

Undergraduate Internship

 Name : Tasnova Tabassum Chowdhury

ID- 1821739

Autumn, 2019

Supervisor:

MD.ABU SYED

Department of Computer Science & Engineering

Independent University, Bangladesh

November 20, 2020

# **Originality Statement**

# This is to certify that the report titled “petty cash management system” is completed by me,Tasnova Tabassum Chowdhury(ID -1821739), submitted in partial fulfillment of the requirement for the Degree of Computer Science and Engineering from Independent University, Bangladesh (IUB). It has been completed under the guidance of Md. Abu syed (Internal Supervisor) and Md..Mushfique Saleheen (External Supervisor). I also to mention the work is original which I have learnt during my internship. All the sources of information used in this project and report has been done according to it.

Name:………………. Signature:……………

**Evaluation Committee**

Signature: .........................

Name: .........................

Supervisor: .........................

Signature: .........................

Name: .........................

Internal Supervisor:.........................

Signature: .........................

Name: .........................

External Supervisor:.........................

Signature: .........................

Name: .........................

Convener:.........................

Letter of Transmittal

January 27th, 2021

Md. Abu Syed

Lecturer

Department of Computer Science and Engineering

Independent University Bangladesh.

Subject: Internship Report submission autumn, 2021.

Dear Sir,

It is a great pleasure and honor to submit my Internship report on petty cash management system. In this report I basically presented my work knowledge, my experience throughout the time period and all the accomplishment in this report.

I have done the internship in NYK Advance Ltd, as a tech marketer and developer.in this time period I have got a lot of understating of the real world work field.

I would like to thank you for your support and guidance throughout the process. And I hope the report do justice to the requirements and expectations

Sincerely

Tasnova tabassum chowdhury

Abstract

In this report, I have explained the knowledge and experiences I have gathered and the work I have done throughout my internship at NYK Advance Ltd as a web developer. I have worked on a web-based solution where mostly my task was to develop new features.

The project was divided into three parts: Users Platform, viewers Platform, and Admin Platform. The project contains all the platforms which is known by Petty Cash Book. All the detailed information are described in this report. It is an online platform that allows users to post instant transactions which is made for company’s business purpose. The requirement is to use this platform for posting instant transactions only to create a user account. User can add a narration with the transactions on this platform.

Viewer can see the reflections of transactions made by users, which will help him to track company’s expenditures. Admin can update or edit user’s role. Admin also has the authority to delete any invalid transactions. This platform will give the opportunity to track all the small expenditures of the company.

Contents

[Chapter 1 8](#_Toc62610791)

[1 Introduction 8](#_Toc62610792)

[1.1 Overview/ Background of the Work 8](#_Toc62610793)

[1.2 Company Profile 9](#_Toc62610794)

[1.2.1 Overview of NYK Advance Ltd 9](#_Toc62610795)

[1.2.2 Projects 11](#_Toc62610796)

[1.2.3 Objectives 14](#_Toc62610797)

[1.2.4 Our Mission 14](#_Toc62610798)

[1.2.5 Our Vision 14](#_Toc62610799)

[1.3 Objectives 15](#_Toc62610800)

[1.4 Scopes of the Project 15](#_Toc62610801)

[1.5 Out of Scope of the Project 15](#_Toc62610802)

[Chapter 2 17](#_Toc62610803)

[Literature Review 17](#_Toc62610804)

[2.1 Introduction 17](#_Toc62610805)

[2.2 How the project is related to the undergraduate courses 17](#_Toc62610806)

[2.3 Petty Cash in Businesses 18](#_Toc62610807)

[Chapter 3 19](#_Toc62610808)

[Project Management & Financing 19](#_Toc62610809)

[3.1 Work breakdown Structure 19](#_Toc62610810)

[3.2 Activity Wise Time Allocation 20](#_Toc62610811)

[3.3 Gantt Chart 21](#_Toc62610812)

[Chapter 4 22](#_Toc62610813)

[Methodology 22](#_Toc62610814)

[4.1 Design 22](#_Toc62610815)

[4.2 Reasons for choosing Agile methodology 23](#_Toc62610816)

[Chapter 5 25](#_Toc62610817)

[Body of the project 25](#_Toc62610818)

[5.1 Description of the project 25](#_Toc62610819)

[5.2 System Analysis of Petty Cash 26](#_Toc62610820)

[5.2.1 Six Elements of System Analysis 26](#_Toc62610821)

[5.2.2 Feasibility Study 27](#_Toc62610822)

[5.3 System Design 28](#_Toc62610823)

[5.3.1 Activity Diagram 28](#_Toc62610824)

[5.3.2 Use Case Diagram 30](#_Toc62610825)

[5.3.3 Functional Requirement 31](#_Toc62610826)

[5.3.4 Non-functional Requirements 32](#_Toc62610827)

[5.4 Product Feature 33](#_Toc62610828)

[5.5 Implementation 35](#_Toc62610829)

[5.5.1 Rest API: 36](#_Toc62610830)

[5.5.2 Single Page Application: 36](#_Toc62610831)

[5.5.3 MEAN Stack: 36](#_Toc62610832)

[Chapter 6 38](#_Toc62610833)

[Results and Analysis 38](#_Toc62610834)

[Chapter 7 41](#_Toc62610835)

[Project as Engineering Problem Analysis 41](#_Toc62610836)

[7.1 Sustainability of the product 41](#_Toc62610837)

[7.2 Business effects and analysis 41](#_Toc62610838)

[7.3 Addressing Ethics and ethical issues 41](#_Toc62610839)

[Chapter 8 42](#_Toc62610840)

[Conclusion 42](#_Toc62610841)

[8.1 Future Work 42](#_Toc62610842)

[8.2 Conclusion 42](#_Toc62610843)

List of Figures & Tables

[Figure 3.1 Work Breakdown Structure for Petty Cash 15](#_Toc62579217)

[Figure 3.2 Activity Wise Time Allocation 16](#_Toc62579218)

[Figure 3.3 Gantt Chart 17](#_Toc62579219)

[Figure 4.1 Agile Software Development Methodology Model 18](file:///C:\Users\Emaduddin%20Rahat\Desktop\New%20folder\New%20folder\New%20folder\New%20folder\Petty%20Cash%20Book.docx#_Toc62579220)

[Table 5.2.1 Six Elements of System Analysis 22](#_Toc62579221)

[Figure 5.3.1 Activity diagram of User 24](#_Toc62579222)

[Figure 5.3.1 Activity Diagram of Admin 25](#_Toc62579223)

[Figure 5.3.2 Use Case diagram of petty cash system 26](#_Toc62579224)

[Figure 5.4 Login Page 29](#_Toc62579225)

[Figure 5.4 Dashboard 29](#_Toc62579226)

[Figure 5.4 Entry Page 30](#_Toc62579227)

[Figure 5.4 Individual Project Update & Delete Transaction 30](#_Toc62579228)

[Figure 5.4 Create User 31](#_Toc62579229)

[Figure 5.4 Manage Users & Change Password 31](#_Toc62579230)

[Table 6 Result & Analysis 36](#_Toc62579231)

# Chapter 1

# Introduction

## Overview/ Background of the Work

Petty Cash is otherwise called cash in hand which is characterized as a limited amount of money put aside to cover for minor expenditures in the company without composing a cheque. A petty cash fund will go through periodic reconciliations with transactions. It is recorded as a financial statement. Generally, in big corporate companies, there might be a petty cash fund which can be a cabinet or a drawer. It can be used for various purposes such as office supplies, cards, and gifts for customers, paying for a catered lunch for employees and reimbursing employees for work related expenses.

The use of a fund circumvents certain internal controls. The availability of funds doesn't mean that it is often accessed for any purpose and by anyone. Many companies employ strict internal controls to manage the fund. Often, limited individuals are authorized to approve disbursements and may only do so for expenses associated with legitimate company activities or operations.

A petty cashier could be assigned to issue the check to fund the petty cash drawer and make the acceptable accounting entries. The fund user is charged with distributing the cash and collecting receipts for all purchases or any uses of the funds. As the fund total declines, the receipts should increase, and tie bent the entire amount withdrawn.

