Enterprise Management System

A project report

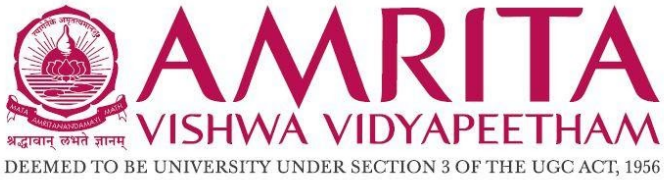
submitted as a part of the internal evaluation process for the course

15CSE313 SOFTWARE ENGINEERING

By

TEAM SAAHO

ENTERPRISE MANAGEMENT SYSTEM



DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

AMRITAPURICAMPUS (INDIA)

May 2020

Acknowledgment

We would like to express our special thanks of gratitude to our mentor Maya Menon mam and our Product owner Rahul Varma sir who gave us the golden opportunity to do this wonderful project on the topic of Enterprise Management System which helped us learn new software and know more about software engineering models. We are thankful to sir for the chance to take up this project.

Akula Sri Deepthi (RollNo: AM.EN.U4CSE17205)
Ch. Dinesh (RollNo: AM.EN.U4CSE17219)
Dheeraj Varma (RollNo: AM.EN.U4CSE17506
Tharun Ravuri (RollNo: AM.EN.U4CSE17258)
N.Harshith (RollNo: AM.EN.U4CSE17507)

Abstract

This application mainly intends on making the enterprise work ease and effortless. It focuses on managing the accounts of the clients by jotting down the transactions done by an enterprise. The transactions will be updated by the enterprise to the accountant through the website by attaching the required documents. The application is having separate websites for the enterprise and the accountant to maintain the data abstraction from the other enterprises. The enterprise website usually consists of a dropbox and a title bar where the enterprise can enter the transaction and its details. Additionally, it can also show the status of a particular transaction.

The accountant website usually consists of transactions from various enterprises. The transactions also contain relevant documents and proofs. If there are no valid documents for a particular transaction then the accountant can file a token against it. Once the transaction is verified the accountant updates the status of the transaction(rejected or pending or accepted).

By this, we can make sure that the transaction is not lost and delivered.

Contents

Chapter 1
Introduction

1. Introduction

An owner who is about to start a business is one of the busiest individuals. There are literally numerous things for him to finalize and do. They have to see to the rents agreements, utilities, storage space, permits, and licenses, suppliers, marketing campaigns, etc. It is the time they should also think of automating a few of their administrative tasks. They have the option of using *accounting software for small businesses* to lessen their burden.

A Startup Need Online Accounting Software

Keeping updated records of financial transactions is a must for businesses. However, the task is time-consuming if you try to do it manually. Accounting using pen & paper or even using spreadsheets is a thing of the past. The business IT software systems have evolved tremendously. It has brought accounting software for SME within the financial means of small business owners. Now, they don't have to shell out large sums for licensing and maintaining these systems.

The upgrading to the accounting software for SME helps the owners in making data-driven and better decisions. Using the latest financial data for decision-making leads to unbroken promises and commitments. Moreover, these web-based accounting systems allow them to email customized invoices to their client. It means that they can deliver their products to the clients' offices. And the next instant, access the software and after doing the data-entry, email it to the client and get paid early.

When you use desktop-based accounting software, it ties you off to your workstation in the office. When you are away from it, it is obvious that you cannot update your data. Moreover, you cannot collaborate with your accountant or any other colleagues.

Online accounting software is a wise investment for every CPA firm.
It is an efficient way of making your business work smarter and faster. It improves collaboration within your team and with your clients.

The desktop-based accounting software for small business may not be useful to you if your team is located at multiple locations. It can lead to data redundancy. It may also involve managing paper-files and bulky storage.

What an Accountant can do with Online Accounting System

As an accountant working in a modern CPA firm or a small company, your best bet is accounting software that can be accessed from anywhere, at any time and using any devices. Online accounting software for accountants fits the bill as it frees your accountants from the workstations in the office.

Online GST Invoicing

After you have delivered your goods or services, it is time to get paid. How do you do it? It matters as it can affect your cash flow. Use online accounting software to prepare customized GST invoices and to email them to your customer instantly.

Online Storage Management

These systems store data on web servers. It is guarded with the use of a security matrix. Only authorized users can access it. Moreover, even they get to access the data based on the need to know basis.
It puts the data at your fingertips or a mouse-click away. Your accountants can access it anytime. There is no need to maintain bulky paper-files.

Chapter 2
Project Plan

2.1 Aim

This Enterprise Management System is a solution for an accountant who accounts for the transactions of an Enterprise.

2.2 Scope

The web application provides a secured platform for an Enterprise where they can log in with a username and password and they can add/edit the transactions. Upon login, the user can view the details, status of the transactions, and his Income Tax returns.

The contents in a transaction are:

- Transaction ID
- Date
- Description
- Amount
- Type (credited/debited)
- Relevant documents (if necessary)

The web application also provides a login for an accountant. He can view the transactions of all the Enterprises he deals with. He is also able to update the status of the transaction and comment on the improper transactions (lack of documents etc). The accountant can generate the Income Tax returns of an Enterprise.

2.3 Development Process

We follow the agile methodology of software development. The product will be developed in two sprints.

Sprint 1

The core modules will be developed in this sprint which would include the following functions:

1. Users can add the transaction they had done with the following information (TransactionID, date, description, amount, relevant documents).
2. Users can edit the information of a previously entered transaction.
3. Users can view the status of the transaction.
4. The accountant is able to view the transaction of all the Enterprises.
5. The accountant can update the status of each transaction for an Enterprise (valid/not valid).

