# Chapter 2 – Analysis

Analysis is that the method wherever the character of one thing is studied or the determination of its essential options furthermore as their relations also. Here, in analysis we tend to break additional complicated compounds merely locution the project into the easier or basic ones. It helps North American nation to try and do and complete the project in time. It helps North American nation to assemble the wants and develop the software package within the totally different stage.

Here, I’m developing a Cargo Management System that a web-based application which

mainly targeted for the import and export. To develop I’ve got gathered some data with the

help of form, interviews, perceptive the market, etc. this may facilitate Maine to understand the

requirement to develop the software package.

## Feasibility Study

Feasibility study is that style of study that is usually done by developers which inaugurates whether or not the conditions are correct to start out a selected project. it's typically done to appear the whether or not the new hardware and software system are possible or not. it's done to specialize in the project and description the alternatives.

For this project, we've got studied 3 elements of the practicableness study and that they are listed and explained below:

### Economical

cargo plays in economic development and factors that condition its impact. After presenting basic empirical relationships between air cargo and both trade and gross domestic product per capita, we discuss three factors that can enhance cargo’s positive impact: air service liberalization, improving customs quality, and reducing corruption.

### Financial

We offer easy digital trade finance solutions to assist you along with your capital wants. A simplified answer which will handle each the flow of your merchandise and funding at a competitive value to assist you expand and develop your business.

### Legal

From the study, I found that the code will maintain the safety. It helps the shopper or the owner to depend on the code. Also, it doesn't have any privacy problems. this method is often employed by every kind of the users within the outlets where they need. they will allow it by collateral from the upper authorities.

## Analysis Methodology

For this project, I’ve got chosen the Soft System Methodology (SSM) analysis as a result of its additional people-focused analysis. it'll facilitate Pine Tree State to assemble necessities. It tries to understand and learn the matter scenario between the stakeholders instead of resolution the pre-defined issues (Anon., n.d.).

I have chosen this technique as a result of this helps America to focus additional on the peoples read rather than that specialize in system. In my read the system ought to be developed as per the purchasers would like as they are the one WHO are mistreatment the system right along. we should always recognize their read which is able to facilitate America gather additional demand for the system throughout analysis. I’ve got followed the 3 steps of the SSM likewise as done the CATWOE analysis. the steps of the SSM that I’ve got followed are listed and explained below.

### Rich Picture

It is a method of exploring, acknowledging and process a given scenario yet as expressing with the assistance of diagrams to ascertain a preliminary mental model. It provides North American country plan on the way to open discussion.