By having a petty cash viewer and a petty cash user, the dual process helps to keep the funds secure and ensure that only those authorized have access to it. In the petty cash book, all the project details are seen in the dashboard along with Tally Balance, IOU (I owe you), and total balance. More details of a project is seen after clicking on a particular project.

## Company Profile

### Overview of NYK Advance Ltd

NYK Advance Ltd. is one of the emerging IT Firm in Bangladesh. It has started its journey since 2014 with lots of potentiality. It has a dream to make the workplace easier and very flexible along with the businesses for every person who wants to think ambitiously with enthusiasm and crave for the social welfare. As a software development company NYK Advance Ltd. is looking to strategically transform, develop and lead in the present challenging business environment throughout the globe. It is upheld by an extremely innovative management group, very experienced software engineer group, and a very dynamic and effective marketing Team.

NYK Advance Ltd. has comprehensive range of software services and Web solutions enable it to serve more then 50+ customers. It has a service portfolio that ranges from traditional to transformational line of web Development & web application, which includes (Business Application Services, Enterprise Solutions, Enterprise Collaboration Services & Infrastructure Management Service.

At NYK Advance Ltd., we see Innovation as a clear differentiator. Innovation, along with focus on deep, long-lasting client relationships and strong dedication, drives every facet of our day-to-day operation. For our clients, it translates into strategic business value and for us it is consistent financial performance. Brand name of NYK Advance Ltd. is known as Destreza Solution.

**Fact Sheet**

|  |  |
| --- | --- |
| Particular | Facts |
| Official Name | NYK Advance Ltd |
| Brand Name | Destreza |
| Year of Establishment | 2014 |
| Business Type | Software Development  IT Enabled Services |
| Trade License Number | 0201024 |
| VAT Registration Number | 19011097509 |
| TIN Number | 376869621831 |
| Affiliation | BASIS Member (A373) |
| Address | House- B/186 (4th Floor), Road- 20  New DOHS, Mohakhali, Dhaka |
| Contact Information | +880 9835437  ahsan.mahmud@destreza.io |
| Website | [www.destreza.io](http://www.destreza.io) |
| Number of Employees | 50+ |
| Key Solutions | Enterprise Resource Planning  PoS Solution |
| Number of Clients | 61 |
| Number of Registered Users | 10000+ |
| Number of Active Users | 5000+ |

### Projects

Here is a list of the corporate projects the company has done

1. **Procurement Management System**
   1. Perform all types of purchases material Issue
   2. Track purchase orders & requisitions.
   3. Vendor management
   4. Material master
   5. Vendor advance payment management
   6. Rich set of reports
   7. Cost budget- cost center or cost ledger levels
2. **Inventory Management System**
   1. MRR (Material Receive Report)
   2. Material Issue
   3. Material Transfer
   4. Stock Adjustment
   5. Material Issue Return
   6. Material Return
   7. Location Management
3. **Accounting Management System**
   1. Dynamic chart of accounts
   2. Unified group of ledgers
   3. Suitable for group of companies
   4. Role & position-based access control
   5. Unlimited profit & cost center management
   6. Account receivable & payables
   7. Bank reconciliation
   8. Ledger statement
   9. Cash/ fund flow statement
   10. Ratio analysis
4. **Bank Management System**
   1. Bank account management
   2. Cheque beneficiaries ‘management
   3. Cheque book management
   4. Cheque print management
   5. Cheque review
   6. Support all type of banking transaction
   7. Bank transfer letter print (BEFTN)
5. **Sub-Contract Management System**
   1. All types of subcontract jobs of a project
   2. Control tool to manager subcontract work
   3. Track advance, TDS, Retention & Payment
   4. Work progress monitoring and reporting
   5. Work order approval process
   6. Manage complex subcontracting jobs
   7. Strong reporting tool
   8. Automated billing process
6. **Budget Management System**
   1. Cost budget
   2. Revenue target
7. **Sales Management System**
   1. Quote generation
   2. Sales order generation
   3. Delivery order
   4. Factory delivery
   5. Sales return
8. **Petty Cash Management System**
   1. Track cash/ Money Flow
   2. Track IO
   3. Count day & monthly balance
   4. Track cash box status
9. **HRIS & Payroll Management System**
   1. Dynamic setup creation management
   2. Employee management
   3. Attendance management
   4. OT, POD, NS Management
   5. Leave Policy Management
   6. Manpower planning
   7. Departmental Budget
   8. Performance Appraisal
10. **LC Management System**
    1. Efficient & convenient L/C management
    2. Track financial document
    3. Easy bank payment
    4. Maintain commercial invoice
    5. Track landed cost
11. **Fixed Asset Management System**
    1. Asset capitalization
    2. Asset revaluation
    3. Asset depreciation
    4. Asset transfer
    5. Asset disposal/ retirement
    6. Asset tracking
    7. Asset location
    8. Lifecycle monitoring
12. **Vehicle Management System**
    1. Vehicle run management
    2. Vehicle cost management
    3. Fuel cost management
    4. Incidental cost management
    5. Statutory cost management
    6. Driver management
13. **File Archiving Management System**
    1. High speed scanning
    2. Data store
    3. Data tracker

### Objectives

1. Getting and staying profitable
2. To establish as the best ICT solution provider
3. Comprehensive individual services
4. Increasing sales in the upcoming years
5. To increase company profile even on International ground.

### Our Mission

Our mission is to become and integrated software solution provider and enhance technology for business growth and success.

### Our Vision

Providing innovative and standard solution to meet our client needs with quality

## Objectives

The objective of this project is to develop a solution for users so that this gives them a platform to track and post every small transaction directly made from the desk. My objective is to develop the Platform and Admin panel activities related to the Platform.

Objectives of petty cash are as follow:

1. To control the disbursement and proper analysis of expenses.
2. To prevent company’s accounts against waste, fraud, and inefficiency.
3. To secure a high level of accuracy and authenticity in the accounting records.
4. To reduce the scope for errors or losses without experiencing unreasonable subsidiary costs for the increments of intramural controls established.

## Scopes of the Project

A department's petty cash fund is used to reimburse suitable low-value expenses for things which generally can’t be purchased through established procurement procedures. That’s why, petty cash fund users are delegated to administer the fund.

Here are the features available in this web application for user, viewers, and Admin in Platform:

1. User, Viewer, and Admin Loggin Page
2. Dashboard with Project list, Tally Balance, IOU Balance, and Total Balance
3. Payment Option
4. Receive Option
5. Make Normal Transaction
6. Make IOU (I owe you) Transaction
7. Notes and coin denomination
8. Add Narration
9. Export Option
10. View all Transaction (Viewer)
11. Update Transaction (Admin)
12. Update/Delete Transaction (Admin)
13. Manage User (Admin)
14. Manage profile (Admin)

## Out of Scope of the Project\

* + - 1. User Password Change (User)
      2. File Attachment

# Chapter 2

# Literature Review

## Introduction

The petty cash fund is accommodated intermittently to verify that the balance of the fund is correct. Typically, as the petty cash balance tumbles to a preset level, the user applies for extra amount from the cashier. now, the absolute of the entirety of the receipts is determined to ensure that it coordinates the disbursed funds from the petty cash cabinet. If new funds are required, the cashier composes a new check to fund the petty cash cabinet/drawer and takes, in return, the receipts from the purchases that lessen the cash.

The reconciliation process ensures that the fund's remaining balance equals the difference between the first balance minus charges detailed on receipts and invoices. If the remaining balance is a smaller amount than what it should be, there's a shortage. If the remaining balance is quite what it should be, there's an average. Although there are often minor variances, when unbalanced, the source of the discrepancy should be identified and corrected.