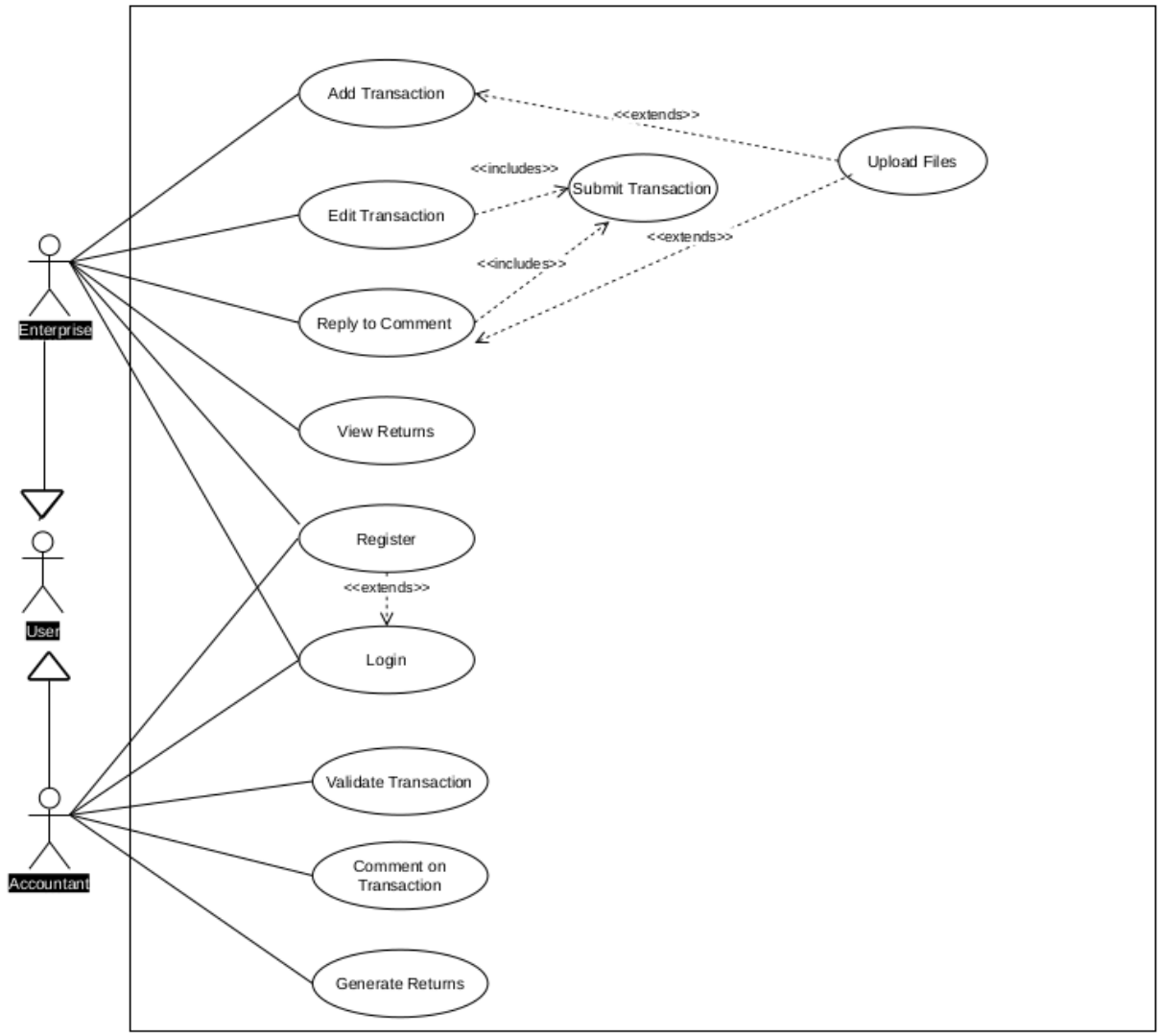
Sprint 2

The following enhancements will be done in this sprint:

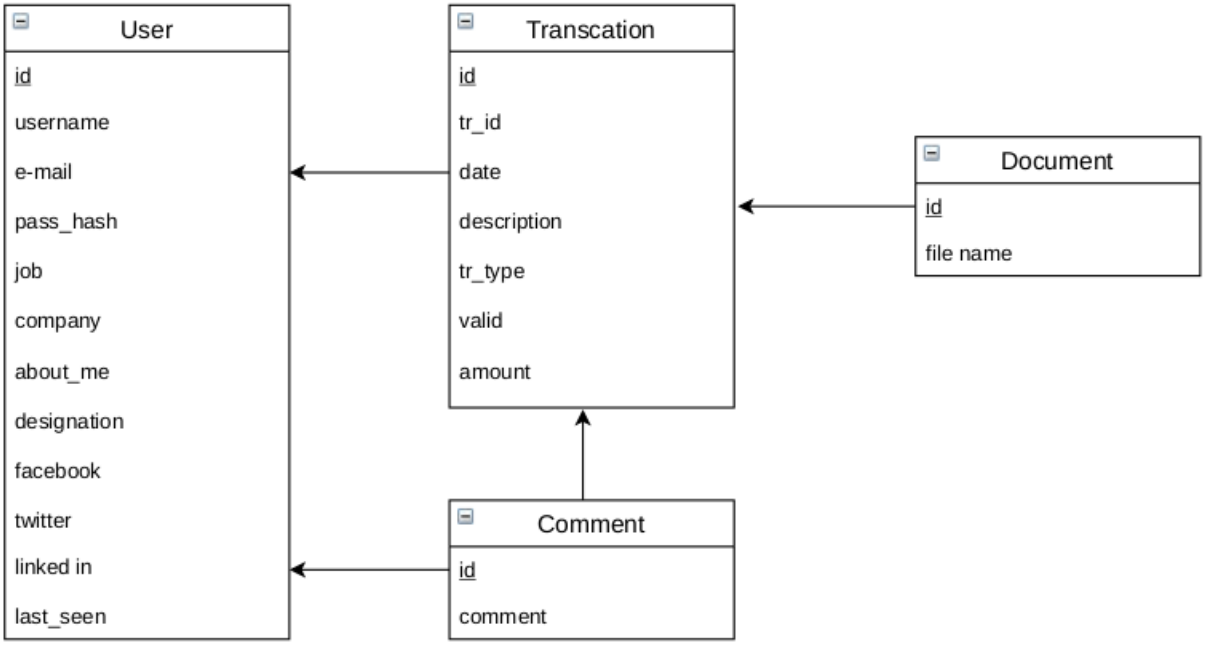
1. Enabling the accountant to comment on an improper transaction for the complete information.
2. Enabling the accountant to generate Income Tax returns of an Enterprise.

Chapter 3
Designs & Diagrams

3.1 Use case diagram



3.2 Entity Relation Diagram



Chapter 4
Technologies Used

4. Technologies Used

The technologies that are used to develop the project are mentioned in the table below:

No	Name	Technologies Used
1.	Operating System	Linux / WSL
2.	Front end	HTML,CSS,SCSS, Bootstrap
3.	Back end	Flask, Ajax
4.	Database	Sql_alchemy, Alembic
5.	Images	Vector illustrations
6.	VCS	Git
7.	Tools	Github, Sublime Text, VSC

