Chapter :2

# Analysis

## Introduction Analysis

Analysis can be defined as a thorough investigation of any data or information by breaking data into different components in order to understand its nature and essential features. The requirements and objectives of the system along with the process of accomplishing it are defined in this phase. It is done to discover vulnerabilities before the development phase which are easier to correct than the ones found during the testing phase. In this phase, the requirements of the users are taken into consideration and ensured whether the system will meet their expectations or not. Problems regarding the requirements and resources along with their solutions are provided and gathered information are submitted to achieve the result project.   
  
Its importance are as follows:

* It can be used to perform security audits.
* It gives idea about the necessity and situation about the business.
* It helps in decisions making and problem solving.
* Acts as the main base for Design and Implementation.

## Analysis methodology

Analysis methodology is a process that provides guidance to analyse. There are many different types of methodology one can use to analyse a system. Some of the famous methodologies are as follows:

**SSM (Soft System Methodology)**Soft System methodology is the analysis methodology I choose for this project.  
SSM is a tool that is used for investigating system requirements. It is more people-focused analysis and is more concerned with user interaction.   
The benefits of using SSM are as follows;

* Improves project management and control
* Encourages good communication between those who are involved in the project.

Several stages should be followed while undertaking SSM, they are as:

1. Rich pictures
2. Root definition
3. Conceptual diagram

1. Rich picture  
A rich picture is an illustration of a situation that shows the main elements and relationships that need to be considered in trying to intervene in order to create some improvement. (Research, 2012) It shows the way to express situation through diagrams to create an introductory mental model.  
A close up of text on a whiteboard

Description automatically generated

Figure 1 rich picture user

A close up of text on a whiteboard

Description automatically generated

Figure 2 rich picture admin

**2. Root Definition:**   
Root definition is a statement that is used to describe the system processes and its aims and purposes. It describes the functions and problems of the potential system that is going to be developed. CATWOE analysis helps in proper formulation of a Root definition.  
  
**CATWOE**  
CATWOE is the generic techniques that is used to identify and categorize the problems and their solutions, and the people involved in it. It analyses every possible stakeholder, business processes and external matter. It considers the needs of the users to enhance productivity and quality of the product. CATWOE stands for:  
**Customers/Clients:**   
They are the character that benefits or suffers from the outputs of the business organization.

**Actors/Agents**:  
They are the ones that are involved in the situation and its resolution. Actors will play different roles in the scenario like sometimes they are employees, vendors or government officials.

**Transformations:**Transformations shows the process of the development of the system and their effects on data.

**World View:**   
This focuses on showing the project influence from bigger area of scope.

**Owners:**   
Owners are the ones that own the organization or the problems.

**Environment:**   
Environment mostly considers outside factors like legal, economic, community, legal factors and their associated constraints and limitations.

For my project,

* **Customers** are the users that will use our system.
* **Actors** of this system are admin and users.
* **Transformations** of this system are the products that are to be sold online.
* **World view**
* **Owner** of this system are the investors and developers.
* **Environment**

Advantages of using CATWOE:

* Since every stakeholder are involved more opinions regarding different topic will be encouraged which will cover wider scope.
* There is will user participation which will cause more commitment of the users for the project.
* It encourages join problem solving.
* It is a problem-solving checklist.

1. **Conceptual Model:**

Conceptual model is the representation of the potential system. It shows how the system should function and the activities that are needed for the process to take place.  
The conceptual model is as below;  
A close up of text on a white background

Description automatically generated

Figure 3 conceptual diagram admin

A close up of text on a white background

Description automatically generated

Figure 4 conceptual diagram user

**Reasons why I choose soft over other analysis methodology:**

The reasons I choose soft methodology over other methodology are:

* Soft approach encourages stakeholder’s participation
* It provides an overview of a system
* It covers wide area of scope.
* It considers the human factors in an organization.  
  1. Feasibility study   
     Feasibility study is the analysis that covers factors like economic, legal, time, cost, etc to ensure that the project will complete successfully. It helps to differentiate the pros and cons of undertaking the project before devoting their efforts into it.  
     The main motive of a feasibility study is to recognize all features of a project and become alert of any possible problems that can occur while executing the project. After considering all significant factors it lets the developers know if the project is worth undertaking.  
     The feasibility tests that I have performed in this task are:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Types of Feasibility Study** | **Questions answered by the study** | **Relation with the project** |
|  | Operational Feasibility | How will the proposed system solve the problem and takes advantages of the opportunities. | It will solve the problem of online shopping of imported good and will provide both users and seller with advantages. |
|  | Legal Feasibility | Does it meet the legal and ethical requirements? | This project follows all the rules that are provided by the education and government system. |
|  | Schedule Feasibility | Is the time enough to complete the project? | The Gantt chart will provide us help with the schedule of the project. |
|  | Resource Feasibility | Whether the resources, technology and skill available at the current moment is enough to complete this project? | The resources that this project requires are sufficient to develop this project. |
|  | Marketing Feasibility | How will it perform when it gets implemented? Who are the target users? | This project has a lot of opportunities in the market and will be successful with proper marketing strategy. |

* 1. Requirement Analysis:  
     Requirement analysis contains the tasks that are required to meets the needs of the project. It shows the requirement of the project and the description of it to identify its clear motive. It assists to keep the requirements in line with the need of the business.  
     The requirements of the project that are to analyse are explained below by dividing it into two parts i.e. Functional and Non-Functional Requirements.

