WORKSSALE

SAAS BASED SALES MANAGEMENT SOFTWARE

a project report submitted by

MANDUVA LEELA SUMANTH (URK17CS230)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

under the supervision of

Dr. J. JEBAVEERASINGH, M. Tech., Ph.D.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF ENGINEERING AND TECHNOLOGY

KARUNYA INSTITUTE OF TECHNOLOGY AND SCIENCES
(Declared as Deemed-to-be-University under Sec-3 of the UGC Act, 1956)
Karunya Nagar, Coimbatore - 641 114, India.

April 2021

ACKNOWLEDGEMENT

First and foremost, we praise and thank **ALMIGHTY GOD** for giving us the willpower and confidence to carry out our project.

We are grateful to our beloved founders Late **Dr. D.G.S. Dhinakaran**, **C.A.I.I.B, Ph.D.,** and **Dr. Paul Dhinakaran**, **M.B.A., Ph.D.,** for their love and always remembering us in their prayers.

We extend our thanks to **Dr. P. Mannar Jawahar, Ph.D.,** our honorable vice-chancellor, **Dr. E. J. James, Ph.D., Dr. Ridling Margaret Waller, Ph.D.,** and **Dr. T. Lazar Mathew, Ph.D.,** our honorable Pro-Vice-Chancellor(s), and **Dr. R. Elijah Blessing, Ph.D.,** our respected registrar for giving us the opportunity to carry out this project.

We are thankful to **Dr. G. Prince Arulraj, M.E., Ph.D.,** Dean (Engineering & Technology) for his support and encouragement.

We would like to place our heartfelt thanks and gratitude to **Dr. J. Immanuel Johnraja, Ph.D.,** HOD, Department of Computer Science and Engineering for his encouragement and guidance.

We are grateful to our guide, **Dr. JEBAVEERASINGH J, Ph.D.** Professor, Department of Computer Science and Engineering for his valuable support, advice, and encouragement.

We also thank all the staff members of the Department for extending their helping hands to make this project work a success.

We would also like to thank all my friends and my parents who have prayed and helped me during the project work.

ABSTRACT

In the age of big data, sales analytics has become an important part of every sales report. Instead of manually gathering analytics through a tangle of spreadsheets, sales analytics software allows sales managers to access it systematically in one place. Effective sales management plays a crucial role in the growth of the organization.

Poor management of resources results in high production costs. It is difficult to manage sales of multiples products in multiple markets and to achieve their individual sales target. Investing in an unsuccessful product brings the organization down.

Manually collecting all the sales performance papers of a particular product and creating a performance report is very difficult and consuming. It is where WORKSSALE comes in handy, it gathers all the information like customer contact details, sales details, sales reports, expenses report in one place. This tool is mainly focused on medium-sized businesses

Works Sale is an efficient point of sales tool for the management of sales and purchases. It facilitates the achievement of sales targets in multiple branches through easy monitoring. This tool is easy to use and effective for organizations of varied sizes. There are many sales tools categories such as CRM software, sales acceleration software, video conferencing software, marketing automation software, etc. This project mainly focuses on CRM and Sales acceleration.

Table of Contents

LIST OF FIGURES

LIST OF ABBREVIATIONS

IT	Information Technology
CRM	Customer Relationship Management
MTV	Model Template View
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
IDE	Integrated Development Environment
VS	Visual Studio
CSRF	Cross-Site Request Forgery
ER	Entity Relationship
URL	Uniform Resource Locator

Chapter-1

Introduction

1.1 Overview:

Sales tools are applications and software that enable sales professionals and managers to effectively track, evaluate, and enhance the performance of their sales activities while predicting trends and results, and finding relevant insights.

In the age of big data, sales analytics has become an important part of every sales report. Instead of manually gathering analytics through a tangle of spreadsheets, sales analytics software allows sales managers to access it systematically in one place.

The adoption of sales analytics software is growing rapidly throughout the business world. In the past, mainly larger enterprises took advantage of this type of technology. Thanks to the rise of Sales operations roles in small and medium-sized businesses, the adoption of sales analytic tools has grown in this segment lately.