Chapter 5
UI ScreenShots

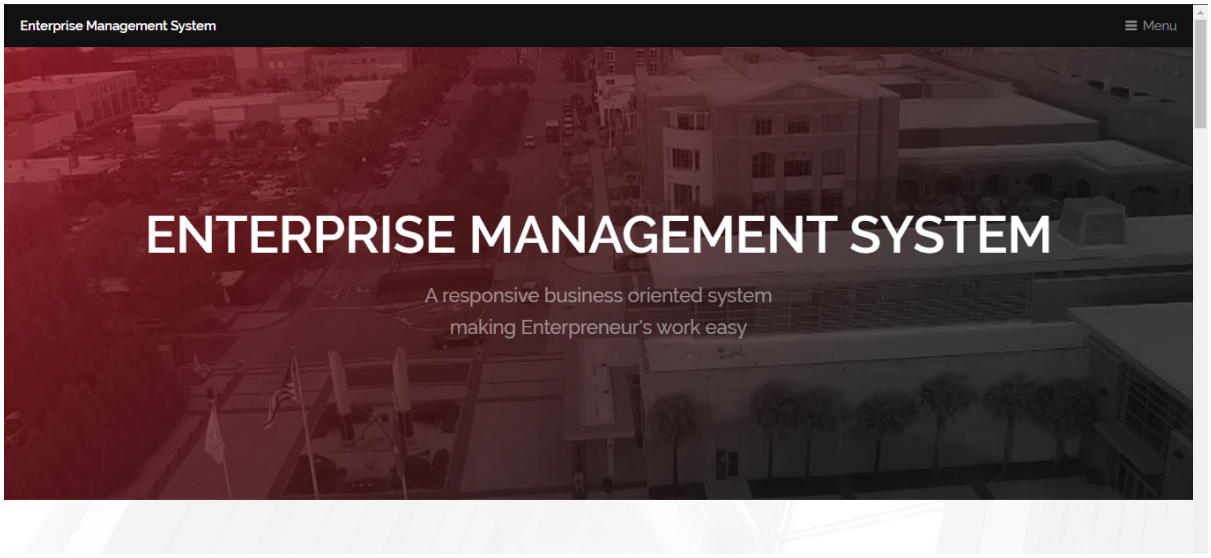


Fig 5.1: Explore page

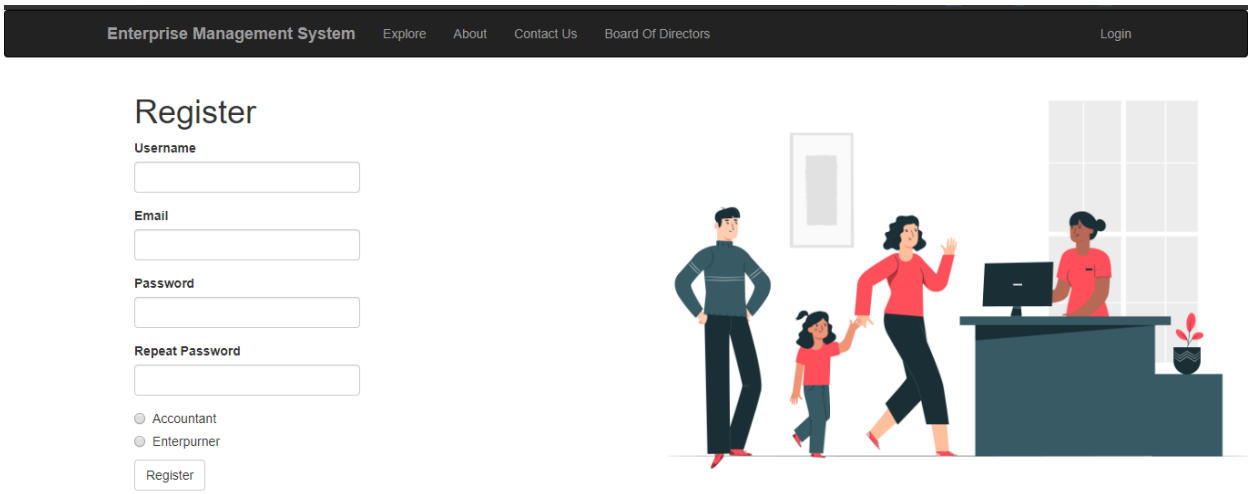


Fig 5.2: Register page

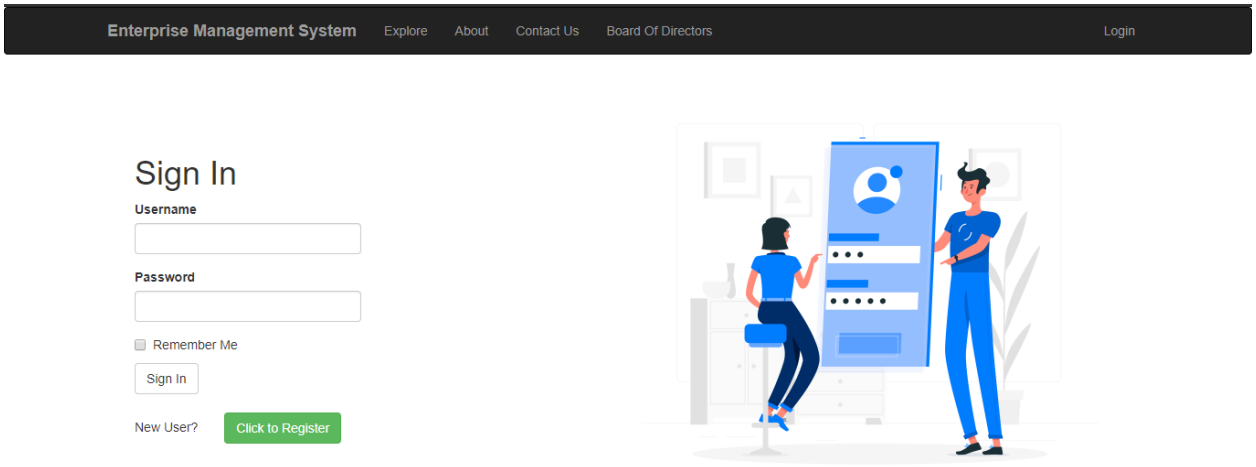


Fig 5.3: Sign In page

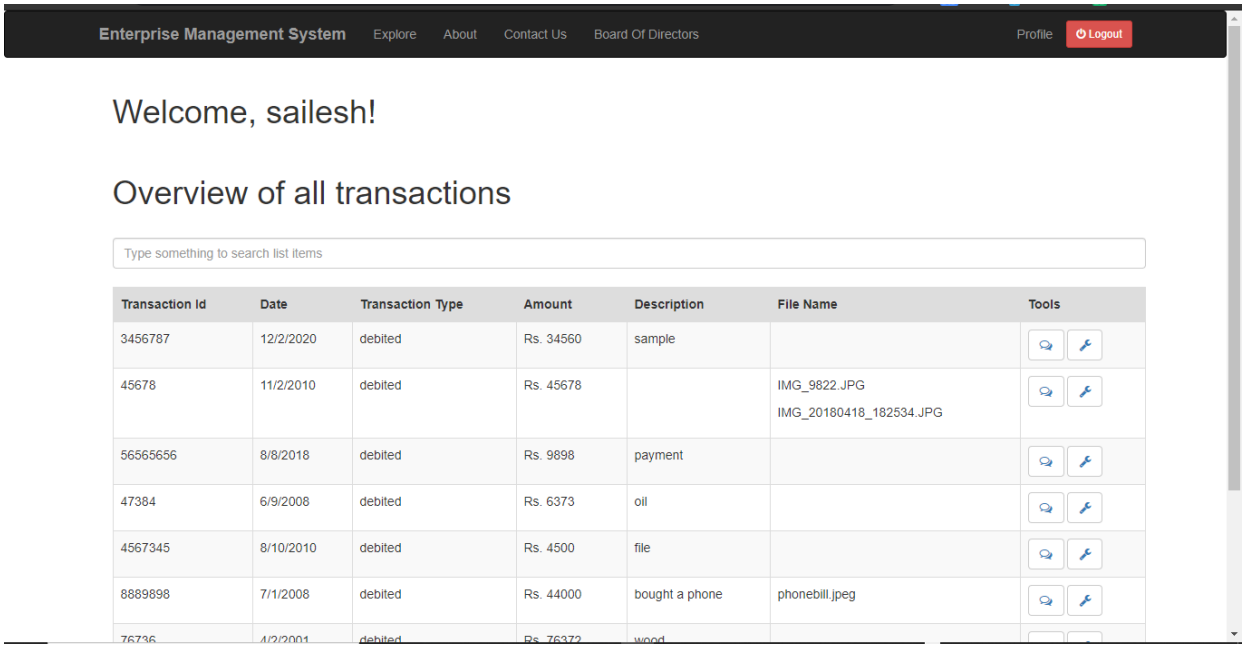


Fig 5.4: Accountant Dashboard

