**Chapter 2:**

2.1 Analysis:

Software Development Life Cycle’s second phase is the analysis phase from where the life cycle takes place. Analysis is a thorough study of examining the features in detail. It focuses on determining the key features such as requirements gathering technique of the end user, quality prerequisites and examining the necessity. It provides a general overview of how the system works and what must be done. It knows the diagram of what a framework will do and discover the hazard and remunerate and execute the exchanging as a business. It picks up the learning and protect our ventures and make benefits. It helps in critical thinking and basic leadership.

2.2 INFORMATION GATHERING TECHNIQUE:

1. Questionnaires:

2. Survey:

3. Observation:

4. Interview:

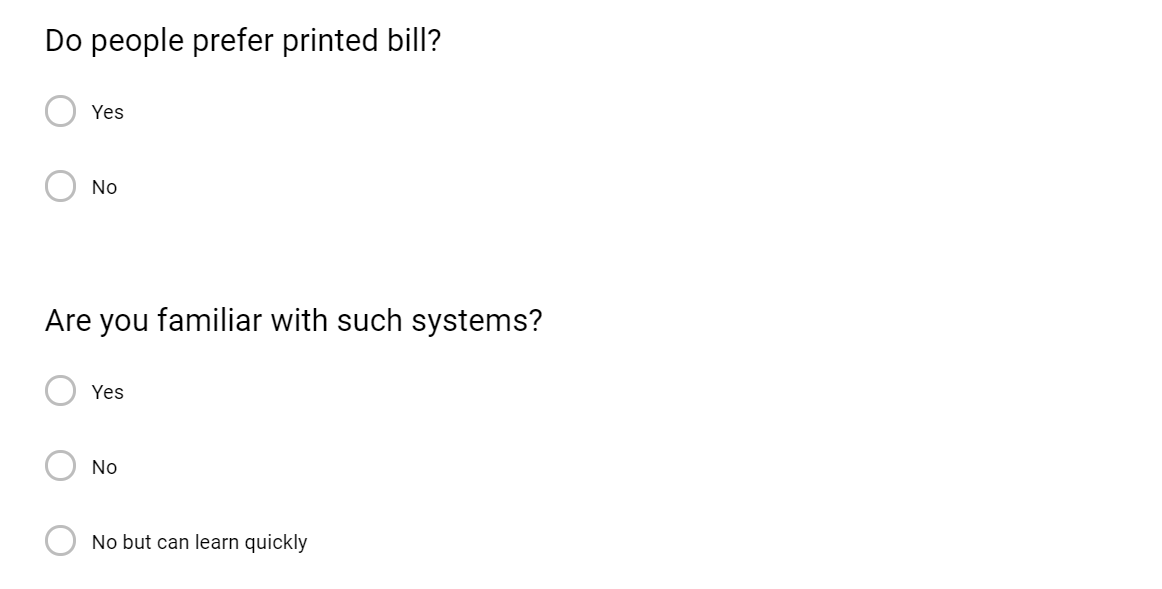
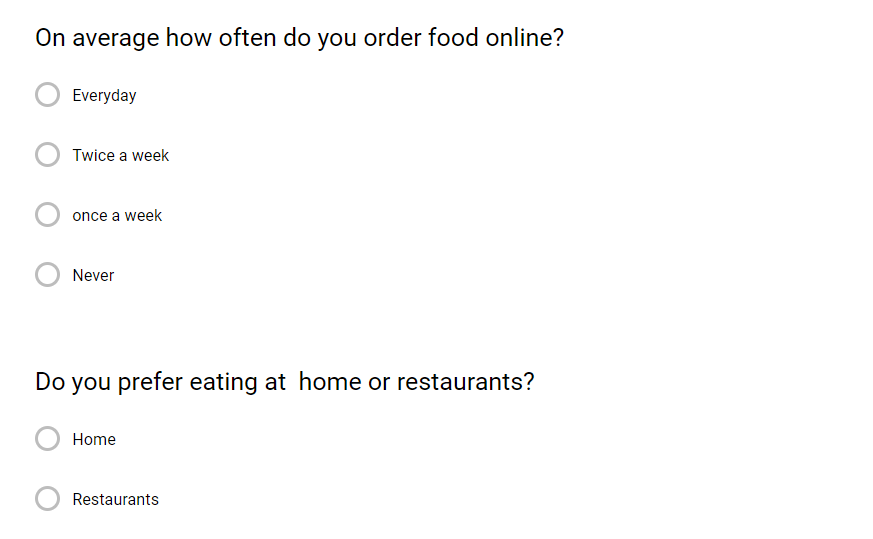
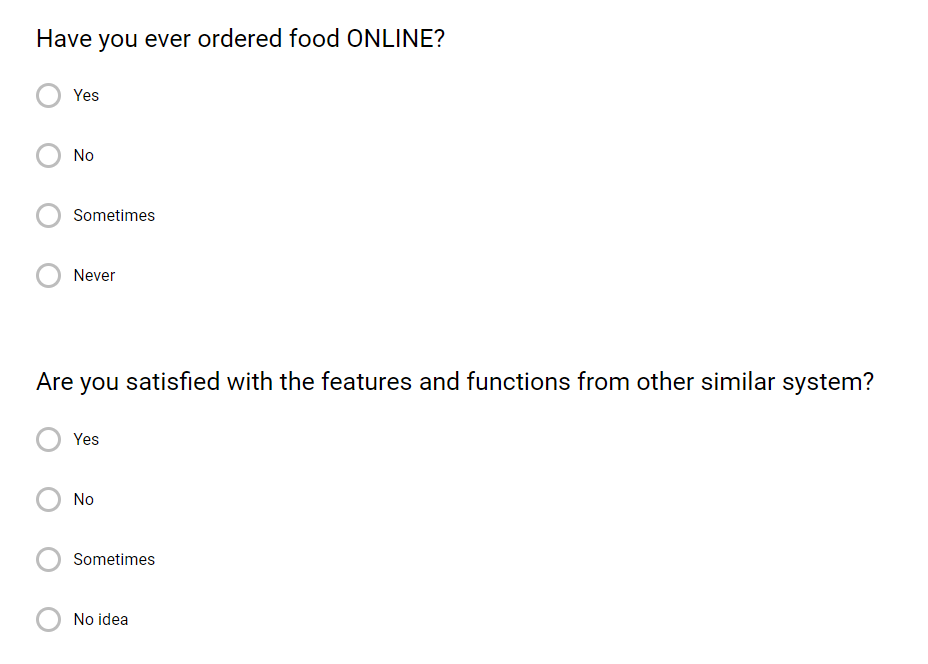
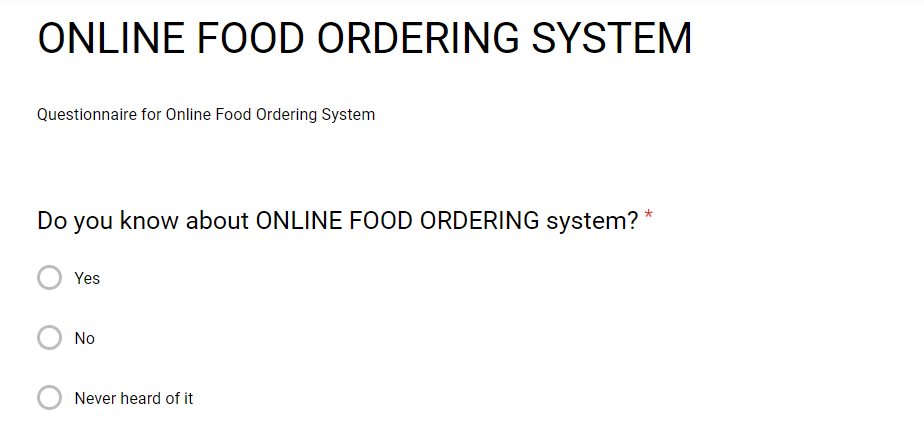
5. Brainstorming:

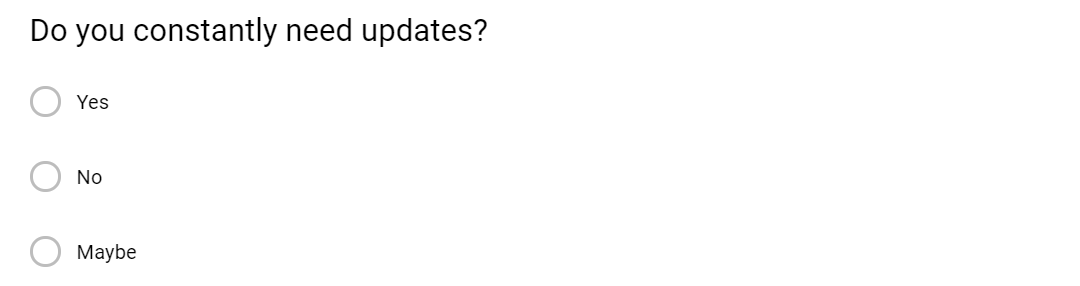
6. Interface Analysis:

7. Prototyping:

1. Questionnaires:

Few questions were asked to the people randomly regarding the online food ordering system. Here, they are.





2.3 Feasibility Study:

This is an assessment and examination of the capability of the proposed undertaking that is related on extensive examination and research to help the procedure of basic leadership. It evaluates the operational, specialized and monetary benefits of the proposed undertaking. The practicality think about is intended to be a primer audit of the certainties to check whether it is deserving of continuing to the examination stage. From the frameworks examiner point of view, the achievability investigation is the essential apparatus for prescribing whether to continue to the following stage or to end the task. There are various kinds of feasibility study which are briefly described below:

2.3.1. Economic feasibility study:

This evaluation means to decide the positive financial advantages to the association that the proposed framework will give. It normally includes a cost/benefits investigation and it's the most much of the time utilized technique for assessing the adequacy of another proposed framework. Possible questions brought up in monetary examination are

2.3.2. Operational feasibility:

Operational possibility is the proportion of how well the task will bolster the client and the specialist organization during the operational stage. It is subject to HR available for the venture and includes anticipating whether the framework will be utilized in the event that it is created and implemented.

The fundamental inquiries that help in testing the specialized attainability of a system include the accompanying:

2.3.3. Social feasibility:

Social feasibility is one of the common sense contemplates where the affirmation of the overall public is considered with respect to the thing to be pushed. It depicts the effect on customers from the introduction of the new structure considering whether there will be a prerequisite for retraining the workforce. It depicts how you propose to ensure customer co-task before changes are I have done social credibility examination of this errand. This endeavor is offering organizations to people and it will recognize by people. So, looking at social point my errand is socially down to earth to make.

2.3.4. Technical feasibility:

This evaluation is centered on picking up a comprehension of the present specialized resources of the association and their appropriateness to the normal needs of the proposed framework. It is an assessment of the equipment and programming and how it addresses the issues of the proposed framework. The systems task is considered actually attainable if the interior specialized ability is sufficient to bolster the venture necessities. The basic inquiries that help in testing the technical feasibility of a framework incorporate the accompanying:

2.3.5. Schedule feasibility:

This evaluation is revolved around getting an understanding of the present specific resources of the affiliation and their pertinence to the typical needs of the proposed system. It is the extent of how reasonable the endeavor time table is or the due date is reasonable or not. During the nonattendance of time or the time become required, we should finish the errand within a given timespan. It chiefly addresses:

Can the assignment genuinely be done in the given time span

2.4 Software Requirement Specification:

A Software Requirements Specification (SRS) is a record which depict system that will be made similarly as the objective of the structure a work in advancement. Programming requirements specific shows what the system ought to do similarly as how it ought to perform. It is recorded before the authentic programming progression work starts.

It's noteworthy are given underneath:

•It diminishes improvement cost.

•It moreover helps clear any correspondence issues between the client and the designer.

•It moreover helps clear any correspondence issues between the client and the designer.

