



**NAME: MBAH ROMARICK FORKWA**

**COURSE: LARGE SYSTEMS ENVIRONMENT.**

**MATRICULE: ICTU1040576**

**END OF SEMESTER EXAM PROJECT**

**Football Stadium Ticket Booking System**

09.02.2021

## Table of Contents

1. Introduction-----	3
1.1 Problems faced with the existing system-----	3
1.2 Proposed Solution-----	4
1.3 Objectives of new System-----	4
1.4 Requirements-----	5
1.41 Functional Requirements-----	5
1.42 Non Functional Requirements-----	5
2. Project Estimates-----	7
2.1 Function Point (FP) -----	8
2.2 Lines of Codes (LOC) -----	10
3. Project Resource Plan-----	
27	
3.1 People-----	27
3.2 Hardware and Software-----	28
4. Schedules-----	28
4.1 Work Breakdown Structure-----	29
4.2 Pert Chart -----	29
5. Risk Management Plan-----	30

## Introduction


Stadium is a centerpiece for the hosting of major events especially sport events such as football or in other words, a stadium can be seen as a place where people go to watch matches like football, cricket, hockey and more. It is one thing to have a stadium, and it's another thing to manage it. Ticket acquisition is a vital aspect of stadium management. The system can also be considered to be a critical one, as the failure of it would result in significant financial loss for the company managing the stadium, and, in the case of the software failing, potentially thousands of angry fans stuck outside the ground, many of whom may decide not to return – against this would have implications for the managers of the stadium, and probably the software developers too. Therefore, the system must be created using well established principles of software engineering that guarantee a high degree of reliability. The main aims of this research are to create a reliable and easy to use system which will simplify the purchasing of tickets.

### **Problems faced with existing system (manual system for booking).**

Stadium managers have the problem of managing the booking of seats for a game or group of games to be witnessed by football fans.

The manual system of purchasing tickets may not be effective considering the fact that counterfeits can be produced and also sold to unsuspecting spectators.

The security issues and financial challenge involved in the management of the identity of people and finances in terms of ascertaining how many tickets were bought for each seat class in a football stadium is enormous. Most stadiums lack an effective



computerized management system that will foster the easy administration of the service.

The manual system of booking tickets for matches can be very slow due to the fact that they may be many individuals trying to book at one point in time which makes it inconvenient for most individuals.

At times, this system may be very inconvenient for individuals who stay far away from the stadium and may be too occupied at work, so they end up not getting a ticket for an upcoming match.

### **Proposed Solution (An online stadium ticket booking system)**

With our online stadium booking system, we will try to eliminate or limit the use of the manual stadium ticket booking system which involves individuals moving to stadiums and standing on queues in order for them to purchase a ticket. So here, we will provide a platform which will permit individuals to be able or capable to purchase their tickets from anywhere there are either from their mobile devices or from the web.

### **Objectives of new system**

The aim of this project is to develop a football stadium ticket booking system that will help in the recording and managing of financial income and booking of seats in the stadium for each registered game or event. The following are the objectives of the study: To develop an automated system for ticket acquisition for all football stadiums within the National territory (Cameron) and also internationally (other less developed countries) to facilitate easy booking and management of ticket based on available seats to provide a secure system for issuing tickets and

also to develop a system that will help prevent overcrowding of the stadiums.

## Requirements

### A. Functional Requirements


1. The system should permit user's book a seat.
2. Users are capable of making payments through the system for seat booked.
3. The system should allow users scheduled matches (administrative function).
4. Users should have an account on the system.
5. Each account supposed to be entitled to a single user.
6. Booking confirmation should be sent to the user through the specified contact (be it email or phone).
- 7.

### B. Non Functional Requirements

1. Performance:
  - a. The system shall support up to 2000 simultaneous users against the central database at any given time who are trying to book for a seat.
  - b. The system must be able to complete 80% of all transactions within 1 and a half minute.
  - c. The client portion shall require less than 20 MB disk space and 32 MB RAM (for computers) and for mobiles, the client portion shall require less than 20 MB disk space and 1 GB RAM.