## How the project is related to the undergraduate courses

1. **CSE-307 System Analysis and Design:** This course helped to establish the project's overall structure before even beginning to interact with codes. This course taught all the required observations that needed to be considered, such as requirement analysis, flow diagrams, UMLs, rich image, functional and non-functional requirements, methodology, WBS, Gantt Map, etc. This course was of great support, not only for the project but also for the report. From the beginning to the end, this course helped to prepare the entire project.
2. C**SE-213 Object-Oriented Programming:** This course helped to deal with the objects, classes, inheritance, chaining of functions, etc. that were required when working with the system controller and model. I was introduced to Java through this course, which supported me during the project.
3. **CSE-303 Database Management:** A large portion of the project is about the backend. The overall understanding of SQL was provided by this course. This course has provided me the knowledge of the database that has supported me throughout the project, starting from communicating with phpMyAdmin to writing queries. This course made it easier for me to develop the project with all the essential questions I was aware of, such as joining two tables using the international key, providing a table with the primary key, etc.
4. **Web Application and Internet**: This course was beneficial when dealing with the frontend design. I have learned basic HTML, CSS, PHP, JavaScript, etc from this course. Even though we were taught raw PHP, because of the experience I learned from this course, it was still not difficult to use a system. This course has been a tremendous help in the project, as I have also learned to use colors and functionality to create an appealing user interface.

## Petty Cash in Businesses

Employees take money from the petty cash account, record the date, amount, vendor (who was paid), and business intent of each transaction.

All transactions using petty cash must be reported in the same manner as other company earnings and expenditures. Using a small cash log or petty cash slips can help capture these costs so that they can be used for business tax purposes to cover profits.

Someone needs to regularly monitor the account, so the balance doesn't get too low. To add to the drawer by writing a check to "Petty Cash" and cashing the check, when the petty cash drawer gets below a pre-set sum you decide.

# Chapter 3

# Project Management & Financing

## Work breakdown Structure

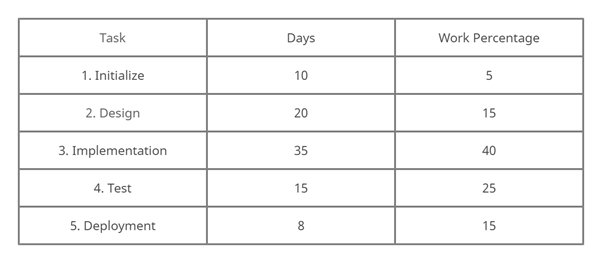
Breaking work into smaller tasks is a common productivity technique used to make the work more manageable and approachable. For projects, the WBS is the tool to utilizes this technique and is one of the most important project management documents. It is the perfect tool for the team for brainstorming and cooperation. We have used the top-down approach in WBS.



Work Breakdown Structure for Petty Cash

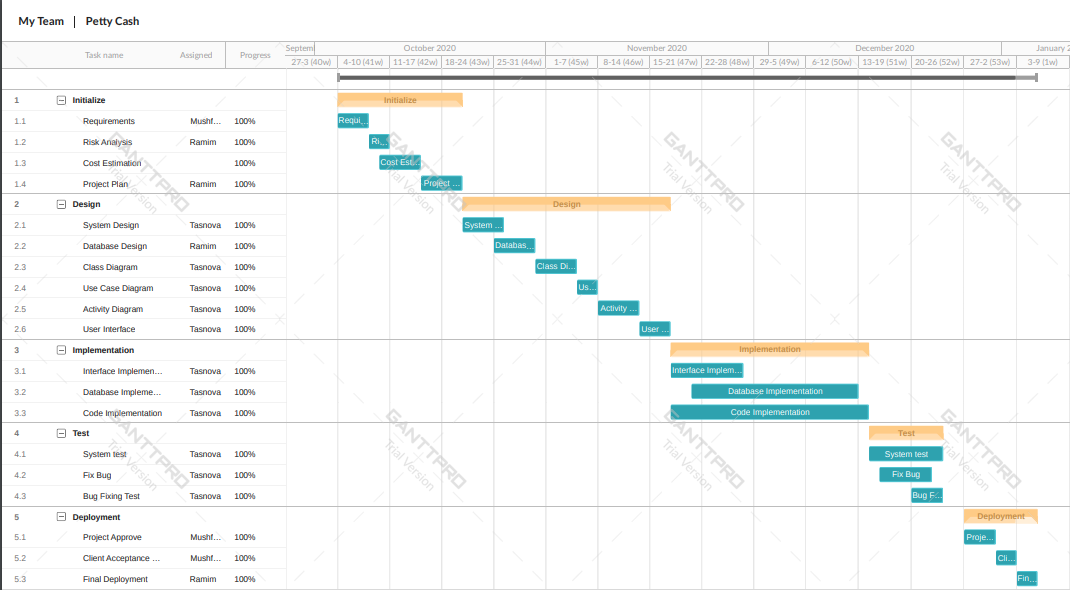
## Activity Wise Time Allocation

For each section that I have mentioned in my WBS, time was allocated because of completing it on time. Time allocation of WBS is shown below



## Gantt Chart

We have used Gantt Chart to plan and schedule all our activities that need to be done to complete the project successfully. This gantt Chart shows the 3- month time frame of my internship period. It also shows WBS where I have mentioned which part was assigned to each member.



Gantt Chart

* 1. Estimated Costing

Cost was calculated based on the Requirements, Functionalities, Design. This includes pre-designed themes, logo design cost, search engine optimization, training option and many other tools that were used to build the website. The cost of resources used on the website were also taken in note. The approximate cost estimation is 3,00,000 with one-year free service support and training up to Twenty members. If service support is needed after one year, then additional charge will be applied. If new requirements come after deployment, additional charge will also be added.

# Chapter 4

# Methodology

## Design

Basically, a software development methodology indicates a process which includes developing software. This process can be divided into four stages to plan, structure, improve and control.

There are several software development methodologies which are:

1. Spiral development
2. Rapid application development
3. Agile
4. Iterative and incremental development
5. Waterfall
6. Extreme programming
7. Prototyping

We would like to go with agile software development methodology for our project. This methodology is based on iterative development. Agile software development methodology includes adaptive planning, evolutionary growth, early delivery and quality improvement along with promoting agile response to changes. It is more people-centric compared to other traditional approaches. The project is simultaneously being observed by us & the client so if any changes are required then it can be made during the development phase.



Figure 4.1: Agile Software Development Methodology Model

At first, we noted down the overall project requirements from the client. We analyzed all these requirements like the target audience of the website, the purpose and the goals of the website etc. We evaluated the complexities, size, risks, feasibility study and cost. A plan was made which includes the structure, team members, technical requirements, etc. We created a system design demo of the overall process of how the software is going to function by using Rich Picture, Use Case & Activity Diagrams. Once the overall process is generated, we distribute works among the two of us.

The first work was gradually developing and testing simultaneously. Before compiling our parts, we tasted each part individually so that bugs are identified and can be fixed easily. After that our task was the frontend design of the business platform was completed. Our client reviewed it. We noted down their feedback. Based on that we made changes if it was needed. We implemented these changes before we started to work on the next sprint of the project. Overall project will be tested and reviewed for any bugs once it is completed. After the client’s approval, it will be finally deployed.

## Reasons for choosing Agile methodology

1. **Faster Development:** The software is developed in different phases and each phase is known as an individual project under agile methodology. We divided our project in two parts and we both had a deadline. Everyone tried their best for each part. This methodology speeds up the overall development process and also provides faster delivery.
2. **Higher Customer Satisfaction:** It allowed us to work closely with the clients as a team and understand their requirements in the project. It helped us to keep both the parties at the same page and also allowed us to make any changes which were required before we went forward to the next stage. Agile also brought transparency to the project as we didn’t deliver the software to the clients at the end and were delivered during the development process.
3. **Better quality:** Agile methodology doesn’t test the software at the end of the development period like other methodologies. The software is tested at every stage which reduces the chances of getting a bug at the end of the project development. This also allows us to inspect the software at every level and make changes as per requirements. Thus, ensuring a higher quality of software at the end.
4. **Reduced risks:** Agile allows us to take calculated risks by not using outdated data while developing the software. Agile methodology allows us to make required changes to any particular phase without disturbing the previous phase. Thus, it accepts changes even when the project development is finished.
5. **Feedbacks are incorporated quickly:** Delivering the product quickly will be of no use if the product can’t fulfill the requirements that the client has demanded. Agile methodology always gives importance to client’s feedback which helps to develop the software efficiently and incorporate the changes before the next meeting with the client.
6. **Productive level is high:** Agile methodology sets a clear vision and goals for programmers. The timeline of the work is divided and each team member knows their task. So ultimately it increases the productivity level of the project.