### Functional Requirement:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Rational** | **Dependencies** |
| FR001 | Registration | Sign up for users and admin. | To give login access to the users. | NONE |
| FR002 | Sign in | Sign in to let users have access in the system. | To give authorized users access to their account | FR001 |
| FR003 | Sign out | Users can logout from their account. | To | FR002 |
| FR004 | Delete account | To delete their account. | To get rid of their account. | FR001 |
| FR005 | Edit profile | Users can update their account. | To update data recorded. | FR001 |
| FR006 | Add comments | Users can state their opinions. | To share personal opinions to let other users know. | FR001 |
| FR007 | Delete comments | Users and admins can delete the comments. | To remove comments that no longer are supported. | FR006 |
| FR008 | Update price | Admins can update the price of the products. | To let customer, know about the change in the price. | NONE |
| FR009 | Event Update | Admins can update the events when new events happen. | To let users know about the events that will take place. | FR0011 |
| FR010 | Booking | Users can book supplies. | To make purchase. | FR002 |
| FR011 | Admin dashboard | Admin can view users’ detail and make changes. | To keep the website updated and under control. | FR002 |
| FR012 | Forgot password | Users can make new password when they forget their old one. | To get access to their account even after forgetting the password. | FR001 |
| FR013 | Booking cancel | Users can cancel their booking. | To cancel the purchase process. | FR010 |
| FR014 | Search Engine | Find the products that the users want to know. | Saves time of users. | None |
| FR015 | Add to basket | Space where users can keep their products before making purchase. | To make it easier for users to make purchase. | FR002 |
| FR016 | Online Chat | Online chat message feature to discuss about features. | To initiate communication. | FR002 |
| FR017 | Subscribe to newsletter | To get notification from the website. | To be up to date with the website. | FR001 |
| FR018 | Favourite products. | Keep favourite products in separate place. | To save products for future purpose for the users. | FR002 |
| FR019 | Link for other social media. | Let’s users get in contact with other social media. | To make the website more user friendly. | FR001 |
| FR020 | Fan vote | To support the artists or the products. | Upvotes the artists or the products. | FR002 |

* + 1. Non -Functional Requirement

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Title** | **Description** | **Rational** | **Dependencies** |
| NFR001 | Security | To provide users with their personal data security and gain trust. | To avoid security issues. | FR001 FR002 |
| NFR002 | Sound effect | Sound effect that’s will occur while performing tasks. | To enhance user’s experience while using the website. | N/A |
| NFR003 | Performance | Website should be fast and up to point. | To make the website easy to use and have an overall good experience. | N/A |
| NFR004 | Availability | Website should be available at all kinds of circumstances. | To be available for users to use it. | N/A |
| NFR005 | Scalability | The greater number of users the harder it is to maintain the system. | Scalability ensures the growth of the number of using without affecting the overall performance of the system. | N/A |
| NFR006 | Maintainability | In maintenance bugs are cleared and if needed changes are made. | To ensure sustainable running of the system. | N/A |
| NFR007 | Reliability | Accurate data is to be delivered to gain confidence to the system. | To create environment of trust between users and developers. | N/A |
| NFR008 | Useability | System should be easy to use and understand. | To attract more range of users and make them feel comfortable. | N/A |
| NFR009 | User friendly | All kinds of users should be able to use the system. | To create equality and demote discrimination among users. And to attract more group of users. | N/A |
| NFR010 | Legal | Every system should be recognized by the law or rules. | To be safe and avoid any backlash from the government. | N/A |

* + 1. MoSCOW Prioritisation  
       MoSCOW Prioritisation is a technique that is used to classify the requirements based on their importance.   
       The term MoSCow is an acronym of (Must have, Should have, Could have and Won’t have). They are further explained below:   
       **Must have**: This category consists of the most important requirements that is mandatory to be completed.   
       **Should have**: This category consists of the requirements that are not vital but if included they will add significant value.   
       **Could have**: This category consists of requirements that have smaller impact on the outcome.  
       **Would have**: This category consists of the requirements that are not included in the current version but will take place sometime in future.