### Performance issues

So with respect to performance, our system may actually have issues or crash when the following occurs;

- 
1. When users that are trying to simultaneously trying to book exceed the maximum limit (2000).
  2. In case of network issues, the completion time for each transaction may exceed the normal or default time of the system which is 1 and half minute.
  3. In systems with low memory and disk space (say systems less than 20 MB and less than 32 MB RAM) may find difficulties loading the site, or may realize the time to complete transactions becomes exceedingly long.

## **2. Security:**

The system provides users with a secured payment method with respect to the available payments gateways like mobile money, orange money, bank transfer and so on. System will not be able to save the payment details for the users.

## **3. Quality:**

- a. The desktop user-interface shall be Windows 98/2000 complaint and for the mobile devices, the user-interface shall be android version 4.0 complaint.
- b. The user interface of the Online Stadium ticket booking system shall be designed for ease of use and shall be appropriate for a computer literate user community with no additional training on the system.
- c. The Online Stadium ticket booking system shall have a built-in help for the user. Help shall include definitions for terms and acronyms and other system directives.
- d. Mean time between failures shall exceed 300 hours.

## Design and Specification

This part present the design and specification that have been developed for our proposed system as follows

: A customer will be having the certain requirements that must be provided by the system. Hence at the initial stage of the software (website) development, a requirement analysis is performed to identify the needs of the customer. Our software is ticket booking software which can be used in stadium. To make it applicable in any of the stadium, we have taken into consideration the following requirements:

- Each stadium can have any number of classes.
- The price of the ticket is based on the class.
- Each class can have any number of rows.

There are two types of users of the system. One is the customer and the other is the administrator;

The Customer should be able to:

- Register with his username and orange money/mobile money number or credit card number (optional).
- Reserve the ticket.
- Make payment either through mobile money, orange money or credit card.
- Unregister.

The Administrator should be able to

- Register himself by giving his details along with the user name and password
- Login into the system.
- Update his information.
- Change password.
- View seat status.
- Delete the old records.

## Software modelling

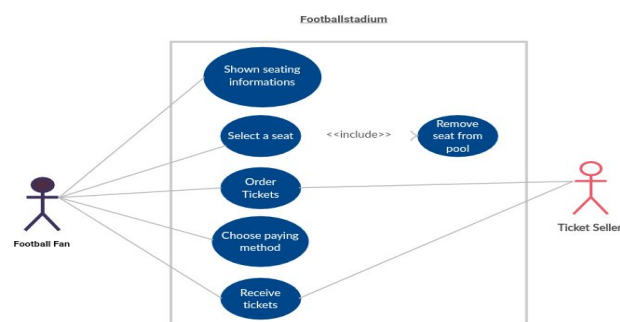
Software modelling is the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application.

Modelling is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and

analysed, system modelling is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system -one that will be difficult to test, one whose quality cannot be assessed until the last stage. During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented.