# Chapter 5

# Body of the project

## Description of the project

This project was divided into two parts: Cash Register and the Admin Panel. My work was to develop the cash platform and the Admin panel. This report contains mostly the cash related which is known as the petty cash on the website and the Admin functionalities. All the detailed information is mentioned in this report. It is an online platform that gives the company and users the opportunity to post the transections related to cash about their prospective projects and uses. The requirement to use this website for entry data related to cash transactions which is operated both by user and the admin panel.

As mentioned before, this software has two separate panels where both the user and admin can log in at the same time to do the data entry and other modifications. Admin can erase data and also modify data at any time by accessed with their prospective user id and password system. This software is also linked with the selective bank with which the organization do their transactions and other financial works.

Petty cash is a cash related software which created in the year ……… For the advance works of cash related transactions, it helps to collect all data in a very appropriate way. Data’s are easily collected and store in one place for future requisitions. There are some determined id and passwords for the users and admin panel. Customers can upload their data in the software using the id and password. Admin can log in to the customer’s account also to gather and isolate the data for future use. Petty cash software has some features for the admin users only.

Here is the list of features which can be accessed by the user. they are

1. Log in page
2. Dashboard
3. Entry Transaction
4. Transaction type
5. Multiple Currency
6. Individual Project transaction

For admin panel there are some common features with the user. But some of the features are different and unique to separate the administrative and user accounts. The features are,

1. Create user
2. Manage user
3. Change password
4. Update transaction
5. Delete transaction

## System Analysis of Petty Cash

### Six Elements of System Analysis

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | System Roles | | | | | |
|  | Human | Non-Computing Hardware | Computing Hardware | Software | Database | Communication |
| Login Page | User | N/A | Computer | Petty Cash | MONGODB | WAN |
| Dashboard | User | N/A | Computer | Petty Cash | MONGODB | WAN |
| Entry Transaction | User | Paper, Pen | Computer | Petty Cash | MONGODB | WAN |
| Transaction Type | User | Paper, Pen | Computer | Petty Cash | MONGODB | WAN |
| Multiple Currency | User | N/A | Computer | Petty Cash | MONGODB | WAN |
| Individual Project Transaction | User | Paper, Pen | Computer | Petty Cash | MONGODB | WAN |
| Create User | Admin | Paper, Pen | Computer | Petty Cash | MONGODB | WAN |
| Manage User | Admin | Record Book | Computer | Petty Cash | MONGODB | WAN |
| Change Password | Admin | N/A | Computer | Petty Cash | MONGODB | WAN |
| Update Transaction | Admin | Record Book | Computer | Petty Cash | MONGODB | WAN |
| Delete Transaction | Admin | Record Book | Computer | Petty Cash | MONGODB | WAN |

### Feasibility Study

Feasibility Study is used to determine the reasonability of a project. It determines whether the project is legal, technically, economically feasible and importantly whether the project is worth investing in. It evaluates the potential of the success of the process. Five feasibility studies are considered for this project.

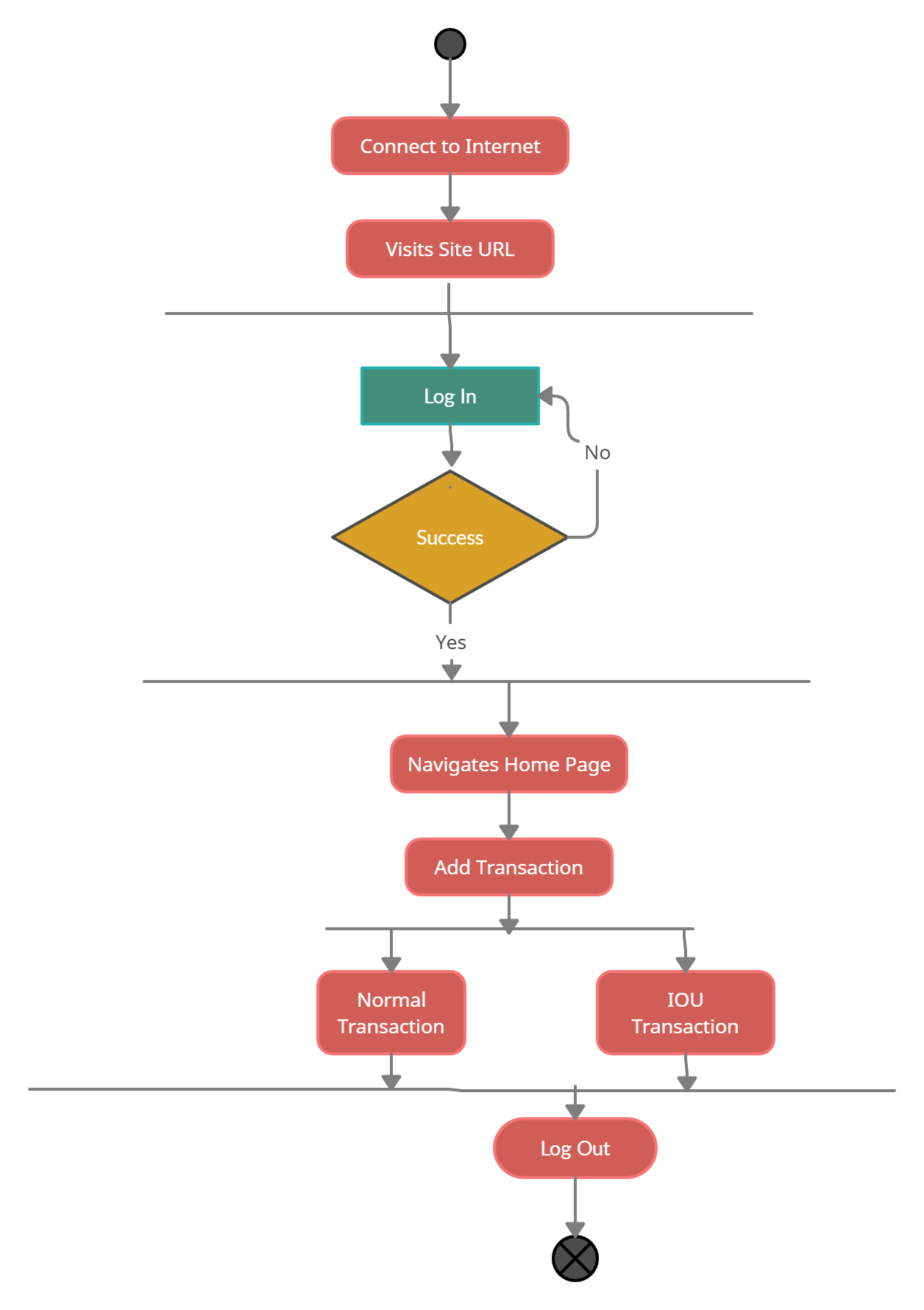
1. **Technical Feasibility:** NYK Advance Ltd is an established company. It is capable of meeting all the technical requirements and the team is capable of converting the idea of the project into reality. All the hardware, software and other technical requirements needed to complete this project are available making this technically feasible.
2. **Economic Feasibility:** The cost and beneﬁts of this website have been considered in this section. This website has been kept open for all users for now without the need to pay for blogs. After gaining popularity, the company might decide to make it paid if needed. Till then the company hopes to gain credibility and positive economic beneﬁt for the users.
3. **Legal Feasibility:** All the legal constraints have been considered before proceeding with this project which includes data protection acts, social media laws, or zoning laws so that it does not face any legal constraints in the future.
4. **Operational Feasibility:** This project solves the problem of unrecognized transactions and people that are not maintain transaction properly this project is more valuable for this business. Moreover, the project plan also satisﬁes all the requirements that have been identiﬁed to complete the project.
5. **Scheduling Feasibility:** To make a project successful, it is very important to complete a project at the given time. The project to date has been completed on time that has been designated for each task.

