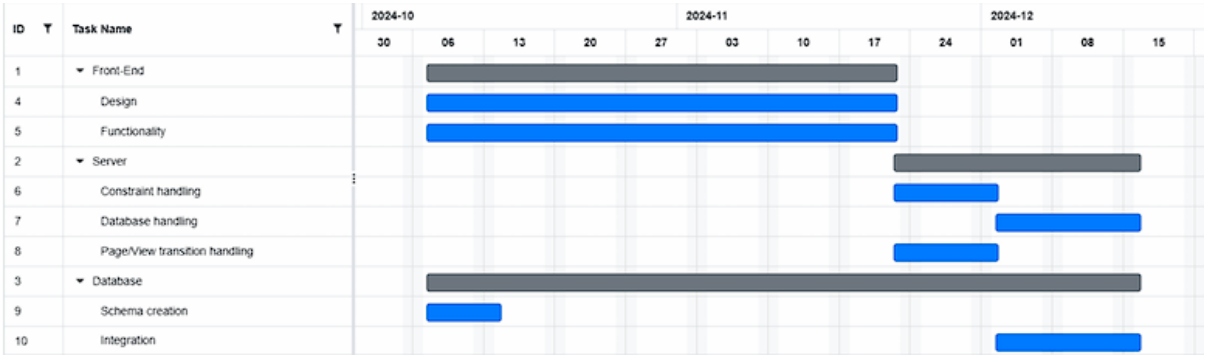
Reuse Components

- Database Management System (MySQL, PostgreSQL)
- Authentication System
- Payment Gateway Integration

Build Components

- Restaurant Management Module
- Menu Management System
- Order Processing and Tracking System
- Delivery Personnel Management Module
- User Profile and Preferences System
- Real-time Order Status Tracking
- Rating and Review System
- Reporting and Analytics Dashboard

Work Breakdown Structure



UML Diagrams