Use case Diagram

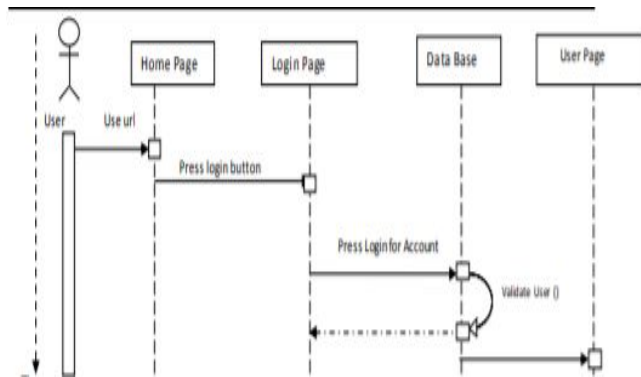


## Sequence Diagrams

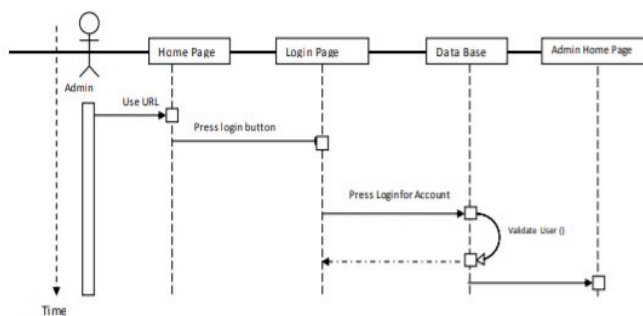
Sequence Diagrams Represent the objects participating the interaction horizontally and time vertically. A Use Case is a kind of behavioral classifier that represents a declaration of an offered behavior. Each use case specifies some behavior, possibly including variants that the subject can perform in collaboration with one or more actors. Use cases define the offered behavior of the subject without reference to its internal structure. These behaviors, involving interactions between the actor and the subject, may result in changes to the state of the subject and communications with its environment. A use case can include possible variations of its basic behavior, including exceptional behavior and error handling.

Sequence diagram 1: user





Sequence Diagram 2: admin



## Project Estimates

In order to effectively develop our OSTR, we need to estimate the cost, effort and duration that will be needed for our project to be completed. With regards to this project, we will use two project estimation techniques which are functions point analysis (FPA) and line of codes (LOC) to measure the size of the software and from there on we can do our estimations.

### Function Points (FP)

FP is methodology to estimate and measure the time and the cost of development and maintenance of the software applications, which was developed by A.J. Albrecht of the IBM Corporation in the early 1980s. Functions in the system can be divided into different categories and complexity weights:

Category

Complexity weight factors

Simple



Average

Complex

Inputs

2

4

6

Outputs

3

5

7

Files

5

10

15

Inquires

2

4

6

Interfaces

4


7

10

Once the total function points calculated, plus several environmental factors, we could get an adjusted function point so that the total FP / men days could be measured.

Function Points (FP)

FP is methodology to estimate and measure the time and the cost of development and maintenance of the software applications, which was developed by A.J. Albrecht of the IBM



Corporation in the early 1980s. Functions in the system can be divided into different categories and complexity weights:

Category

Complexity weight factors

Simple

Average

Complex

Inputs

2

4

6

Outputs

3

5

7

Files

5

10

15

Inquires

2

4

6

Interfaces

4

7

10

Once the total function points calculated, plus several environmental factors, we could get an adjusted function point so that the total FP / men days could be measured.

### Lines of Codes (LOC)

Lines of code (often referred to as Source Lines of Code, SLOC or LOC) is a software metric used to measure the amount of code in a software program. LOC of each module will be estimated by assumption based on historical data, analysis of defined modules and experience. Below table shows the estimated relationship between function complexity and the required cost and effort.

Complexity

LOC per month

Cost per LOC

Simple

350

550 frs

Average

300

1100 frs

Complex

200

1650 frs

External Interface:

This interface is designed for external customers. They can enquire and buy the ticket through online process.

Modules:

Perform Searching

Customer Login

New Member Registration

Reset password



Buy Ticket

Bill Payment

Internal Interface:

This interface is designed for internal staff. They can perform daily administration work and generate report for management review.

Modules:

Staff Login

Customer Maintenance

Generate Report

User Account Maintenance

Perform Searching

Purpose: The new systems have features that are capable to help the customers to search for a stadium base on criteria's like name or location. It presents the main page where the customer can select a stadium and view the different games to be played in that stadium within a year and shows the seats available for which the customer can book.

Function Description:

Category

Description

Complexity

Inputs

A text box will be shown to customers allowing them to respond with a valid answer for performing searching. The following text boxes and buttons will require SDK programs like Visual Studio.

Simple

Outputs

A new webpage will be shown to the customer after searching the result from the database being inquired. The result comprise of showing the customer the stadium searched by name and all other stadiums within that location. Same settings will be generated by SDK like Visual Studio.



Average

Interfaces

Connection to TIES and SEATS is required for getting the reservation information.

Average

Inquiries

The text being inputted from the textbox will be used in query for the database. The input must be verified correctly to avoid data error.