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Requirements** | **MOSCOW** |
| FR001 | Registration | Must have |
| FR002 | Sign in | Must have |
| FR003 | Sign out | Should have |
| FR004 | Delete account | Could have |
| FR005 | Edit profile | Could have |
| FR006 | Add comments | Must have |
| FR007 | Delete comments | Should have |
| FR008 | Update price | Should have |
| FR009 | Event update | Should have |
| FR010 | Booking | Must have |
| FR011 | Admin dashboard | Could have |
| FR012 | Forgot password | Could have |
| FR013 | Booking cancel | Should have |
| FR014 | Search Engine | Must have |
| FR015 | Add to basket | Must have |
| FR016 | Online Chat | Must have |
| FR017 | Subscribe to newsletter | Should have |
| FR018 | Favourite products. | Should have |
| FR019 | Link for other social media. | Must have |
| FR020 | Fan vote | Could have |

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Requirements** | **MOSCOW** |
| NFR001 | Security | Would have |
| NFR002 | Sound effect | Could have |
| NFR003 | Performance | Should have |
| NFR004 | Availability | Should have |
| NFR005 | Scalability | Could have |
| NFR006 | Maintainability | Could have |
| NFR007 | Reliability | Must have |
| NFR008 | Useability | Must have |
| NFR009 | User friendly | Should have |
| NFR010 | Legal | Could have |

* + 1. SRS (Software and hardware requirements)  
       SRS is the description of a software system that is going to be developed. It shows the lists of sufficient and necessary requirements for the project development and implementation. It is a way to establish an agreement between suppliers and customers on how the project should function.   
         
       The system that I will use while performing the project are as follow:  
       Pre SRS

|  |  |
| --- | --- |
| Hardware Requirements | Software Requirements |
| RAM minimum 8 GB | Application: Xampp |
| Hard Disk space 1TB | Google chrome Mozilla firefox Internet Explorer |
| Processor 2.5 Gigahertz processor or higher | Front End: JavaScript, PHP(OOP), CSS & HTML |
| Screen Resolution 1024\*768 or higher resolution monitor | Back End: My SQL |

Post SRS:

|  |  |
| --- | --- |
| Hardware Requirements | Software Requirements |
| RAM minimum 2 GB | Linux windows 7 or above  Mac compatible |
| IOS  Android |  |
| Hard disk space 100GB |  |

* 1. Use Case Diagram  
     Use Case diagram is a graphical representation of the interactions among the elements of the system. It helps to identify and organize both internal and external factors that will inspire the system. Use case diagrams are employed in UML (Unified Modeling Language). They separate the system into actors and use cases. The use case contains all system activities that have significance to the users within a given system.   
     The use case for our project’s system is provided below:

A close up of a map

Description automatically generated

Figure 5 use case diagram admin

**Scenario Description:**  
Actor: Admin  
Admin can perform the following activities:

* Admin can sign in and sign out
* Admin have the power to control events. They can perform CRUD Operation.
* Admin can manipulate user’s account and can delete users.
* Admin can add and update the products and its price.
* Admin can view, add or delete comments of users.

A picture containing text, map

Description automatically generated

Figure 6 use case diagram user

**Scenario Description:**

Actor: Unauthorized users  
Unauthorized users can perform the following activities:

* They can sign up

Actor: Authorized users  
Authorized users can perform the following activities:

* They can sign in and sign out
* They can make or cancel their purchase.
* They can view blog and events.
* They can maintain their personal account. They can perform CRUD operation.
* They can vote
* They can add and delete their comments.
* They can keep their favourite products to separate folder.
  1. NLA and initial class diagram  
     K-Merchandise is a web-based application system which is created to provide online shopping and blog service to its users. It contains products that come from abroad and is established to serve customers with best quality product. The main aim of the system is to promote online shopping for its users.   
       
     This web application should consist of two users: Admin and users. Admin and users should provide their personal information to get registered to the system. Personal information includes username, email, password, address and phone number. After they get registered to the system, they can get access to different services based on their access right. Admin can update their profile and have the right to delete the user’s account. They can add, edit or delete events and products and label the product’s price. The users can view products, make comments or delete them.  
     The system involves the service of voting for their artists or products so that they can recommend it to other users. The users can favourite their products which will keep them into a different secured place so that they can view it in future. When the users make a purchase, they must add in the cart to make a purchase. All this information are to be stored in a database. The payment system is based on delivery on cash. Users can cancel the purchase before the delivery date.   
     The website should also contain chat feature that will allow users to communicate with each other.

From the above scenario, the Natural Language Analysis is performed below:

While doing NLA the noun, adjectives and verbs are separated.   
Noun= candidate class  
Adjectives= attributes

|  |  |  |
| --- | --- | --- |
| **Noun (Candidate class)** | **Adjectives (Candidate attributes)** | **Verbs(Candidate operators)** |
| Admin, user, product, event, purchase | Username, email, password, phone number, address, price | Edit, delete, add, register, vote |

Verbs= Operation

**Class diagram:**

Class diagram is the blue print that represents the static view of the application. It describes the attributes and operations of a class. It shows a collection of classes, associations, collaborations and constraints. (jackson, 2018)

A screenshot of a cell phone

Description automatically generated

Figure 7 Class diagram