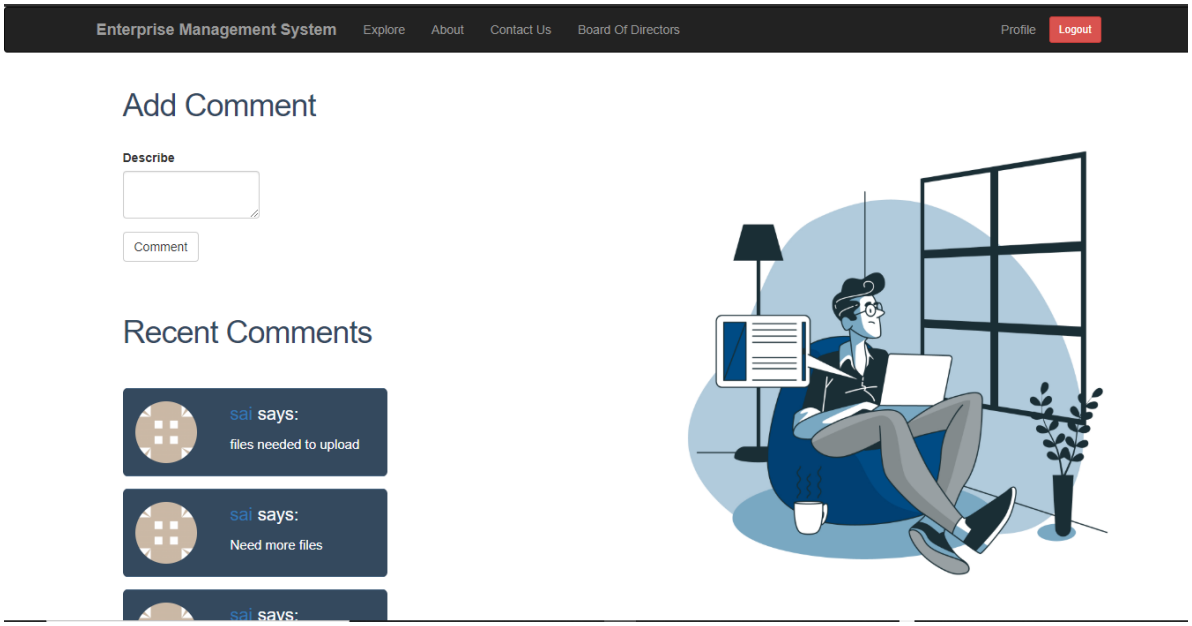


Fig 5.5: Comment page

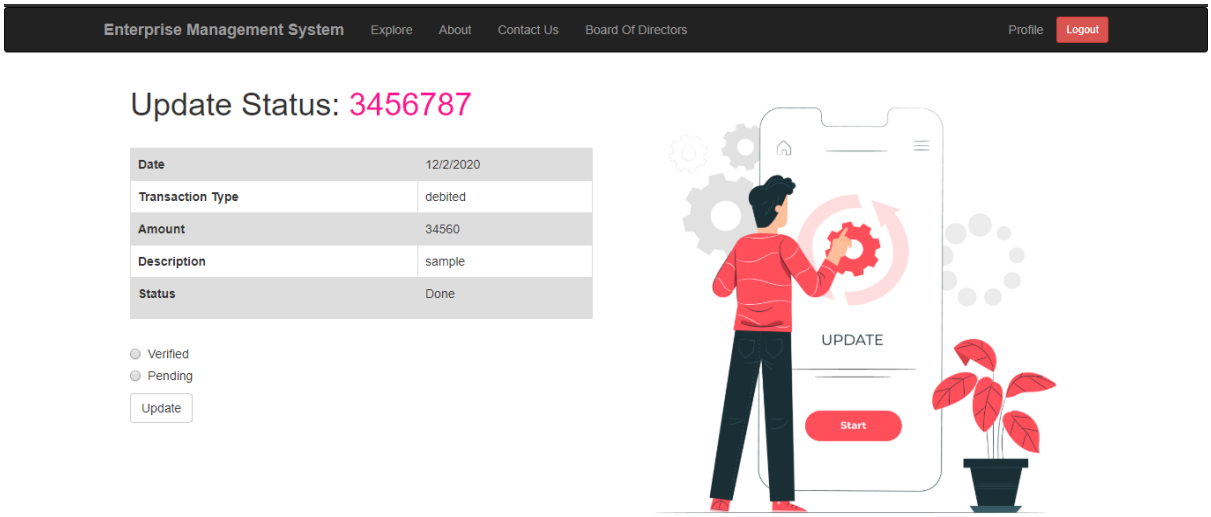


Fig 5.6: Status Page

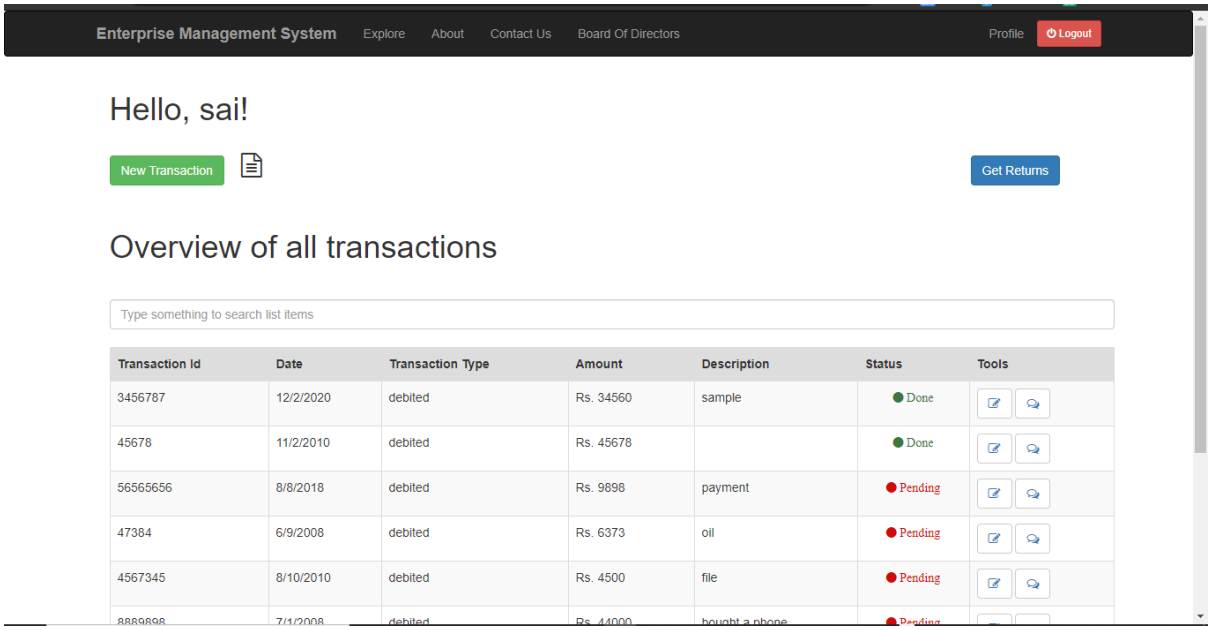


Fig 5.7: Entrepreneur Dashboard

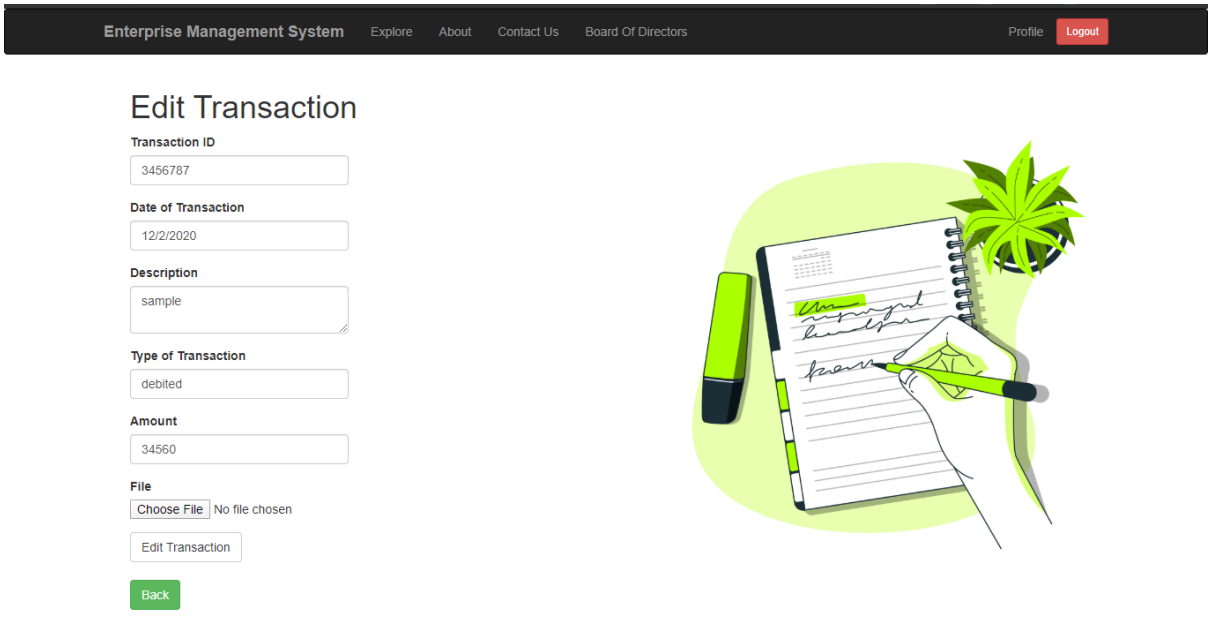
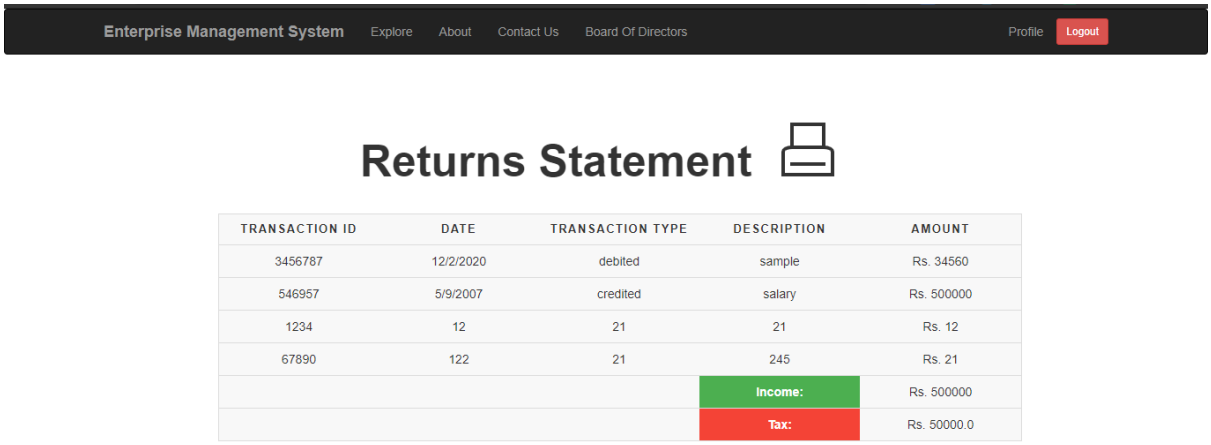


Fig 5.8: Edit transaction page



TRANSACTION ID	DATE	TRANSACTION TYPE	DESCRIPTION	AMOUNT
3456787	12/2/2020	debited	sample	Rs. 34560
546957	5/9/2007	credited	salary	Rs. 500000
1234	12	21	21	Rs. 12
67890	122	21	245	Rs. 21
			Income:	Rs. 500000
			Tax:	Rs. 50000.0