Average

Function subtotal:

Category

Count

Simple

Average

Complex

Inputs

1

Outputs

1

Files

Inquires

1

Interfaces

1

Customer Login

Purpose: Before a customer can buy tickets, the system will require customers to register their profile to the database. Here they will input their user name and passwords for security procedures before they can have access to the Buy Ticket Page.

Function Description:



Category

Description

Complexity

Inputs

Customers will be asked to input their user name and passwords to the textboxes and click submit. Textbox and buttons are generated by SDK such as Visual Studio.

Simple

Inquiries

After customers have inputted their user name and passwords on to the textboxes, the system will use this 2 data as reference to check if it is registered inside the systems database. Data will be verified correctly to avoid error during comparison with the database.

Average

Function subtotal:

Category

Count

Simple

Average

Complex

Inputs

1

Outputs


Files

Inquires

1

Interfaces

New Member Registration



Purpose: A customer must become a registered member before buying tickets. This module provide functions for customer to fill in personal and credit card information, verify customer's input, duplication checking on the database and crate new customer record on to the database.

Function Description:

Category

Description

Complexity

Inputs

A form for customer to input personal information. The necessary text boxes and buttons could be easily generated by SDK like Visual Studio.

Simple

Inquires

Verify customer's input to avoid data type / data range error. Search the database to avoid duplicate registration. Include program logics to verify credit card information. This kind of function is common in many enterprise software.

Simple

Files

A database table for storing customer records.

Simple

Function subtotal:

Category

Count

Simple

Average

Complex

Inputs

1





## Outputs

### Files

1

### Inquires

1

### Interfaces

#### Reset password

Purpose: Enable customer to submit the request of reset password online. This module should automatically generate a new password and send an email to the specific customer's email address.

#### Function Description:

##### Category

##### Description

##### Complexity

##### Inputs

A text box for customers to input their login ID (Same as registered email address). The necessary text boxes and buttons could be easily generated by SDK like Visual Studio.

##### Simple

##### Inquires

Verify customer's input to avoid data type / data range error. Search the database to verify login ID, retrieve customer's account information and also updating the new password. Include program logics to generate new password, generate email contents and sending out by email.

##### Average

##### Function subtotal:

##### Category

##### Complexity weight factors

##### Simple



Average

Complex

Inputs

1

Outputs

Files

Inquires

1

Interfaces

Buy Ticket

Purpose: Customer can select which seat they want to buy. Customers have to select a stadium and a match/game first, and then he can see the available seat and select the one they want to sit. After the selection, there is an information on the selected seat and total amount.

Function Description:

Category

Description

Complexity

Inputs

A stadium seat plan will be shown to customers, with identification on available or being booked. The stadium seat plan can be generated by SDK like Visual Studio, but additional effort in layout accuracy is needed in order to let the customers have a better estimation.

Average

Outputs

After the customers selected a seat, a corresponding box below the seat plan will show the quantity of seat selected and the total amount. Those can be acquired from the Inputs with a pre-defined formula calculation.

Average



Function subtotal:

Category

Count

Simple

Average

Complex

Inputs

1

Outputs

1

Files

Inquires

1

Interfaces

1

Bill Payment

Purpose: After the customers selected the seat, they need to checkout in order to confirm the booking and make payment. The bill page is a summary of all the information, such as selected seat, match schedule, payment amount, method of payment and so on.

Function Description:

Category

Description


Complexity

Outputs

There is several of information to be displayed back to the customers. Once the data is queried from the database, they can be placed in to relevant description fields.

Simple

Files



Once the customers press the confirm button, the system will automatically update the SEATS system, COINS system.

Complex

Inquires

At this page, all information will be queried from the other databases and will be displayed to customers for their review. Thus, here involve more than one database to queries.

Average

Interfaces

Since there is a few query actions and databases update actions, interfaces needed to ensure all databases queried correctly and, after pressing the confirm button, all database are updated accordingly.

Complex

Function subtotal:

Category

Count

Simple

Average

Complex

Inputs

Outputs

1

Files

1


Inquires

1

Interfaces

1

Generate Report



Purpose: The Report page purpose is to provide the staff member to generate a summary report for their daily transaction and confirm how many tickets they have sold online. The summary comprise of all the information, such as selected seats, match schedule, amount and method of payment.