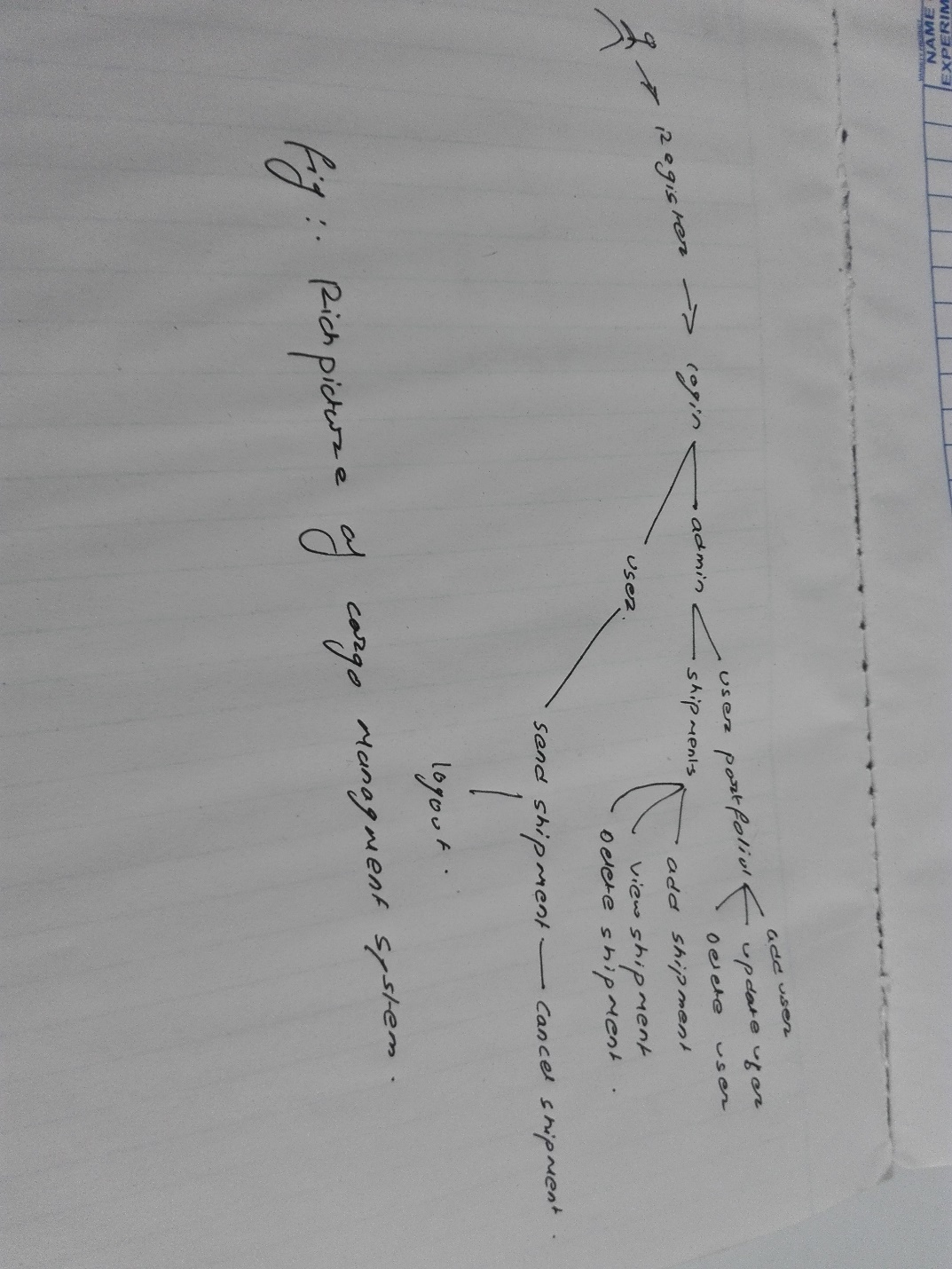


Figure : Rich picture

### Root Definition

Here the fundamental motivation behind the task is to assist the little business visionaries with having the online application which will help transports. My venture can be equipped for bringing in and sending out the merchandise, overseeing distinctive information. It very well may be utilized in long haul as it is a little speculation venture which can be sued from little scale business to extensive scale. It will spare the d information of the customers as long as they need in the database.

### Conceptual Model

After drawing the rich picture and defining the root definition, I have followed the triple E if the conceptual model to know the systems performance. They are explained below

* Efficacy:

The system can be major transformation in small scale business. It can help. Import and exports the goods in time.

* Efficiency:

As it is a web base application it needs internet to run. This software can run with minimum resources as it is a simple software.

* Effectiveness:

The framework can hold information as long as the proprietor of the framework need. They can use for the long time yet I propose them to keep up the framework by overhauling time for the better outcome and execution.

## Software requirement Specification

It is a nitty gritty portrayal of a product framework to be created with its utilitarian and non-practical necessities. The SRS is created based the understanding among client and contractual workers. It might incorporate the utilization instances of how client will collaborate with programming framework. The product prerequisite determination archive steady of every single vital necessity required for venture improvement. To build up the product framework, we ought to have clear comprehension of Software framework. To accomplish this, we have to ceaseless correspondence with clients to assemble all necessities.

## Hardware requirements Specification

Operating System: Windows 7 or higher

RAM: 2GB or higher

Processor: 1 GHz or higher

Screen resolution: 1024 \* 768 resolution for better experience

## Functional Requirements

It is the announcement of expected capacity of a framework and its segment. It can identify with equipment, programming or both regarding computation, specialized subtleties, information control and handling or other explicit usefulness that characterizes what a framework should achieved.