2.4.1. Functional Requirement:

Functionality prerequisite are rundown of exercises that item or framework must accomplish for clients. It powers what a product should do. Think about a model, where a framework must enable client to login when clients have entered right username and password. It very well may be arrangement of calculations or data control or processing of data. Here are few useful necessities that is required in the framework.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirement ID | Title | Description | Rational | Dependency |
| FR1 | Registration | User should be able to register when user provide their details. | To register account for new user. | N/A |
| FR2 | Login | User should be able to Login in using email and password only after they have been registration. | To use services of application. | FR1 |
| FR3 | Change Password | The user should be able to change their password when they are logged into their account. | Change and make new password. | FR2 |
| FR4 | Navigate Menu | The user should be able to navigate the menu. | Navigate details of the menu. | N/A |
| FR5 | Select Items | The user should be able to select the food items from the menu. | Selecting Items | FR2 |
| FR6 | Customize Options | The user should be able to customize the item they select. | Customize selected Items | N/A |
| FR7 | Add Item | The user should be able to add the item to their current order. | Add | FR5 |
| FR8 | Review | The user should be able to review the items they ordered. | Review current order | FR7 |
| FR9 | Delete | The user should be able to delete the item or the items from current order. | Custom Changes |  |
| FR10 | Order | The user should be able to place an order. | Place the order | FR7 |
| FR11 | Delivery and payment | The user should be provided with details of payment and delivery. | Details about payment and delivery. | FR10 |
| FR12 | Confirmation | The user should receive confirmation in the form of an order. | Confirmation of the order | FR11 |

2. Non functional

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirement ID | Name | Description | Rational | Dependency |
| NFR1 | Scalability | System should withstand data of different sizes i.e. from bytes to megabytes. | To make system work with different volumes of data. | N/A |
| NFR2 | Performance | System should respond in time with fast performance. | To make system function properly. | N/A |
| NFR3 | Reliability | System should give right result on search | To ensure result is correct. | N/A |
| NFR4 | Availability | System must be easily accessible by users. | To provide system availability. | N/A |
| NFR5 | Maintainability | System must recover possibly even after failure. | To make operational system by restoring failures. | N/A |
| NFR6 | Security | System must maintain data integrity and encrypt user details. | To secure confidentiality. | N/A |
| NFR7 | Environmental | System should be accepted socially. | To increase flow of users. | N/A |
| NFR8 | Usability | System should be easy to use. | Encourages user to use the system. | N/A |
| NFR9 | Legal | System should follow legal policy. | To provide legal security. | N/A |

2.5 PRORITIZATION:

So as to isolate the necessities into various advancement stages, prioritization of prerequisite is required. This area depicts decision of prioritization technique for various kinds of necessity recorded previously.

Prioritization is procedure of recognizing which errands are progressively vital at current minute. It conveys most quick need during venture advancement. It finishes distinctive necessity in various occasions as all usefulness and requirements are not same.

MoSCoW prioritization methodology orders requirements into high, medium and low hugeness. MoSCoW standard confines most huge and least critical ones. It delineates which essential to be done from the start and which one at long last. MoSCoW rule in this system is recorded underneath.

Must have: It joins least game plan of essentials undertaking set to pass on. Any need in this order can change achievement of undertaking. In the occasion that must have essential is rejected, the endeavor can be called as disillusionment.

Should have: It depicts set of necessities which are possible to pass on. Any segment which has high a motivator to all customers is all things considered not huge current falls under this characterization.

Could have: It consolidates need which doesn't impact anything on achievement of undertaking. These requirements are not too appealing and can be removed at endeavor disillusionment time.

Won't have: It depicts set of necessities which isn't to be complete during current stage yet included later in further progression cycle.

|  |  |  |
| --- | --- | --- |
| **Requirement type** |  | **Prioritization level** |
|  |  |  |
| Must have |  | High |
|  |  |  |
| Should have |  | Medium |
|  |  |  |
| Could have |  | Medium |
|  |  |  |
| Won’t have |  | Low |
|  |  |  |

MoSCoW Prioritization of functional requirements are given below:

|  |  |  |
| --- | --- | --- |
| **Requirement ID** | **Name** | **Priotization** |
| FR1 | Registration | High |
| FR2 | Login | High |
| FR3 | Change Password | High |
| FR4 | Navigate Menu | High |
| FR5 | Select Items | High |
| FR6 | Customize Options | High |
| FR7 | Add Item | High |
| FR8 | Review | High |
| FR9 | Delete | Medium |
| FR10 | Order | High |
| FR11 | Delivery and Payment | High |
| FR12 | Confirmation | High |