Function Description:

Category

Description

Complexity

Inputs

A text box will be shown to staff users allowing them to respond with answers. The data being input will be used in the query. The following text boxes and buttons will require SDK programs like Visual Studio.

Simple

Outputs

The data will show the result of the staff user requested from the database, this result is considered valid and correct based on the comparison with the stored data.

Average

Inquires

Besides verifying input data, complex program logic and database SQL is required in order to select and calculate large amount of data from several databases for reporting.

Complex

Function subtotal:

Category

Count

Simple

Average

Complex

Inputs



Outputs

1

Files

Inquires

1

Interfaces

1

Unadjusted Function Point

It is calculated by the total sum of multiply the function count and the function point.

Category

Complexity weight factors

Total Points

Simple

Average

Complex

Inputs

2 x 8

4 x 1

6 x 0

20

Outputs

3 x 3

5 x 3

7 x 0

24

Files

5 x 3



10 x 0

15 x 1

30

Inquires

2 x 4

4 x 5

6 x 1

34

Interfaces

4 x 0

7 x 2

10 x 2

34

Unadjusted Function Points

142

Technical Complexity Factor

There are many technical complexity factors can affect the development of the application software. It is rated from zero to five and can be calculated by the formula:

$$TCF = 0.65 + (\text{sum of factors}) / 100$$

Technical Complexity Factor

Weighting

Data Communications

5

Performance

4

Heavily Used Configuration

1



Transaction Rate

4

On-line Data Entry

1

End-User Efficiency

2

On-line Update

1

Complex Processing

3

Reusability

2

Installation Ease

1

Operational Ease

1

Degree of Influence

25

$$TCF = 0.65 + (25 / 100) = 0.9$$

Final Function Point

It is calculated by the formula:  $UFP \times TCF$

$$FP = 142 \times 0.9 = 127.8$$

Therefore, the final function point of the application software is 128 (Round to digit).

Cost, Effort and time estimation

In this section, we will estimate the cost, effort and time scale of the development of STRS based on Total FP and historical data of our software house.

Total FP: 128





Historical data:

Description

Value

Average salary:

(In Cameroon FCFA)

45000 FCFA

Productivity:

10 FP / person-month

Cost, Effort and time estimation

Description

Value

Effort:

(Total FP / Productivity)

128 / 10

=13 person-months (rounded)

Overall cost estimation:

(Average salary X Effort)

45000 x 13

= 585000 FCFA

Cost (money) per function point:

(Overall cost / Total FP)

585000 / 128

= 4570 FCFA (rounded)

Time estimation:

13 months (1 person)

Or

1 month (13 persons)

## Lines of Codes

Here, the Lines of Codes (LOC) are estimated by assumption based on historical data, analysis of defined modules and experience. The overall complexity of each Module is weighted by the number of function it contains, the complexity of particular function and also the effort of building the function.

For example, the function of Customer Login is common on every system therefore the complexity is Simple because we can reuse the code from previous project. The function of Bill Payment should be considered as Complex because it is required to handle interfaces with three existing system and also the security requirement of payment transaction is very high.

Complexity

LOC per month

Cost per LOC

Simple

350

16, 500 frs

Average

300

33, 000 frs

Complex

200

49, 500 frs

Once we got the above information, we can estimate the cost, effort and duration by the technique of LOC using the formula in the following

Item

Formula

Effort (person-month)

LOC / LOC per month

Cost



LOC x Cost per LOC

A table of LOC components can be generated by the above information and formulas:

Module

Complexity

LOC

LOC per month

Effort

Cost per LOC

Cost

Perform Searching

Average

432

300

1.44

33, 000 frs

47, 520 frs

Customer Login

Simple

144

350

0.41

16, 500 frs

6, 765

New Member Registration

Simple

216

350



0.61

16, 500 frs

10, 065 frs

Reset password

Simple

144

350

0.41

16, 500 frs

6, 765 frs

Buy Ticket

Average

480

300

1.60

33, 000 frs

52, 800 frs

Bill Payment

Complex

768

200

3.84

49, 500 frs

190, 080 frs

Generate Report

Average

552



300

2.76

33, 000 frs

91, 080 frs

Total

3408

13.26

405, 075 frs

In conclusion, the STRS can be implemented with a budget of 404, 075 FCFA with 13 person-months which simply means this project can be completed with 13 persons in one month or 1 software engineer in 13 months.

## Project Resources Plan

In order for this project to be effectively execute, a wide range of resources are needed which may include people, hardware, software and other special resources.

a. People;

According to our project estimate, our project can be implemented by 13 engineers within a month or by 1 engineer with several skill sets within 13 months. Getting an engineer with a single skill set to perform all the task required to develop the software may be difficult and time consuming. So we will go for the case of using 13 engineers. Amongst them, we will need designers, analyst, coders and a general oversee who ensures the project is implemented according to plan. For the project to be completed within the time frame, all this individuals need to be full time dedicated to the project.

b. Software Requirements:

Operating system: Microsoft Windows XP or later.

Microsoft Visual Studio 2012 Ultimate.

SQL Server 2012.

c. Hardware requirements:



Processor Pentium 4 or higher.

RAM- 1GB or higher.

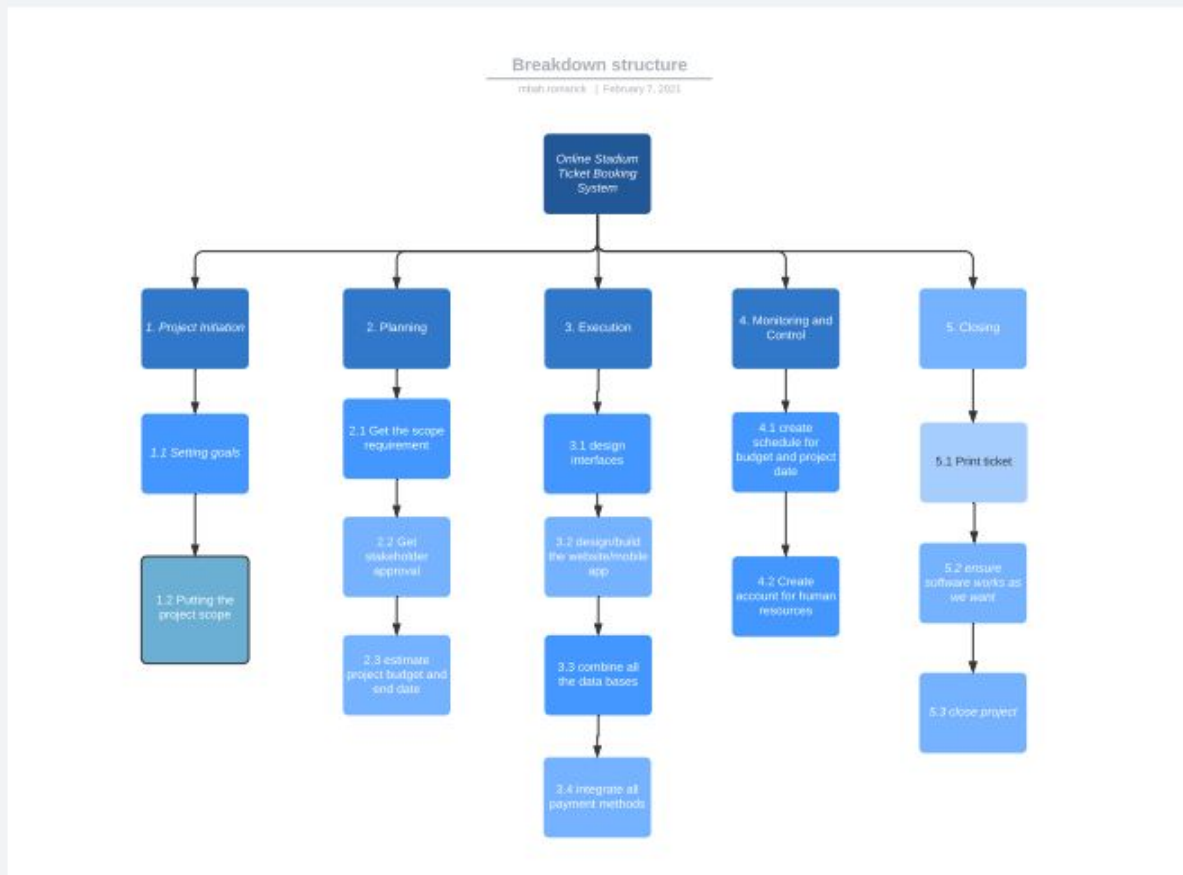
HDD-10GB or larger.

Mobile phone to receive the SMS.

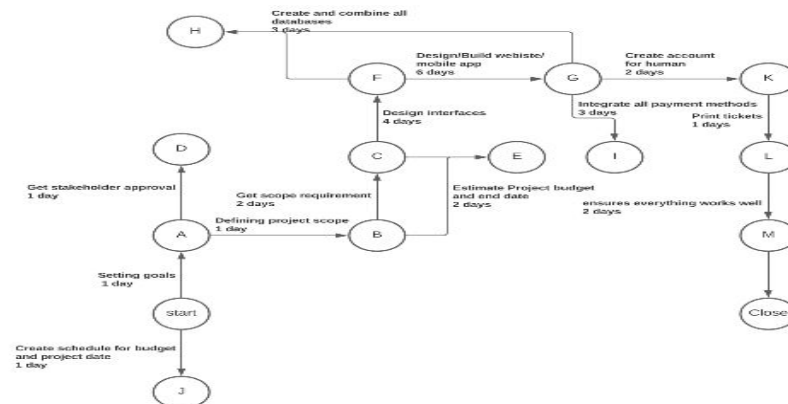
- d. Other Resources: other resources may include API's used for different payments like orange/mtn mobile money, bank transfers and crypto like bitcoin, Ethereum and so on.

## SCHEDULES.

**Work Breakdown Structure:** A work breakdown structure is a productivity technique used to breakdown work into smaller parts so that it can be easily manageable and approachable.



## PERT CHART



Activity	Description	Predecessor	Expected time in days
A	Setting goals		1
B	Defining project scope	A	1
C	Get scope requirement	B	2
D	Get stakeholder approval	A	1
E	Estimate project budget and end date	B, C	2
F	Design interfaces	C	4
G	Design/build the website/mobile app	F	6
H	Create and combine all databases	F, G	3
I	Integrate all payment methods	F, G, H	3



J	Create schedule for budget and project date		1
K	Create account for human resources	G	2
L	Print tickets	K	1
M	Ensure software works as we want	L	3
N	Close project	M	

## RISK MANAGEMENT PLAN

A risk management plan is a document that a project manager prepares to foresee risks, estimate impacts, and define responses to risks.

Risk analysis: it refers to the technique used to identify and assess factors that may jeopardize the success of a project or achieving a goal.

Before diving deeper, we will start by doing a risk identification that is identify the different risk that may affect our project. Below is a list of different risk we can face;

1. Financial risk: in these project, this risk may occur as a result of shortage in budget either for settlement of team members or for purchase of hardware resources needed for the project.
2. Time risk: it may occur due to limited time. Delay in the project like team members not completing their work in time may lead to the project extending more than its intended dateline.
3. Device risk: this is risk which can be caused as a result of hardware devices getting bad. For example a PC may go bad and this may disrupt the smooth flow of the project.
4. Performance risk: poor planning in terms of design errors and certain constraints may lead to poor performance from both the development point of view and from the product perspective.

5. Project complexity: Lack of identifying modules and functions In detail.
6. Involvement: lack of customer's full involvement in the definition of requirements and their commitment to the project.

Name of Risk	Probability (%)	Impact
Financial risk	10	3
Time risk	10	2
Device risk	5	1
Performance risk	10	3
Project complexity	30	4
Involvement	40	5