Below is the function requirement which of my project

ID: FUN1

Title user Registration

Description: User should register with valid information

Rational: It helps the user to acquire log in credentials

Dependencies: N/A

ID: FUN2

Title: Login

Description: User can login with the right username and password

Rational: It helps the user to enter the system.

Dependencies: N/A

ID: FUN3

Title: Forgot Password

Description: User can change password.

Rational: It helps user to change password when they forget it.

Dependencies: N/A

ID: FUN4

Title: Add, update, delete data

Description: User can use the CRUD function

Rational: It helps them to manipulate the data.

Dependencies: FUN1

ID: FUN5

Title: View report

Description: User can view the debits and credit

Rational: It helps the user to keep the track of profit and loss

Dependencies: FUN1

ID: FUN6

Title: Logout

Description: User can logout from the system.

Rational: It helps to user to logout when they have used the system and helps in the security reason.

Dependencies: FUN1

ID: FUN7

Title: Internet for booking/reservation

Description: Customers can reserve tables in the café and restaurants.

Rational: Reserving the table in advance

Dependencies: N/A

## Non-Functional Requirements

In non-utilitarian testing the quality attributes of the part or framework is tried. Non-useful alludes to parts of the product that may not be identified with a particular capacity or client activity, for example, adaptability or security. E.g. What number of individuals can sign in without a moment's delay? Non-utilitarian testing is likewise performed at all dimensions like useful testing.

ID: NFUN1

Title: Performance

Description: The system should run fast without any lag.

Rational: Productivity maintained

Dependencies: N/A

ID: NFUN2

Title: Usability

Title: The software should be user friendly.

Rational: User will not feel monitors.

Dependencies: N/A

ID: NFUN3

Title: Response time

Description: The software should be quick to response to user.

Rational: rapid responding.

Dependencies: N/A

ID: NFUN4

Title: Security

Description: User with the right password and username can access data.

Rational; Data confidentiality

Dependencies: N/A

ID: NFUN5

Title: Reliability

Description: The system should be accurate and reliable.

Rational: Maintenance of reliability:

Dependencies: N/A

ID: NFUN6

Title: Availability

Description: Software should run whenever the user wants.

Rational; Upholding the availability

Dependencies: N/A

ID: NFUN7

Title: Maintainability

Description: the software can be maintaining in the future.

Rational: upgrading the system will have better experience

Dependencies: N/A

ID: NFUN8

Title: Recoverability

Description: There should be backup plan to recover the data in case there is any problem.

Rational: Data availability

Dependencies: N/A

Prioritization

Functional Requirement

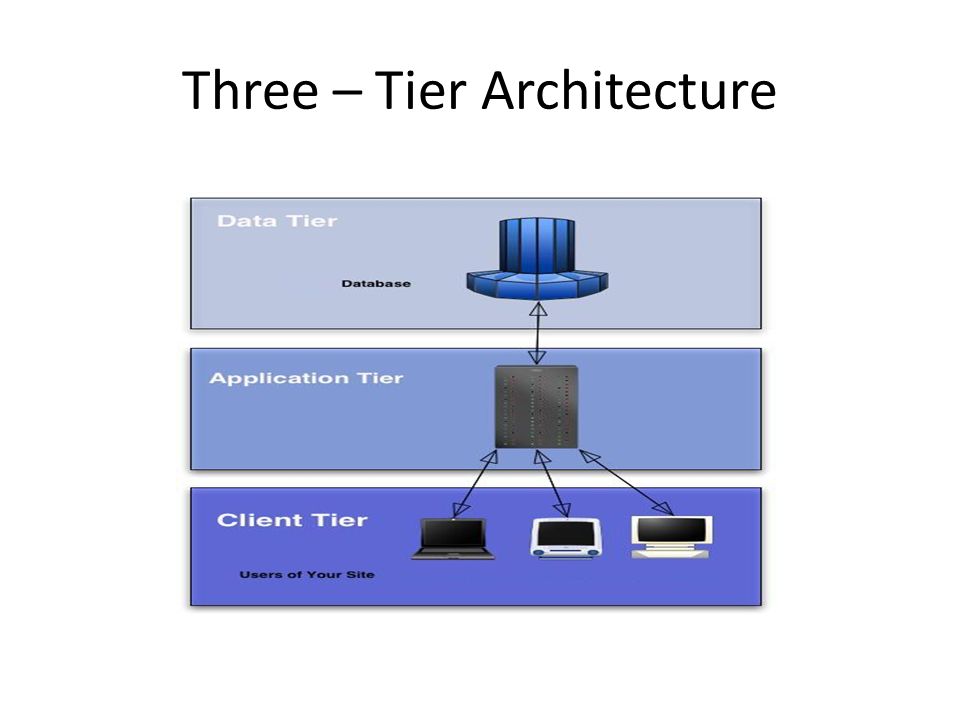
|  |  |  |
| --- | --- | --- |
| ID | Functional Requirement | MoSCoW |
| FUN1 | Registration | Must Have |
| FUN2 | Login | Must Have |
| FUN3 | Forgot Password | Could Have |
| FUN4 | Add, Update, Delete | Must Have |
| FUN5 | View Report | Should have |
| FUN6 | Logout | Must Have |
| FUN7 | Internet For Booking | Wont Have |

Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| ID | Non-Functional requirements | MoSCoW |
| NFUN1 | Performance | Could Have |
| NFUN2 | Usability | Could Have |
| NFUN3 | Response Time | Could Have |
| NFUN4 | Security | Should have |
| NFUN5 | Reliability | Should have |
| NFUN6 | Availability | Could have |
| NFUN7 | Maintainability | Could have |
| NFUN8 | Recoverability | Could Have |

## Architecture

Software architecture is the characterizing and organizing of an answer that meets specialized and operational necessities. Programming engineering improves traits including a progression of choices, for example, security, execution and reasonability. These choices eventually sway application quality, support, execution and by and large achievement.



A three-tier architecture is a client-server architecture in which the functional process logic, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms.

## Use Case Diagram

A use case is a product and framework designing term that portrays how a client utilizes a framework to achieve a specific objective. A utilization case goes about as a product displaying procedure that characterizes the highlights to be executed and the goals of any mistakes that might be experienced.

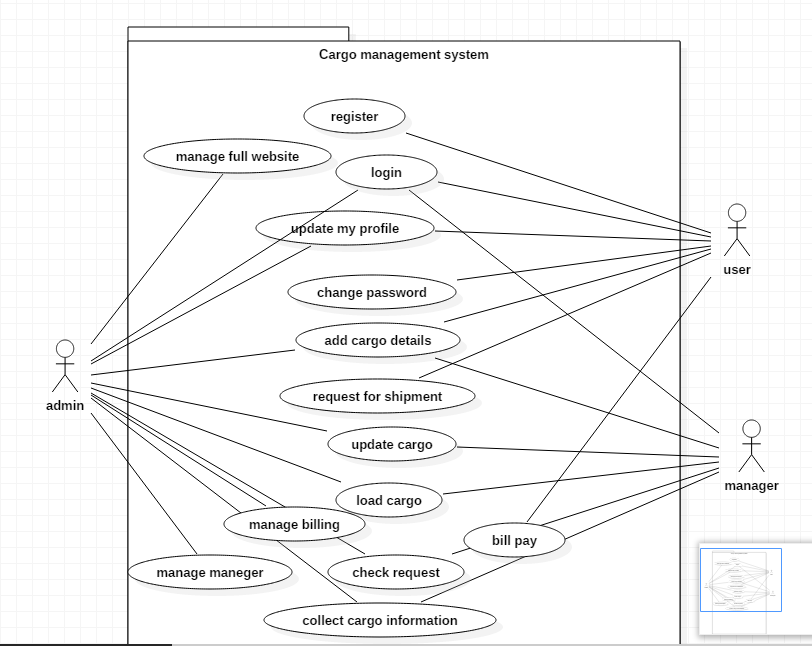


Figure :use case Diagram