Fig 5.9: Returns page

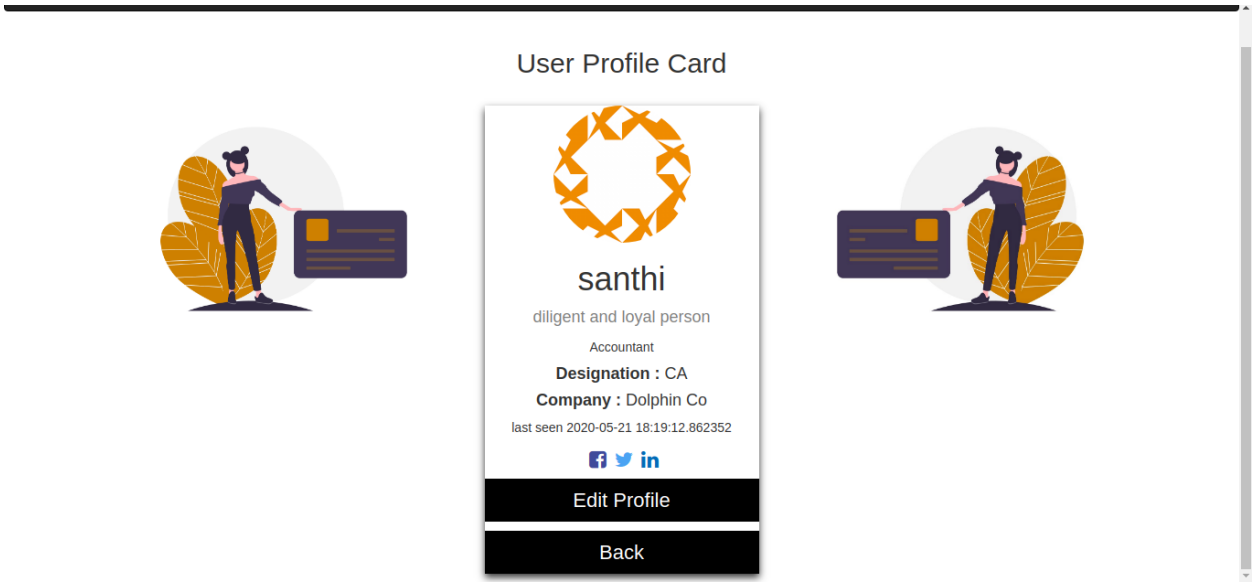


Fig 5.10: User Profile page

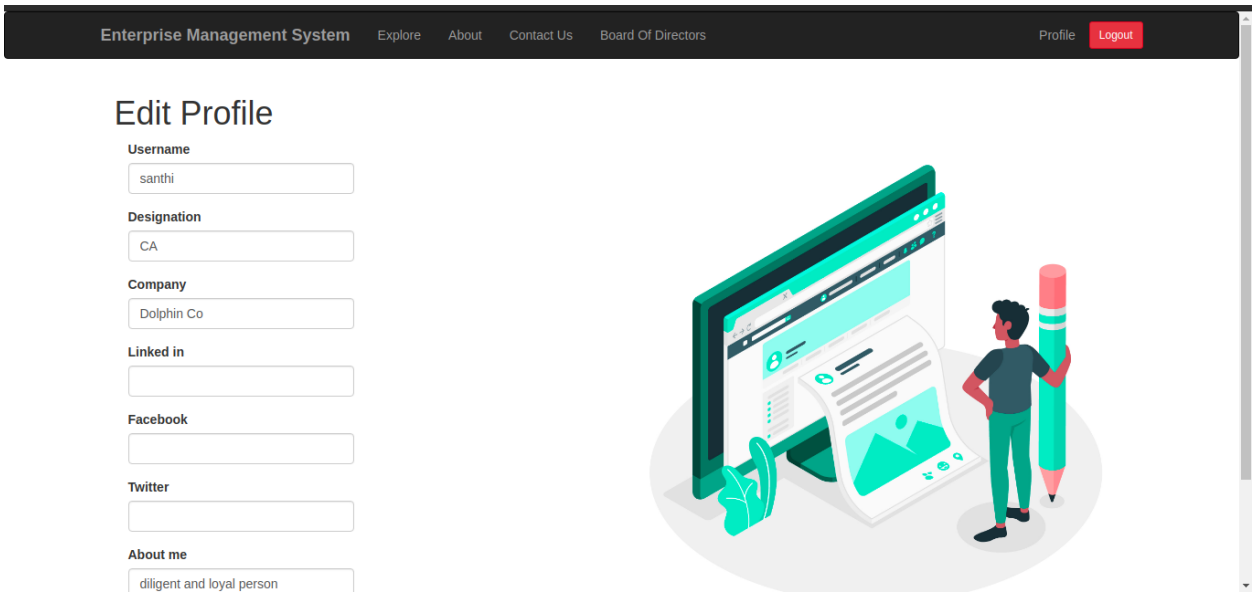


Fig 5.11: Edit Profile page

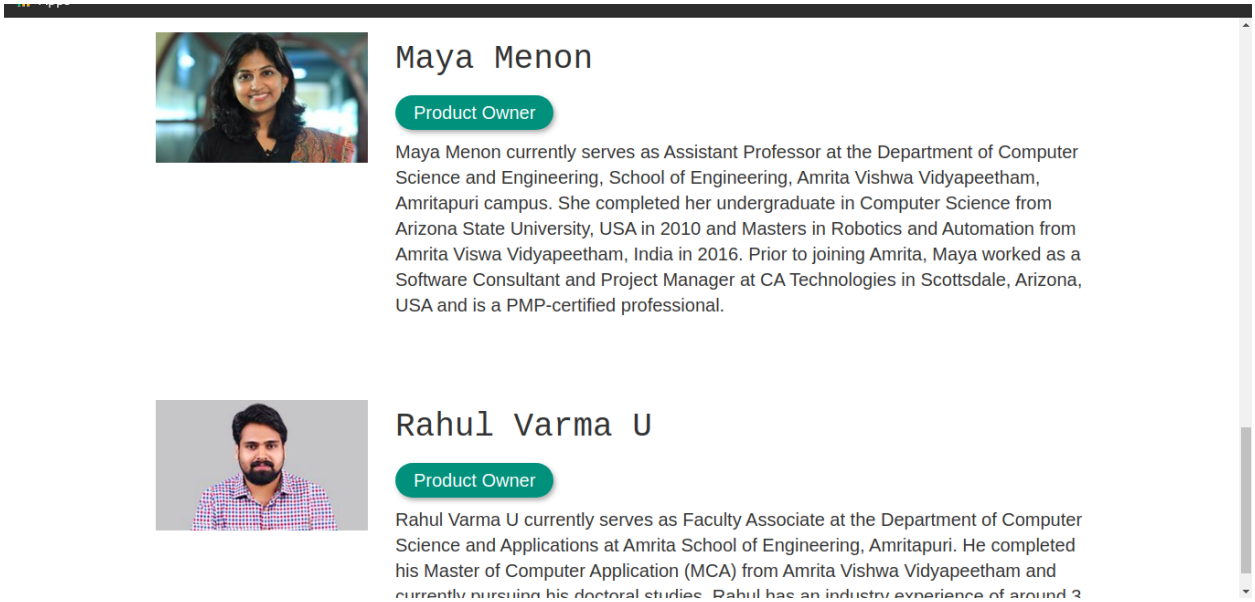


Fig 5.12: Board of Directors

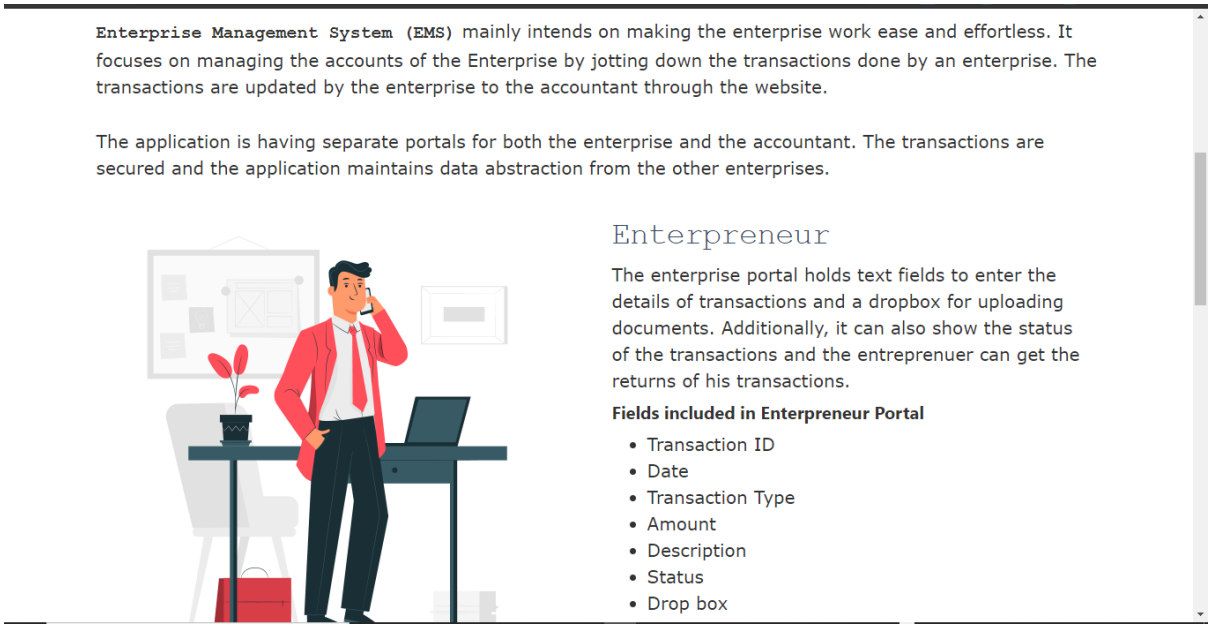


Fig 5.13: About page

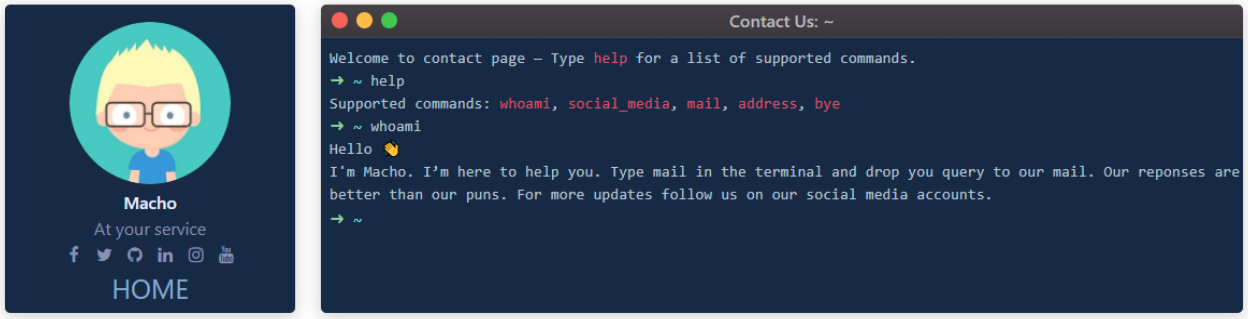


Fig 5.14: Contact page

Chapter 6
Testing Plan

Testing Strategies

The testing phase consists of

- **Unit Testing**
- **Integration Testing**
- **System Testing**

Unit Testing:
In this phase, each module is checked individually. Each module runs independently from other modules and checks the errors.

Integration Testing:
In this phase two or more modules combine and run and are checked for errors. It is two types:

1. **Top-down testing**
2. **Bottom-up testing**

System Testing:
In this phase, we run the software as a single unit. These are again types:

1. **Alpha Testing**
   This testing performed at the company level.
2. **Beta Testing**
   This testing is performed by users/beta testers who typically are end-users.

Testing Enterprise Management System

As the project(EMS) is OOAD it's beneficial to use unit testing to prevent any kind of bugs from individual modules first. Later Integration testing eliminates inter compatibility issues and any bugs when two modules work together. When all the interconnections and individual bugs of modules are fixed. Testing the overall system is the best way to test the working and get reports of any bugs that are there.

Our testing plan went in the following steps:

1. **Analyzing the product**
   In this phase we noted down all the features of the product, how users use each feature, and how each feature works.

2. **Designing the Test Strategy**
   For designing the Test Strategy we followed the following 4 steps

      1. **Defining the scope of the testing:**
         In this phase, we decided **In-Scope** and **Out-Scope** of the testing.
         In-Scope:
            - UI elements
            - Authentication & Navigation
            - Reloading the Edited Transaction
            - Income Tax generating algorithm

         Out-Scope:
            - Database merging
            - Hardware dependent APIs

2. Identifying Testing Type:

In this phase we decided to go with the **Unit Testing** method, as we have several independent classes working on individual goals in the software.

3. Risk and Issues:

If a Transaction with a file is edited a new file is uploaded to that Transaction there is a risk of losing the previous file which is added to solve this issue a new table is created for files by this every transaction can have multiple files attached to it.

4. Test Logistics:

In this phase we assigned roles for testing the software and schedule the tests after each development to a particular part of the software. For example the person responsible for UI testing will test all the pre-defined tests whenever there is an update to the UI.

1. **Defining Test Objectives**

   In this phase we listed down all the software's features that may need to be tested, and defined the target of the tests based on the features.

Features needed testing with their targets:

i. Income Tax generation: Check the Income Tax generating algorithm is calculating the tax correctly.

ii. GUI: Test the responsiveness of the dynamic elements (table, search bar, command-line interface, buttons)

iii. Adding new Transaction: Make sure the Transaction added and files attached to it (if any) are added properly.

iv. Editing existing Transaction: Test the edited Transaction is and files (if any) are updated in the database

V. File Uploading: Check the traffic that the software is able to handle for downloading

Vi. Authentication & Sign in: Check whether the account is not able to visit the Entrepreneur page and vice-versa. Also, Test for unseen vulnerabilities in Sign in & Sign up page.

2. **Defining Test Criteria**

   We used Suspension Criteria, and we kept on iterating the development and testing of a feature simultaneously until the suspension criteria were met.

5. **Resource Planning**

   Regarding Human Resource, everyone took the roles of Developer, tester and documentation as it is a small project.

   Regarding the System Resource, Most of the tools or frameworks we used for testing or developing the software are OpenSource and everyone worked in their own network and individual machines with different operating systems (Windows, Linux).

6. Plan Test Environment

The two-unit test frameworks for python that I've used are:

- **unittest** – built-in unit test framework that is based on the xUnit framework
- **pytest** – module for building unit tests

7. Schedule and estimation

We planned a schedule for testing the application. Right After adding a feature to the application we tested the specific feature using unit test frameworks and tested the application on iterations.

8. Determining the Test deliverables

Test deliverables are provided **before** the testing phase.

- Test plans document
- Test cases documents
- Test Design specifications

Test deliverables are provided **during** the testing

- Test Scripts
- Test Data
- Error logs and execution logs

Test deliverables are provided **after** the testing cycle is over.

- Test Results/reports
- Defect Report
- Installation/ Test procedures

Chapter 7
Conclusion & Future Work

7.1 Conclusion

Accounting was a tedious task back in the days but using the current technologies we could build a robust online accounting software for the companies.

Whatever the companies are whether it is a large-scale company or a small scale company. The accounting software will handle all the transactions by the Enterprise and the accountant generates the income tax returns in an efficient and accurate manner.

Benefits of Online Accounting Software

Real-Time Data

The online accounting system provides real-time data to give your accountants a clear understanding of clients' current financial position.

Multi-user

It allows multi-user access and promotes collaboration between accountants and clients.

Focus on Your Clients

The software is hosted on web servers. Your CPA firm does not have to worry about its upgrading or maintenance. It leaves you completely free to spend your time doing things important for your business.

Use Latest Version

The software is updated online. You always have access to its latest version. You do not have to install it. Your accountants access it by using its URL in the internet browser on their devices.

Auto-save

These software records drafts and asks whether to save it or not. They auto-save session data. Each bit of data is stored online.

Using a small business accounting software for an accountant is a worry-free option. You do not have to take care of version upgrades, IT team for software maintenance, costs for system administration costs, or server failures. These issues are taken care of by online accounting software providers, leaving you to do your best work for your clients.

7.2 Future Work

The current application can still have various features like:

- Multi-Currency Management
- Streamline the inventory Management
- Reconcile any business accounts
- Budget planning and forecasting
- Graphical view of tax and income
- Graphical view of valid and invalid transactions

Chapter 8
Our Take away

Notes: what you learned – your takeaway from this experience as a group & as an individual

As a Group:

This project has brought the members of the team to work together with efficiency. It has helped us develop our skills as working in a team and enhancing the spirit of teamwork. The Enterprise Management System project was a great experience for all of us since we had to do the project as individual parts and then had to merge them without conflicts. This was the most exciting part of the project where we had to sit and discuss each other's code to integrate into a single application. Working on the project and its corresponding documentation as a team has given us a great experience of successful collaboration and hence we could complete the project without any hassle.

Coming to the technical learning as a group, we learned the Flask and its features and designing the frontend of the Application. Also got revised with Database Systems which we previously learned in Semester 5. And we could also explore a lot of Python-related packages and its features.

Work Done:

Coding and Designing

- Basic workflow and Algorithm
- Database Structure design
- Implementing file uploading
- Implementing add, edit on transactions
- Implementing Comment on transactions
- Basic prototype
- Designing the UI
- Implementing the UI using HTML, CSS, JS, and Bootstrap
- Integrating Database into the application
- Adding additional features - Status, Print, Profile, and Edit profile
- Testing and correcting a few corner cases

Documentation and Diagrams

- Weekly Status Reports
- Project Plan
- Project Abstract
- Use Case Diagram
- Class Diagram
- System Requirements Specifications
- Feasibility Study
- Testing Strategies and Methods
- Future Work

Ch Dinesh

Worked on Database Designing

I worked on the database of our project. Databases play a major role in every project. As a beginner to flask initially I was confused about how to design the database and I faced a huge challenge in writing the database as classes in a flask and understanding the underlying concepts like ORM(Object Relational Mapping) and all. The database was not updated when I tried to add more columns in an existing database and also when I tried to migrate the scripts I faced issues with the database.

What I learned regarding this project as an individual was implementing databases and designing it. I understood how ORM(Object Relational Mapping) and how it works and its underlying properties and I also understood how the alembic and SQL_Alchemy work when we are working with flask.

As a group I came across to know about the importance of group discussions and helping each other in hard times, work distribution and it makes work faster and it is possible to complete in time.

Tharun Ravuri

Worked on Frontend development

I worked on the front-end of the application. I used Bootstrap as a framework for frontend and I used languages like HTML, CSS and Js for designing the application.

The challenges that I faced during the development of this application were Initially I am not an expert in UI/UX but eventually I improved myself and understood various concepts and properties in CSS.

And installing the application in the local PC became a difficult task but somehow I resolved the issue by using references like stack overflow.

As an individual, I learned about CSS, and also concepts behind UI/UX

and As a group I understood how time management is important in life and also I am glad to apply all the concepts and principles of SE in real-time projects.

Harshith

Worked on Backend Development

I worked on the Back - end of this project. I used flask as a back-end and SqlAlchemy for databases, I have also used Gavatars for the profile pictures and I also used ajax in this application.

As I am a newbie for flask I faced many issues while setting up the environment and developing the application. I learned flask from the flask mega tutorial written by Miguel Grinberg. Usually I faced issues with my database and StackOverflow resolved most of my issues.

As a group I've learned how our software engineering models work and how to apply it.

Dheeraj Varma

Worked on testing the application

I worked on the testing of the project and finding the bugs in the application became a big challenge for me.

At the initial stage of the project we found that the user can access the content even if he was not logged in. And the accountant is able to access the entrepreneur page. So I tested the application each and every time and found various bugs and our team resolved them.

What I learned regarding this project as an individual was implementing waterfall development in the technical projects.

As a group, I came across to know about the importance of scrum meetings and sprint reviews

Deepthi

Worked on Integration of the application

I worked on the Version Control System of our project. VCS plays a major role in every project. As a beginner to VCS initially I was confused about how to understand the workflow of git. I faced huge challenges in understanding the branch workflow. As the project develops there is a huge code and it was difficult for me to handle the branches and revert it.

What I learned regarding this project as an individual was understanding the workflow of git and the major concepts like a branch and its underlying properties. As a group I can learn about developing ideas of everyone. And working together and gain more knowledge and developing the technical skills