## System Design

### Activity Diagram

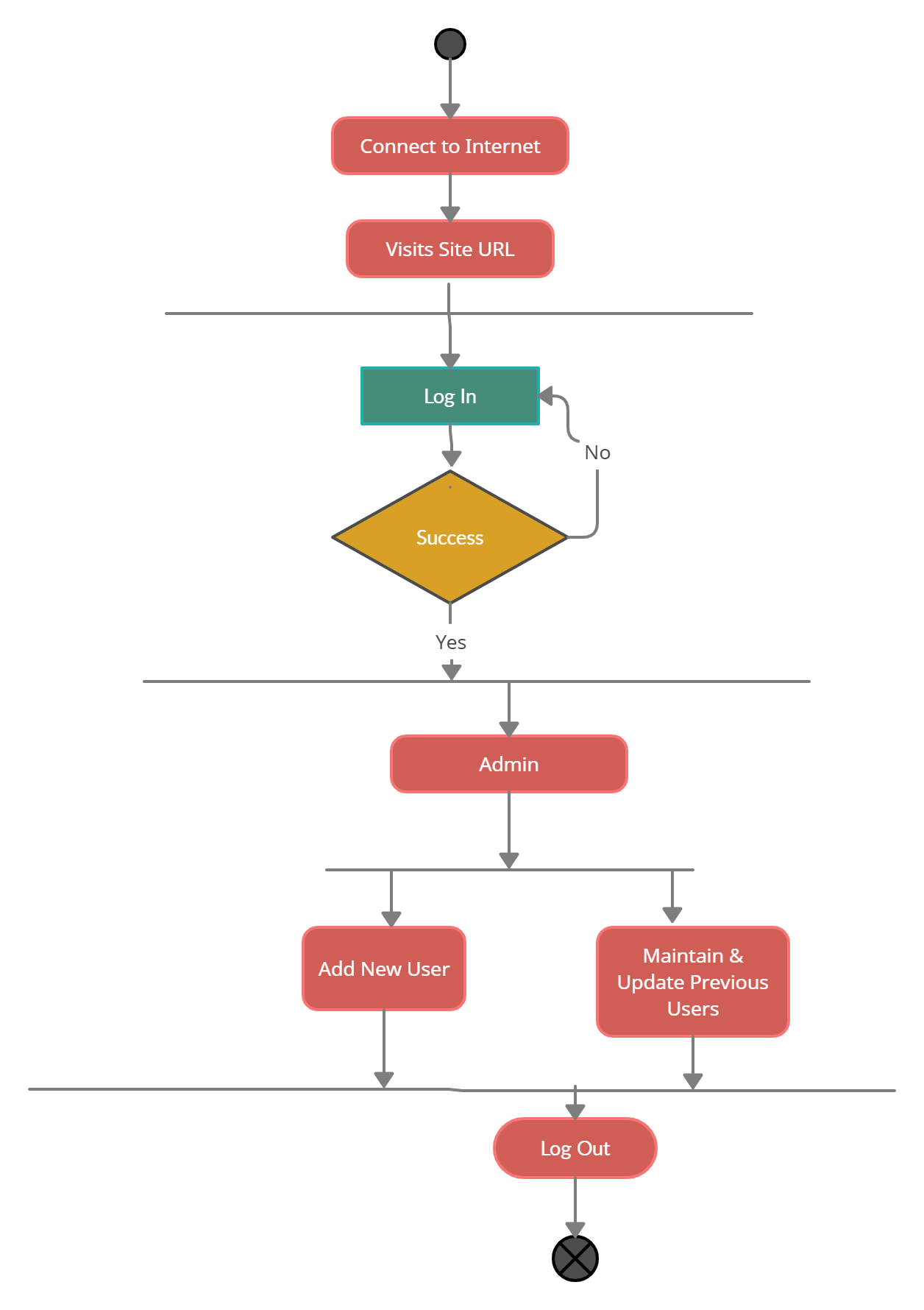
Activity diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that models the flow from one activity to another activity.

**Activity Diagram of User**

****

Activity diagram of User

**Activity Diagram of Admin**

****

Activity Diagram of Admin

### Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.



Use Case diagram of petty cash system

### Functional Requirement

Functional requirements define” what the system should do”. It defines what the next step should be when a certain condition is fulfilled. It is directly related to what we expect the web application should do. The functional requirements of this website are given below.

**Login for User**

**Input**: Provide user information

**Process**: Visits Website and login to User Interface

**Output**: User successfully logs in

**Post**-**condition**: User enters the Homepage and can perform the activities

**Alternative** **Options**: New password can be requested using forgot password option (implementation is going on)

**Operations by User**

**Input:** Userprovides the desired amount

**Process:** User Makes two types of payment. 1.Normal payment, 2. IOU payment.

**Output:** User successfully makes the payment

**Post-condition:** User made the payment.

**Alternative Options: N/A**

**Login for Admin**

**Input:** Provide Admin information

**Process:** Visits Website and login to Admin panel

**Output:** Admin successfully logs in

**Post-condition:** Admin enters the Homepage and can perform the activities

**Alternative Options:** N/A

**Operations by User**

**Input: N/A**

**Process:** Manage users & their Roles

**Output:** User successfully manages the User

**Post-condition:** N/A

**Alternative Options: N/A**

### Non-functional Requirements

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

**Performance Requirements**

Performance defines how fast a software system, or its particular piece responds to certain users’ actions under certain workload. In most cases, this metric explains how much a user must wait before the target operation happens (the page renders, a transaction is processed, etc.) given the overall number of users at the moment. But it’s not always like that. Performance requirements may describe background processes invisible to users.

**Scalability Requirements**

assesses the highest workloads under which the system will still meet the performance requirements.

**Reliability Requirements**

Reliability. This quality attribute specifies how likely the system or its element would run without a failure for a given period of time under predefined conditions. Traditionally, it’s expressed as a probability percentage. For instance, if the system has 85 percent reliability for a month, this means that during this month, under normal usage conditions, there’s an 85 percent chance that the system won’t experience critical failure.

**Maintainability Requirements**

Maintainability defines the time required for a solution or its component to be fixed, changed to increase performance or other qualities, or adapted to a changing environment. Like reliability, it can be expressed as a probability of repair during some time. For example, if you have 75 percent maintainability for 24 hours, this means that there’s a 75 percent chance the component can be fixed in 24 hours.

**Security Requirements**

This non-functional requirement assures that all data inside the system or its part will be protected against malware attacks or unauthorized access. But there’s a catch. The lion’s share of security non-functional requirements can be translated into concrete functional counterparts. If you want to protect the admin panel from unauthorized access, you would define the login flow and different user roles as system behavior or user actions.

**Usability Requirements**

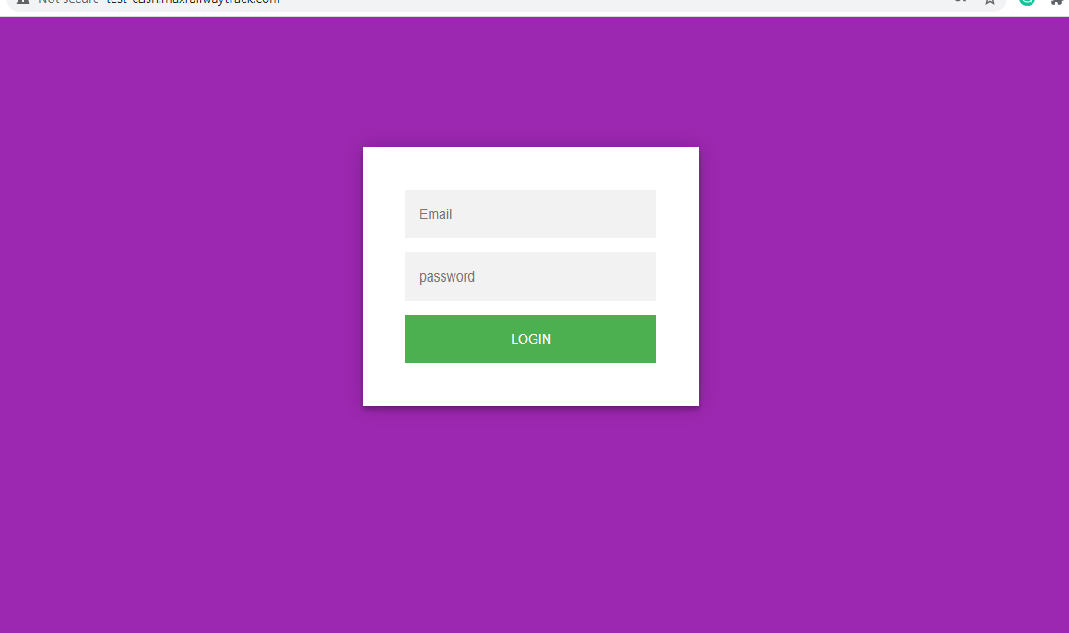
Usability is yet another classical nonfunctional requirement that addresses a simple question: How hard is it to use the product? Defining these requirements isn’t as easy as it seems.

## Product Feature

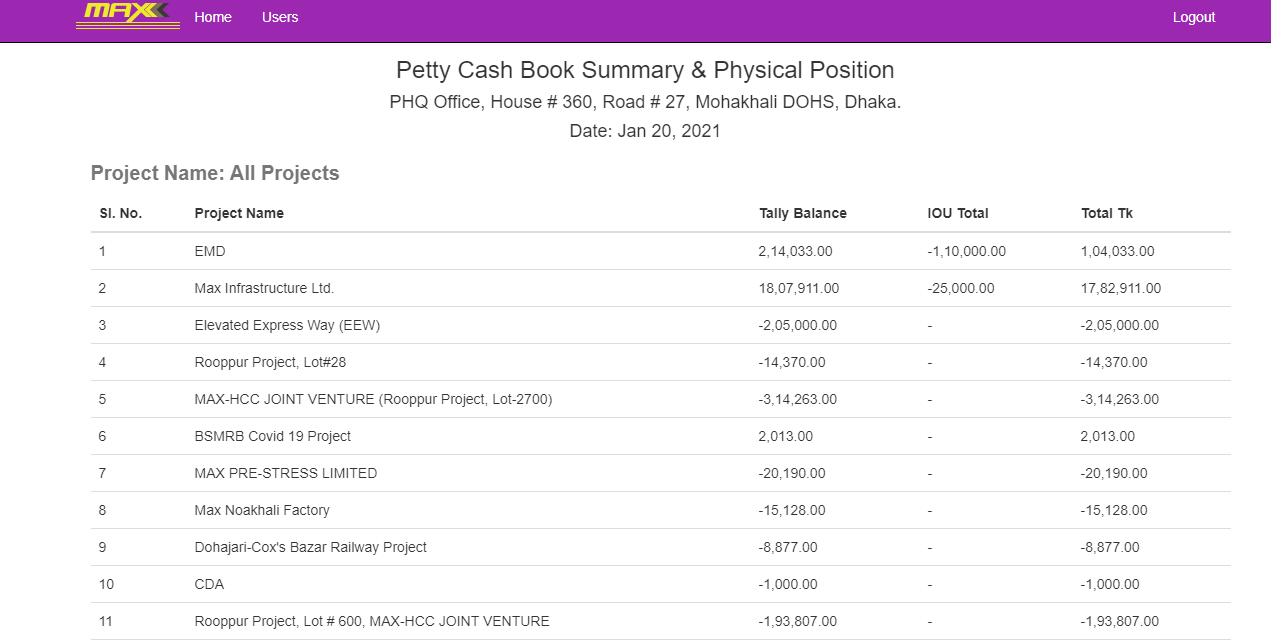
I have included all the feature which is included in the project:

**Login Page**

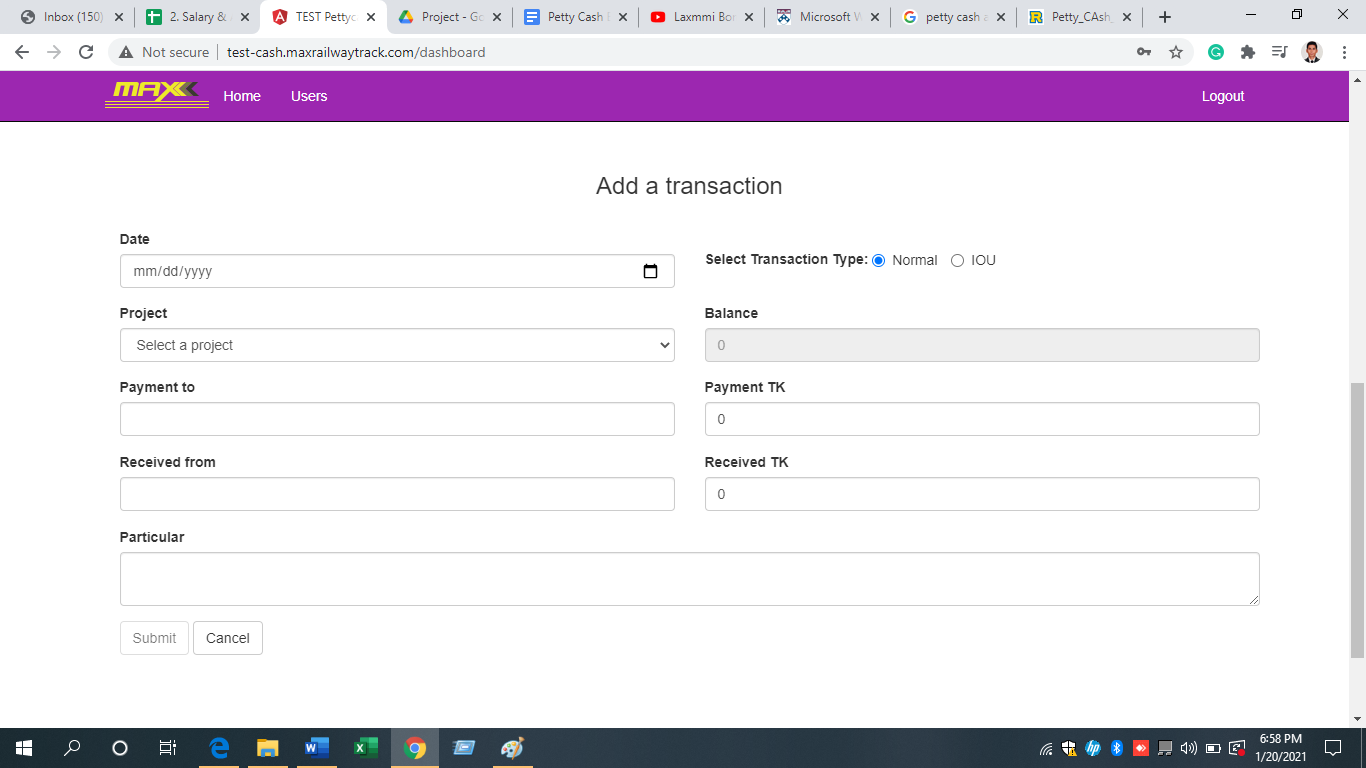
The user logs in to their account.



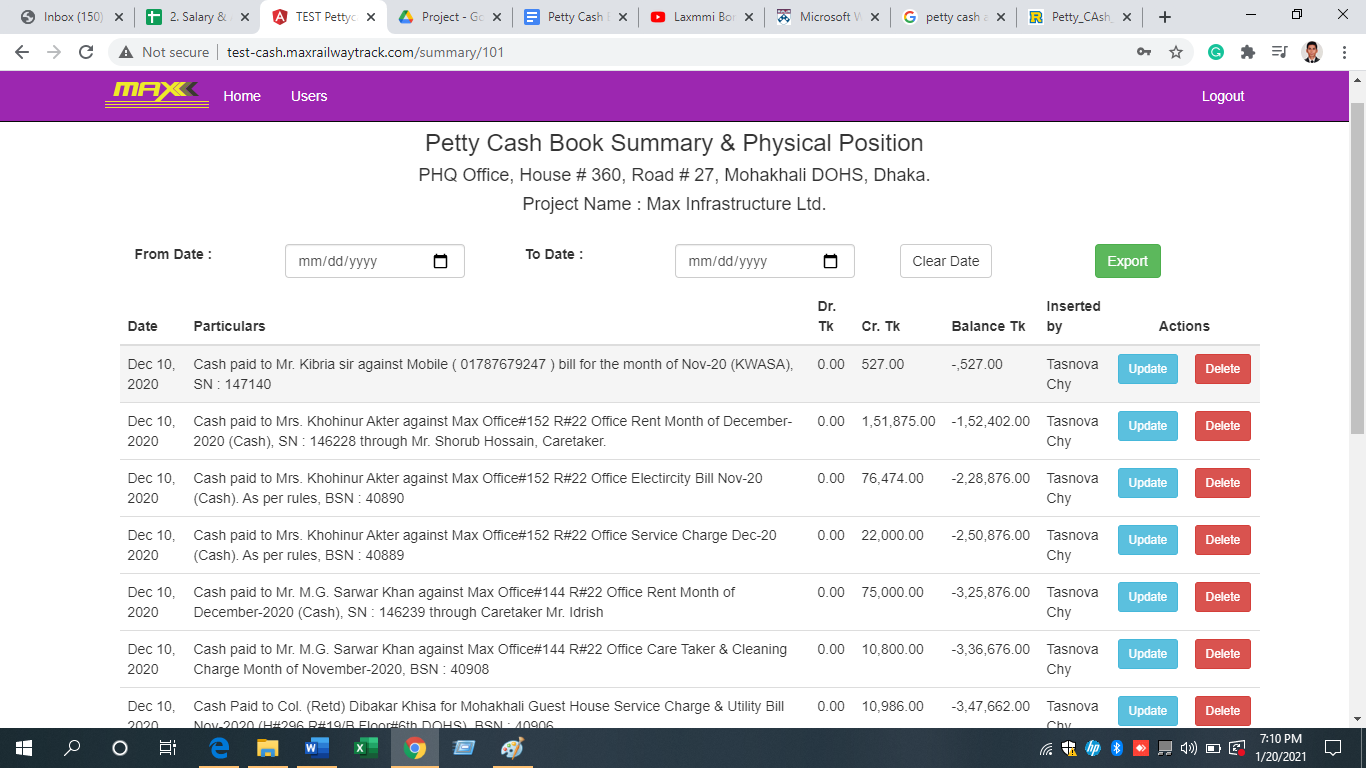
**Dashboard**



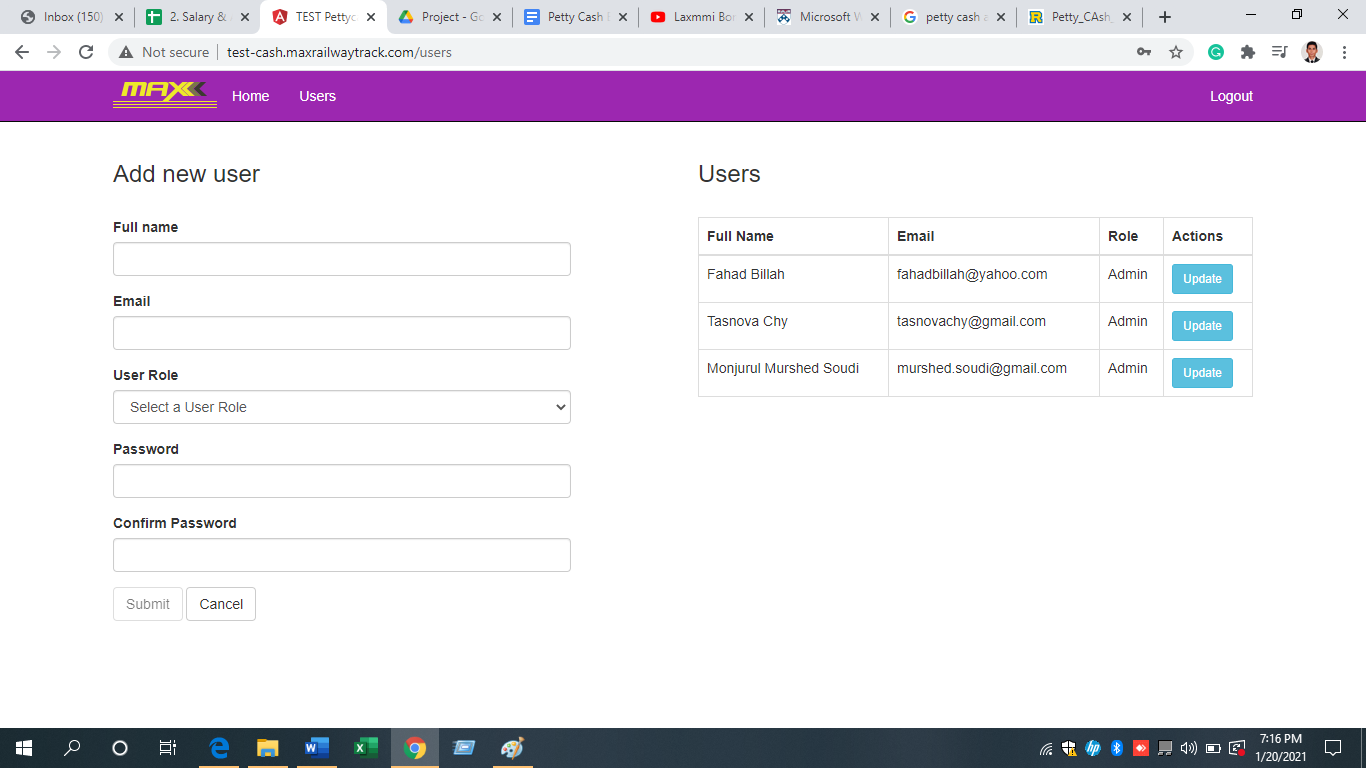
**Entry Transaction**



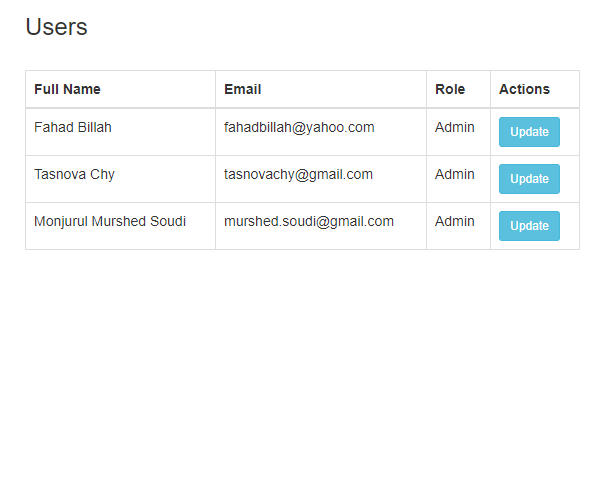
**Individual Project Update & Delete Transaction**



**Create User**



**Manage Users & Change Password**

****

## 5.5 Implementation

As the whole module application is lightweight feature wise yet has heavy usage, we needed a technological stack that is super-fast, easy to maintain and has a slim hardware footprint. For server side of the application, we picked ExpressJS as our backend framework which is written in NodeJS language. For database, we picked MongoDB which is highly scalable and feature rich lightweight DBMS. For frontend we selected AngularJS framework and Bootstrap 3.1. The whole client-side application in written in HTML5, CSS3 and JavaScript ES5 language. To communicate between server and client side we used a Rest API. The client-side application is a single page application that does not require page reloading to browse from page to page.