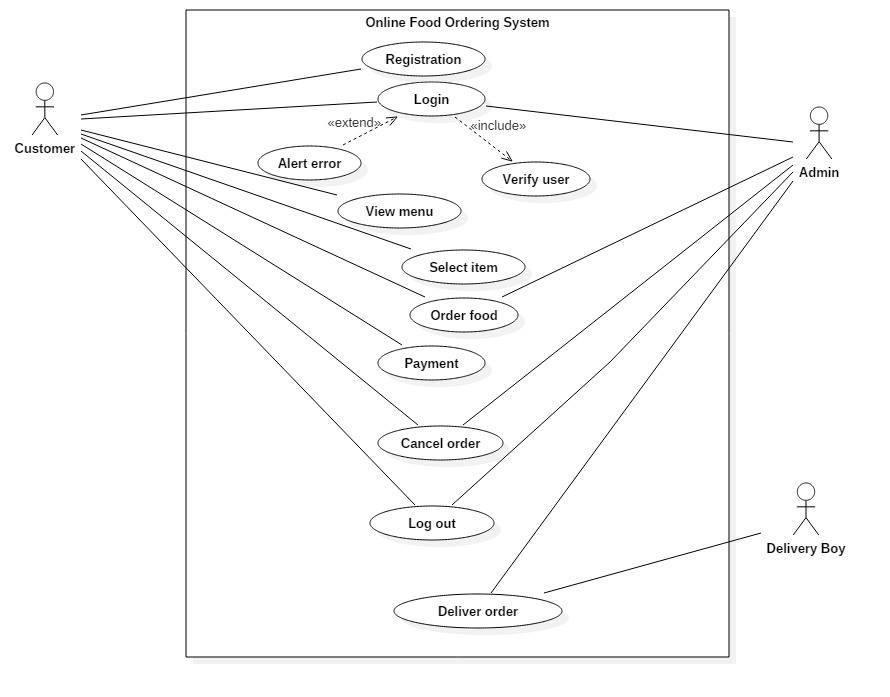
|  |  |  |
| --- | --- | --- |
| **Requirement ID** | **Name** | **Remarks** |
| NFR1 | Scalability | High |
| NFR2 | Performance | High |
| NFR3 | Reliability | Medium |
| NFR4 | Availability | Medium |
| NFR5 | Maintainability | Medium |
| NFR6 | Security | High |
| NFR7 | Environmental | Medium |
| NFR8 | Usability | Medium |
| NFR9 | Legal | Medium |

**Hardware and Software Specifications:**

Here are the specifications required of hardware and software.

|  |  |
| --- | --- |
| Hardware Specification | Software Specification |
| RAM: Minimum 1GB | OS: Windows/MAC OS/ IOS/ LINUX/ ANDROID |
| Storage: Minimum 1GB | Browser: Chrome, Safari, Mozilla, Opera, Microsoft Edge |
| Processor: 1.5 Ghz | Adobe Flash Player |

**2.6 Use Case Diagram:**

This area contains enlightening data about movement proposed in this framework. The framework comprises of partner who are those individuals or things who controls conduct of the framework. An utilization case graph speaks to how framework reacts to demand from any partner. It shows set of conceivable association among clients and frameworks for certain condition to perform specific errand. Use case graph in proposed framework are recorded beneath.

NLA:

Background of the Project:

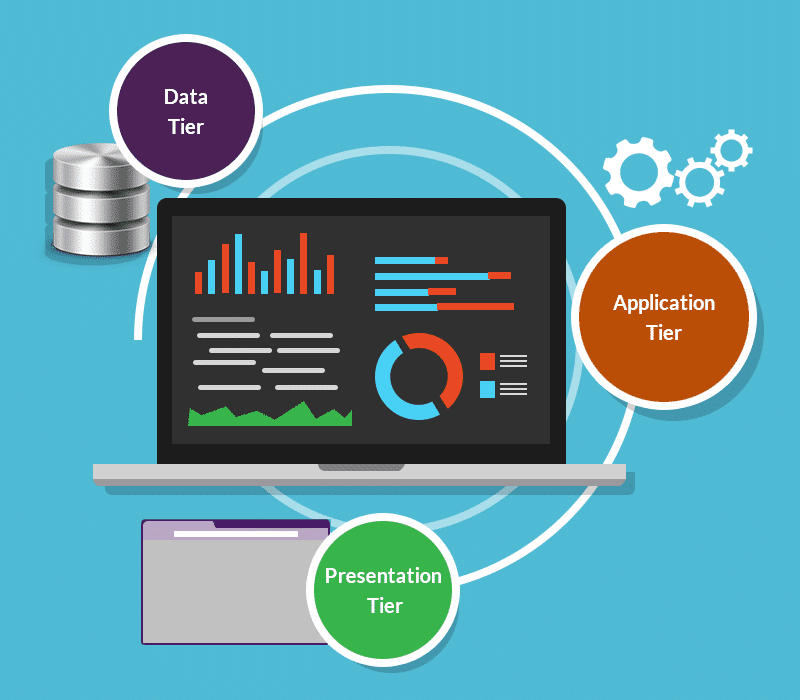
Online Food Ordering System (OFOS) is an application based on web that enables users to order food items online. It is an easy access for the individual who are busy as it provides convenience for them. This website aims to provide user to order the food items and deliver on time.

Here, user firstly, register their account and after logging in they can order the food items they want to eat. To view the items on the website, the users can see the products without logging in. But if they want food to be delivered to them, they must register first by creating an account. The customers will have to pick whether they need the nourishment to be conveyed to them or it will be bundled to get and the installment technique will be upon conveyance or get. The customer can change their password as well after they have been logged in and can review their item. There will be a system administrator who will reserve the privilege to include and oversee customer accounts, and managing item and requests and to wrap things up a dinner deliverer will's identity dealing specifically with pending conveyances. The client will be in a situation to see the products, register and submit a request. There will be an affirmation receipt for every single request made by the client which can be printed.

|  |  |  |
| --- | --- | --- |
| **Noun** | **Verb** | **Adjective** |
| Admin  Customer  Payment  Order  Food Item  Menu  Password | Review items  Add items  Delete items  Deliver food items | Name  Address  Phone number  amount |

**2.7 System Architecture:**

3 Tier Architecture:

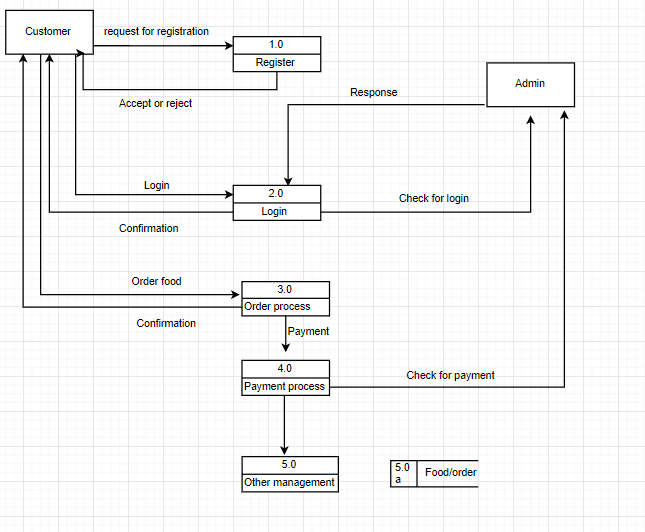


**1. Presentation Tier:** Presentation layer lies at the top most level which displays a user interface. It provides a secure platform for users to interact with business tier. Consider a model, a customer fills login structure in level 1. The structure regards are submitted to second level which checks database regards to perform exercises.

**2. Application Tier:** The mid-layer is application layer which procedure directions, settle on legitimate choice and perform arrangement of assessments. This layer is in charge of a large portion of preparing errands. In past model, if numerous client submit structure at same time, the business layer need to decide for which client to allow first and whom finally. Without this database layer is straightforwardly gotten to that builds burden to server.

**3. Information Tier:** The lower layer is information layer which stores and recover information from database. Data is passed to rationale layer and after that to introduction to show data. Consider in above model, when client enters right information, it is prepared checked and result is passed. At long last, for progress client is taken to dashboard or blunder message is appeared.

DFD(Data Flow Diagram):



Class Diagram:

