PROJECT REPORT
ON

Software Development Lifecycle (SDLC) Analysis of Online Food Delivery

A brief study on different models in relation to online food delivery

Submitted to

NMAM Institute of Technology, Nitte
(Off-Campus Centre, Nitte Deemed to be University, Nitte – 576110, Karnataka, India)

On partial fulfilment of the requirements for the award of the
Degree of Bachelor of Technology
In
INFORMATION SCIENCE AND ENGINEERING
By

Apoorva Shetty NNM23IS021

Under the guidance of

Dr. Jason Elroy Martis, Associate Professor,
Department of Information Science and Technology,
NMAM Institute of Technology, Nitte, Karnataka, India

{ nnm23is021@nmamit.in}

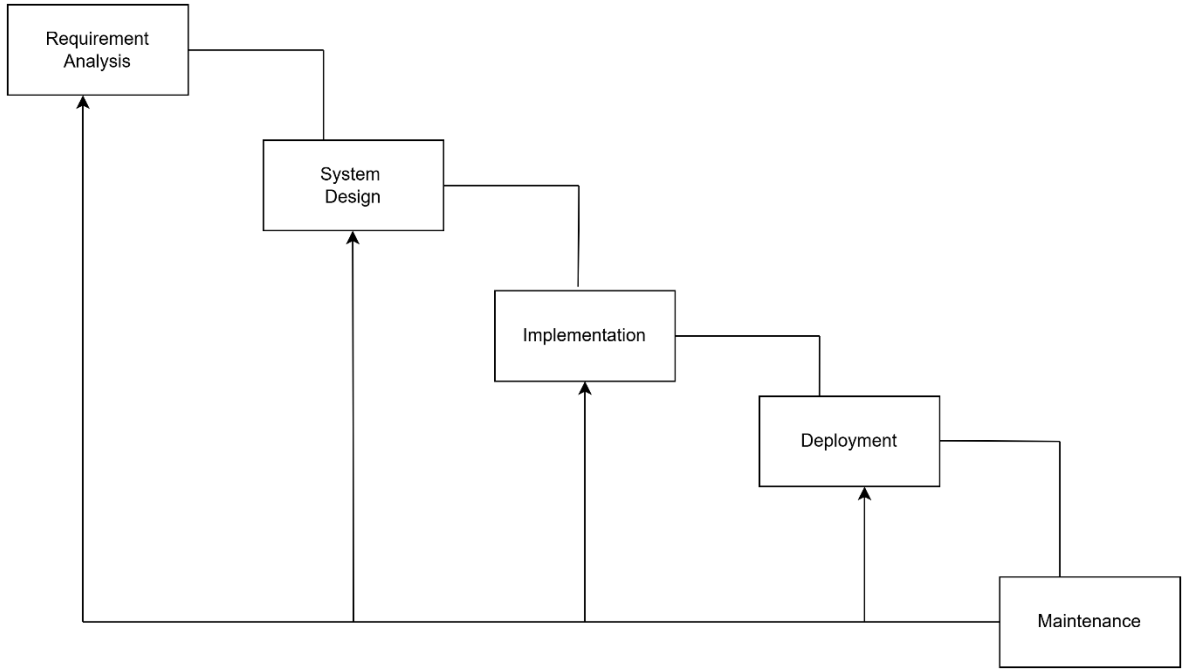
1. Introduction

Online food delivery plays a very important role in our day-to-day life ,it provides good services for the customers with convenient access to a wide range of restaurants and cuisines to directly to customers location which provide comfort for customers . This system depend on robust, cloud-based infrastructures to handle high traffic,real time order processing ,and secure payments .By the dynamic nature of industry, an perfect Software Development Lifecycle(SDLC) is very critical to ensure a seamless customer experience ,timely updating, and reliability.

This report shows us a comparative analysis of the Waterfall model, Incremental Development and Spiral Model with reference to online food delivery app's. It also examines requiremental engineering process and key challenges in developing and maintaining these types of system.

2. Comparative Analysis of SDLC Models

2.1 Waterfall Model



Outline of Waterfall Model

Development of Online food delivery using Waterfall model

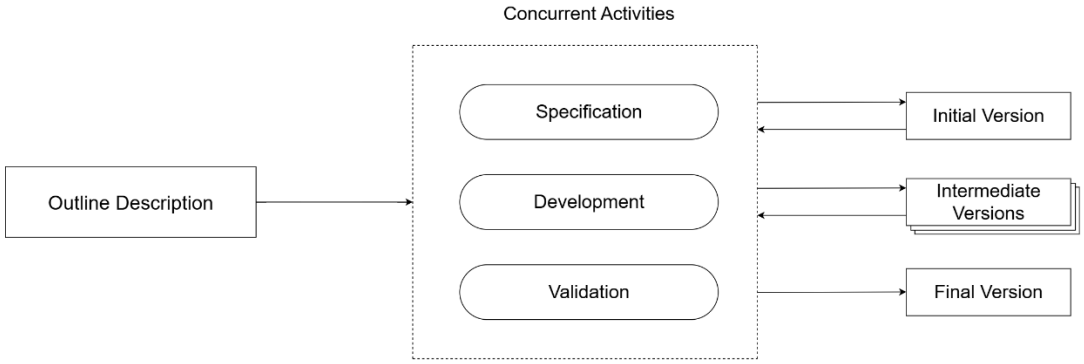
- **Requirements Analysis** – All system requirements are defined at only once in this phase. It includes defining user authentication, content streaming capabilities, personalized recommendations etc . In short, developers try to gather any and all requirements in this stage. Since changes are difficult to implement later, this is very important and exhaustive documentation is required at this stage if in case we are developing an app for online service.
- **System Design** –The complete architecture of online food delivery service happens before any coding begins. This phase gives blueprint of requirement analysis phase by trabsforming function and non-functional requirements, since this model follows a sequential process the layouts should be fully designed before the actual development starts .This model requires exhaustive planning upfronts ,thus this phase becomes very critical to the projects development.

- **Implementation** -The actual development of online food delivery happens in this stage .After completing the requirements analysis begin by forming code from raw .This process involves linear development. Since waterfall model doesn't support iterative method the developer should be careful while following the pre-defined architecture of the system structure of database and its features so that the system would be developed in well planned manner. Thus, this model requires the whole system to be implemented before testing the system.

- **Deployment** – After development is fully completed, the entire system undergoes integration and rigorous testing to bring into real world so that it will be friendly to the users. This phases involves certain steps of deploying the application on servers, app stores and wed system, by doing this we check whether the system is fully operating and accessible to users .But waterfall model mainly focuses on sequential approach ,deployment happens after gathering ,designing ,testing, implementation of the requirements are fully completed.

- **Maintenance** – Before the customer use the app the developer tests , deploy the platform this ensures good maintenance for future **.**In maintenance stage the system ensures functional, efficiency and security for end users , including restaurant management ,customers and delivery partner .These are the attributes that should be included in this stage hence, it becomes challenging to maintain waterfall model.

- ## 2.2 Incremental Development Model



Outline of Incremental Development Model

Incremental Development of Online food delivery:

- **Outline Description** – For online food delivery system an high level plan is required by identifying and including core features example adding essential functionalities like user registration, lists of the restaurants ,locations of the respective orders and payments.

- **Concurrent Activities**–This activities in parallel with each increment by including certain stages like specification, development ,validation. Data are gathered and documented for the requirements for each increment. Like the very first rise in increment may specify the need for users restaurants browsing. In development stage for defined requirements we need coding and implementation example adding login system. For validation testing is performed by checking whether the functionality meets users expectation., example testing user's login system or display of the restaurants .

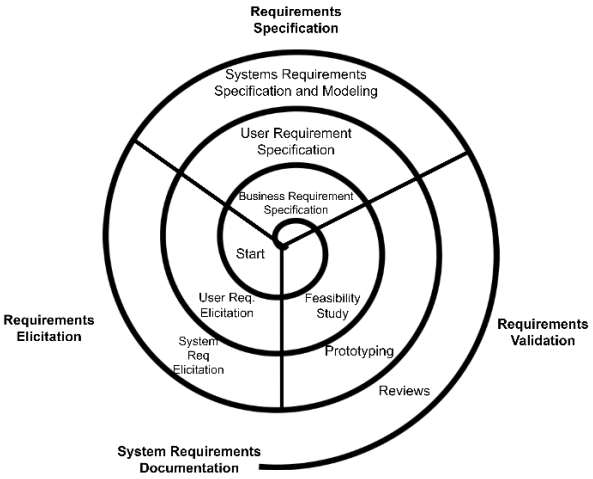
- **Initial Version** - Initial version of incremental model focuses on building and developing the system's functionality for online food delivery system. In this version it provides the basic frameworks of the system's application , so that the customer's can interact and give feedback. The main features that are present in initial version is user's registration or login, restaurants menu and the design of ux/ui models. The main of initial version is that this stage provides prototype for working and test the concepts.

- **Intermediate Version**-Intermediate version mainly focus on slowly adding the features by day-by-day by including the feedbacks which in return results with gradual improvements of the system .In intermediate stage increment happens by building upon on initial version, so that the system becomes

user friendly and functional .This stage includes food ordering, payments integration and order tracking system user reviews & ratings ,restaurant dashboard and ui/ux improvements.

- **Final version**–In final version of online food delivery system we fully develop and optimize this platform which includes all features such as previous version and updating. For large scale deployment ,smooth functioning and scalability we need this version.

2.3 Spiral Model



Outline of Spiral Model

- **Development of Online food delivery using Spiral model**
- **Risk Analysis and Prototyping** – The objectives highlights of gathering of all documents and in activities the developer should include identifying stake holders such as customers ,restaurants and delivery agents. In defining the key features we include user registration, restaurant listing , order placement ,payments ,tracking etc .By following these steps we can achieve software requirement specification document commonly known as srs.
- **System Designing Phase** –By using objectives we design system architecture and interface .In activities area we create high level architecture like frontend, backend and database design. We can also design ui/ux mockups like user dashboards ,restaurant interface and order page ,here we can also define database like user details ,restaurant menus, or order tracking .By doing this 23 can achieve system design documents also known as sdd
- **Implementation (Coding) Phase**-Using objectives we convert system into actual code .In frontend department user registration, restaurant listing and order placements happens .In backend development department business logic for order processing, payments and notification happens .Thus, this ensures basic functionality and in basic functionality works are done as per designs. Therefore we can achieve executable code for initial features.
- **Testing Phase** –By using objectives we can identify , fix debug to ensure software quality .In activities area unit testing, integration testing ,performance testing, security testing happens .By doing this we can achieve bug-free and fully functional system.
- **Deployment Phase** – Using objectives we can launch the online food delivery system for users .In activities the developer should include deploying on live servers like aws, google cloud, and azure .In activities the developer should make system available on web ,android and ios and also provide training to restaurants and delivery partners .By doing this we can achieve live online food delivery system
- Maintenance & Support Phase–In objectives developer continuously improve and update the system .In activities are or phase the developer should fix bugs and security vulnerabilities found after previous phase and also the developer should optimize performance of the app for the better users experience and

also should release new features like ai- based recommendations and voice recording .By doing this we can achieve scalable and stable online services.

- **2.4 Highlighting Suitability by Comparing the models**

1.Wterfall Model-This model lacks flexibility ,making it less suitable for online delivery uses because online delivery requires lot of frequent updates and improvements day by day. It has low risk management capabilities, a slow development process and average cost .Due to model's hard structure, it is not easy for dynamic and evolving platforms like delivery services.

2.Incremental Model-Flexible and gradual improvements, making it a great fit for online service system.Risk management is balanced here and new features can be included very quickly ,ensuring for smooth service s.Due to ongoing development cost will be raised eventually .Therefore, this model is highy suitable for our topic.

3.Spiral Model-This model gives higher risk management and higher flexibility making it suitable for developing high-risk features such as AI-based recommendations ,advanced route optimization or fraud detection .While it ensures whether time-to time market is moderate or not, development costs are high and also ensures structured approach .Therefore, this model is best for specific high-risk features instead of the entire online food delivery organization

3. Requirements Engineering Process for Online Food Delivery Analysis

Requirement engineering plays a very important role in developing an efficient and scalable online food delivery system .It ensures that system meets to the customers expectation ,fulfil their business requirements ,and function properly in real world situations .The information showed below includes functional and non-functional requirements ,in a detailed structured manner.

3.1 Functional Requirements

By Functional requirements, I mean the specific behaviours and functionalities that it must support to meet the user expectations. Now, these will include:

- **User Authentication & Registration**: Customer should be able to register using their any one of their personal details for security purpose and smooth functioning in future .Forgotten password recovery option should be available .Secure login system should be given including password authentication ,OTP receiving option and biometric options like fingerprint ,face recognization etc.
- **Food Menu**: The app should allow the restaurant owner to create and manage their profile ,including their names ,location, and operational hours .The owners should be able to update the menu in real time .Owners should be able to list their food with description .
- **Filters & Search**: Customers should be able search restaurants ,cuisines, or dishes using keywords .Advanced code should be included so that we get sorted list of prices that applies to the search button. Search button should be inappropriate to AI-based recommendations based on orders and user preferences.
- **Order for the location & Customization**: User should be able to add foods as much as they want as per their wish based on quantities even options for customization of food should be available ,the app or system should provide real-time order confirmation and arrival time and also track the order .System should provide the option to track customers location.
- **Payment Integration** :System should allow multiple payment method including credit/debit, cards , UPI, cash on delivery etc .Secure payment method should be provided since user will be sharing beneficial data.
- **Ratings and Experience of Customer** :User should able rate the service and also the food delivered at their residency so that it will easier to restaurant owners to keep track on their workers and taste of their food .It is also helpful for the organization to keep track on the delivery services.

3.2 Non-Functional Requirements

Non-functional requirements define the system's operational quality and constraints.

- **Scalability** : The system must handle millions of concurrent users without a drop in performance. Cloud-based infrastructure should be included to support auto-scaling.
- **Performance**: The system should be able to handle growing users day-by-day. API response should be optimized .
- **Security**: User's beneficial data including their personal details and payment information must be stored securely, MFA should be provide for the user.
- **Availability**: The should be available for the user or customer 24/7,so that the user can order at any time .Notification alter should be provide for future misconception .
- **Maintainability** :It should main characters such as efficiency and compliance so that system adhere to food industry regulation(fssai) ,tax ,gst policies .The app should provide low consumption, ensuring minimal online payments.
- **Compatibility** : System should include integration with smart assistants .UI should include different screen sizes for laptop or mobile phones.

3.3 Strategy for Requirements Validation

A well-defined requirements validation strategy must be implemented so that it meets users expectation ,objectives of business standards for regulation. By including following techniques validating software requirements can ne achieved i.e, stakeholders reviews, users feedbacks ,security check and automated testing .In stakeholders reviews developer engage restaurants owners ,customers and delivery partners requirements before starting the implementation . By including certain components such as login, order placement for users location and payments we can ensure unit testing ,through testing we can check interaction between various modules like restaurants data history, tracking the delivery and payment history. Security plays a vital role in building the strategy because users personal data will be stored, so keeping this beneficial data of users is very important. By collecting the reviews from users actually improves services because we can meet to their requirements through app-survey .

3.4 Challenges in Requirements Validation

Despite a good structured validation approach, online food services faces several challenges arise in requirements validation which should taken care to build reliable ,scalable , and efficient system. In online food delivery service different stake holders faces different types of problem so if we go on including their expectation it becomes very hard to find one solution for many problems . Restaurant management may prioritize menu customization and analytics but the customers require fast delivery and secure payment procedure .The market trends day-by-day so its very difficult to keep on updating the requirements and build model .Without disrupting system stability demands for different seasons, food trends and advertising need to be integrated. Verifying payments with high security now and then becomes problematic because of multiple transaction. Also the platform should manage time especially in peek hours due to high traffics.

4. Conclusion

Requirement engineering process checks the an online food delivery system is well-planned , proper functionality and aligned with user expectations .The functional requirements explains the core features of the app or system as registration, order placement ,real-time tracking and payment processing. While the non-functional requirements ensures performance , scalability ,availability, maintainability, compatibility .By above structured approach ,the software developer's can build efficient and user friendly food delivery platforms.