5.5.1 Rest API: Full form of REST API is Representational State Transfer Application Programming Interface or REST API. It means when a REST API is called, the server will transfer a representation of the requested resource’s state to the client system.

5.5.2 Single Page Application: A single-page application is an app that works inside a browser and does not require page reloading during use. You are using this type of applications every day. These are, for instance: Gmail, Google Maps, Facebook or GitHub.

SPAs are all about serving an outstanding UX by trying to imitate a “natural” environment in the browser — no page reloads, no extra wait time. It is just one web page that you visit which then loads all other content using JavaScript — which they heavily depend on.

SPA requests the markup and data independently and renders pages straight in the browser. We can do this thanks to the advanced JavaScript frameworks like AngularJS. By implementing SPA we got the following benefits:

• SPA is fast, as most resources (HTML, CSS, Scripts) are only loaded once throughout the lifespan of application. Only data is transmitted back and forth.

• The development is simplified and streamlined. There is no need to write code to render pages on the server. It is much easier to get started because you can usually kick off development from a file file://URI, without using any server at all.

• SPAs are easy to debug with Chrome, as you can monitor network operations, investigate page elements and data associated with it.

• SPA can cache any local storage effectively. An application sends only one request, store all data, then it can use this data and works even offline.

• Single-page sites help keep the user in one, comfortable web space where content is presented to the user in a simple, easy and workable fashion.

5.5.3 MEAN Stack: The letters in MEAN stack stands for: (M)ongoDB, (E)xpress, (An)gularJS. MongoDB stores data in a JSON-like format (BSON, a binary JSON extension), the MongoDB Query Language (MQL) is defined in JSON, and its command line interface (CLI) is a JavaScript interpreter. Not only is MongoDB essentially a JavaScript/JSON data store, it’s full of advanced features like indexing and querying deep into JSON documents, has powerful native Node.js drivers, and is designed for horizontal scale-out. It’s even easier to develop apps in the cloud using MongoDB Atlas, the cloud-native database-as-a-service from the creators of MongoDB.

The E and A of MEAN (Express and Angular) are two of the most popular and well-supported JavaScript frameworks for back-end and front-end development, respectively. Express makes routing and managing HTTP requests and responses super easy, and includes great support for middleware to handle JSON endpoints and form posts. Angular is a powerful tool for building dynamic HTML pages that communicate with a back-end server.

# Chapter 6

## Results and Analysis

A Table showing below, have all the results of my tasks that I was performing. Every task tests properly and it gains all the terms and conditions successfully that are given. This web application is tested several times by professional developers. At the beginning there are lots of bug in this application, several SQA professional testers test this app again and again, and then I fix the bugs. All testing and development procedures are completed by going through Agile Method. Backend and frontend codes are developed by using Git procedure. The percentage of error is below 1% only. This project is live now and host in amazon server. All features are performing well, though some features will be added in future. Then the features will be tested and developed as per requirement.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Feature | Description | Conditions | Success Rate | Error Rate | Upcoming | Working/Not Working |
| Log In | User needs to log in successfully to performing action. | User needs to  have a smart device, connected to an internet.  User needs to  enter an unique  email address  To login. | 100 | 0 |  | Working. |
| Password | User needs to log in successfully to perform action. | User needs to  have a smart device, connected to an internet. | 100 | 0 | Recovering password through email if he/she forgets password. | Working. |
| Menu bar | To see features and performing action. | User needs to  have a smart device, connected to an internet. | 100 | 0 | Will add some more features in future. | Working. |
| Users | To see admin list and updating users Action. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Add New User | To register new Admin, User or viewer. And also following registration policy. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Log Out | To logout from application | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Home Menu | To see All projects, Add a transaction and denomination. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Add a Transaction (Normal) | Add transaction by filling up Date, Project, Payment to, Particular, Balance, Payment TK etc. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Add a Transaction (IOU) | Add transaction by filling up Date, Project, Payment to, Particular, Balance, Payment TK etc. same to normal transaction. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Notes and Coin Denomination | Add note amount with quantity, this is editable. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| All projects | Several project and their tally balance and total balance are showing with proper calculation. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| After click on any Project | Showing particulars, Dr. TK, Balance TK, Inserted by and action. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Total Amount | Showing Dr. Amount, Cr. Amount, and Balance TK properly. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Filter by Date | Summary data view and take action by filter from date to date. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Update | Admin and User can Update any information | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Delete | Admin, User can delete and perform action. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Clear Date | Clear Date from list. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Export Data | Export all data in excel sheet or csv format. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| Show data | Shows data if there are lots of data. Then data can be shown by using preview or next button. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |
| User Role | There are three types of rules. Admin can perform any kind of action. User is in 2nd position. And viewer only can see data. | User needs to  have a smart device, connected to an internet. | 100 | 0 |  | Working. |

# Chapter 7

# Project as Engineering Problem Analysis

## Sustainability of the product

By making these minor but important improvements, the module has been made sustainable.

1. Delete unused data: We intend to regularly delete unnecessary data once a year so that we can free up our database space. This may include users who have been inactive for more than a year, groups, themes that have not been used, etc.
2. Site Navigation: We have made the website as basic as possible so that the website is not hard for users to use. So, they don't have to navigate around the website and find the correct details quickly.

## Business effects and analysis

Petty cash operates on the impress scheme, in which an initial sum of money is placed into an account that is drawn for a particular purpose (in this case, petty cash). The device is replenished when the account goes below a certain specified number.

It is mostly set up in the front office with a petty cash drawer or box or in a retail location using a cash register for organizations that have a regular need for petty cash. (Yes, a cash register is still used at some locations.)

The use of a Petty cash system (with a "cash drawer") has declined because the cash economy is getting smaller, and people use debit and credit cards more for small transactions. But it is also necessary for every organization to make sure that all small transactions are accounted for so that you can deduct them as business expenses.

## Addressing Ethics and ethical issues

People have a different code of ethics around the world. Problems can arise when those who can trigger a fight point out one's ethics wrongly. Other problems can arise if an individual without the permission of the writer copies other contents and does not give the creator credit. That can cause problems with copyright. Third, the problem that could arise is whether the module created is not legitimate and deceptive and given with the purpose of harming individuals. All these problems will be documented by approved administrators so that ethical problems can be avoided. The company also plans to provide users with ethical guidelines that include copyright and credit, post validation, spam content, user deactivation, if applicable, so that all of the above issues can be resolved.

# Chapter 8

# Conclusion

## Future Work

NYK Advance Ltd needs more professional developers to improve this project further, who can shape this project in a more appealing way. In front-end design, developers that have a better concept could be used to make the website more attractive. In my opinion, users should have been provided with functionality such as design options or different layout options to make the site more useful. Such features work a lot in attracting businessman to the website, with various font sizes, types, colors, and languages.

## Conclusion

Nowadays, Instant transactions is a very essential part of businesses. The idea of Petty Cash emerged from the needs of tracking instant transactions in businesses. Today people are faster and more transparent of doing transactions for the businesses through using papers, excel and software. This platform offers a forum for sharing one's knowledge of their business. I have provided detailed information about this platform in this project. The context, features, how it works, the methodology used, outcomes, etc. This report consists of three sections; the users, the viewers, and Admin but this project mostly contains details on the instant cash register that I have worked on during my internship program. Petty Cash or Cash Register is a platform that gives users the opportunity to track instant transactions on the module with every details. Users also have the opportunity to add narration for every transaction. Admin has the option to add users, viewers or update their roles. The admin has access to all the user profiles, all the transactions and provides the authority to give the payable amount to the concerns for the users. Admin also has the authority to delete any transactions is he thinks that is an unnecessary transaction.

I have finished the tasks I was assigned to do, and it is functioning efficiently. Since this project has three separate elements, it is a big project, so it takes a little more time to complete all the functionality. It is still in the process of growth. I have tried to complete the tasks I have been assigned to, but at the end of the project, the features I have listed in my findings (shortcomings) will be implemented.