1.2 Problem Statement:

Within a traditional manual system, salespeople prepare their forecasts by reviewing current accounts and overall projected sales. The time spent on forecasting is huge. The time spent forecasting is less time spent selling. In more data-driven processes, a company often has marketing, IT, and sales staff involved in building a system to collect and interpret data.

Effective sales management plays a crucial role in the growth of the organization. Poor management of resources results in high production costs. It is difficult to manage sales of multiples products in multiple markets and to achieve their individual sales target. Investing in an unsuccessful product brings the organization down.

This means that companies using conservative or old-fashioned sales methods not only risk losing sales but even seeing their best sales talent leave them. To make a reporting staff has to gather all the files related to the products and have to analyze every file, which is very time-consuming. Even though if they want to report about a single product, they have to go through all the related documents.

1.3 Objective:

Works Sale is an efficient point of sales tool for the management of sales and purchases. It facilitates the achievement of sales targets in multiple branches through easy monitoring. This tool is easy to use and effective for organizations of varied sizes.

There are many sales tools categories such as CRM software, sales acceleration software, video conferencing software, marketing automation software, etc. This project mainly focuses on CRM and Sales acceleration.

1.5 Chapter Wise Summary:

Chapter 1: It consists of basic information of the internship company, problem statement, overview, and objective of the project.

Chapter 2: Consist of a complete analysis of the existing system and the proposed system along with the use case analysis of login and admin module and Requirement Specification for the product.

Chapter 3: Gives brief information about the detailed design of the product, modules implemented. It also shows how to database is implemented in the back end.

Chapter 4: This focuses on the information regarding each module implementation in depth.

Chapter 5: Contains the project status and the source code of the project.

Chapter 6: Shows sample screenshots of the projects. It gives a basic idea of how the project is implemented.

Chapter 7: Describes the conclusion of the project along with how it can be enhanced in the future.

Chapter-2

System Analysis

2.1 Existing System:

Sales tools refer to digital tools used by sales professionals to make their work easier. Sales tool categories include CRM, sales intelligence, sales acceleration, sales gamification, sales analytics, video conferencing, e-signature, account-based marketing, marketing automation, and customer service software.

Sales tools help salespeople understand which prospects are most likely to be successful with their product or service, when to reach out to those prospects, and what they should be talking about with them. They save time on time-consuming administrative tasks and can focus on what matters the most: providing value for people and businesses.

For example, Nutshell is a web and mobile CRM service. It is composed of a web application, as well as mobile applications for the iOS and Android platforms. In addition to that, it has different tools like Sales automation, Team Collaboration, Marketing, etc.

2.2 Proposed System:

Manually collecting all the sales performance papers of a particular product and creating a performance report is very difficult and consuming. It is where WORKSSALE comes in handy, it gathers all the information like customer contact details, sales details, sales reports, expenses report in one place. This tool is mainly focused on medium-sized businesses.

The aim is to automate its existing manual system with the help of computerized equipment and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same.

The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces manual work. Some of the advantages of the proposed system are:

- Minimize manual data entry.
- Security of data.
- Security of data.
- User-friendliness and interactive.

2.2 Use case Analysis:

The purpose of the use case to define a piece of coherent behavior without revealing the internal structure of the system. A use case typically represents a sequence of interactions between the user and the system. These interactions consist of one mainline sequence is represents the normal interaction between the user and the system.

The use case model is an important analysis and design art craft(task). Use cases can be represented by drawing a use case diagram and writing an accompanying text elaborating the drawing.

In the use case diagram, each use case is represented by an ellipse with the name of use written inside the ellipse. All the ellipses of the system are enclosed in a rectangle which represents the system boundary.

The name of the system being module appears inside the rectangle. The different users of the system are represented by using stick person icons. The stick person icon is normally referred to as an Actor.

The line connecting the actor and the use cases is called the communication relationship. When a stick person icon represents an external system, it is annotated by the stereo type<<external system>>.

Use case diagram:

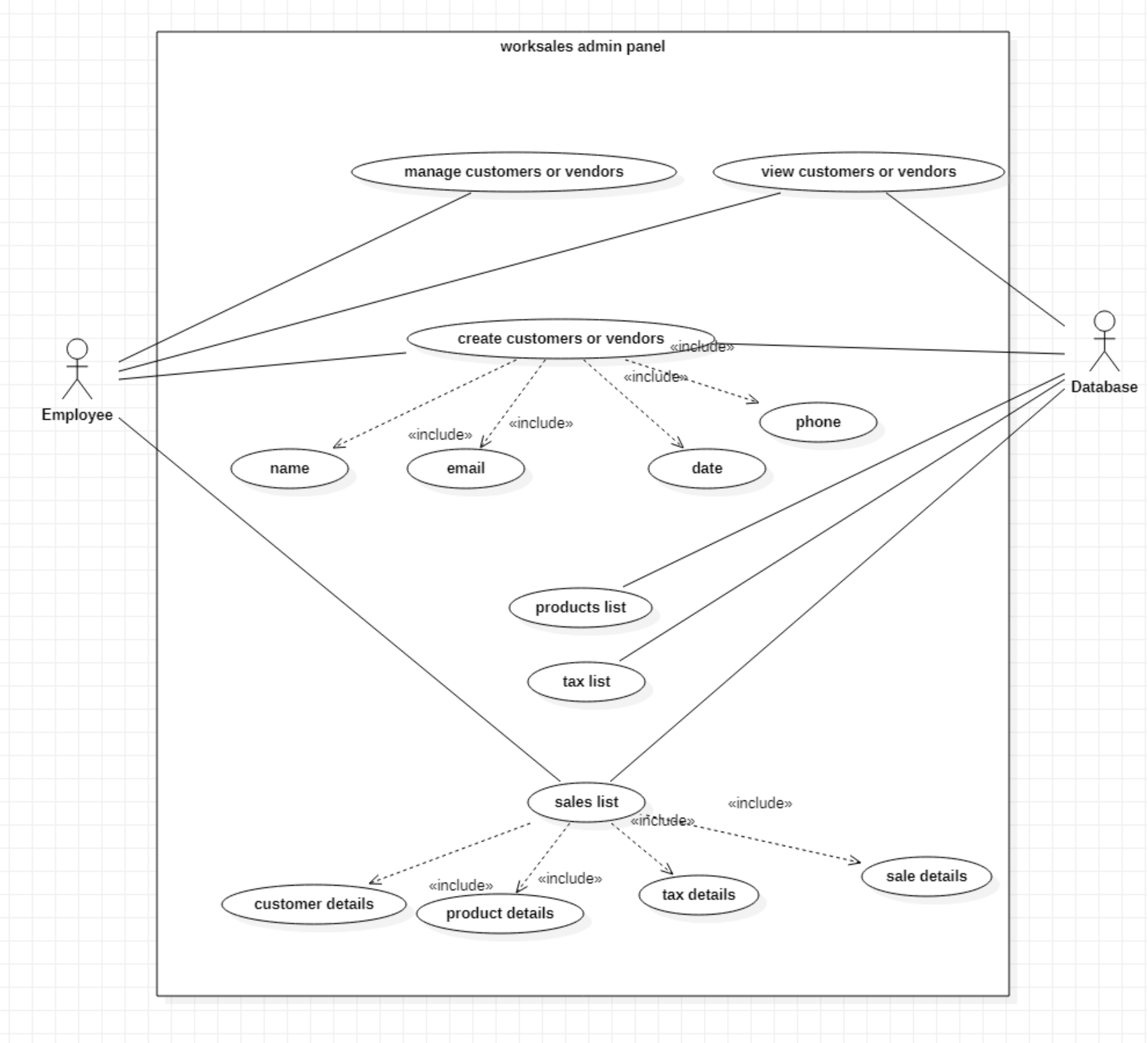


FIG 2.2.1 USE CASE DIAGRAM FOR CUSTOMERS AND SALES

Actor - Employee, Database

Methods - manage customer or vendor, view customer or vendor, create customer or vendor, sales list, product list, tax list

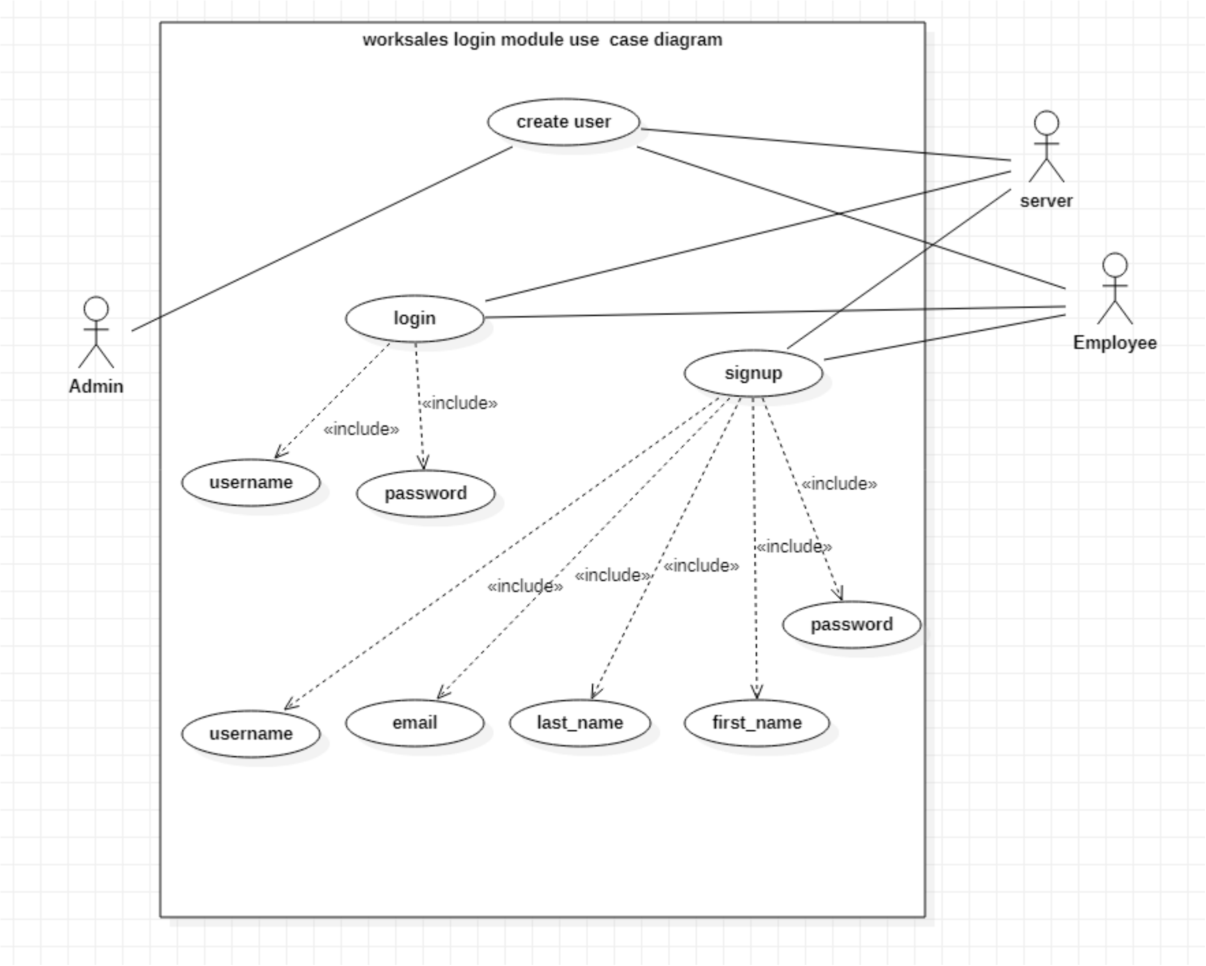


FIG 2.2.2 USE CASE DIAGRAM FOR LOGIN AND SIGNUP MODULE

Actor - Employee, Database, Admin

Methods - create user, login, signup

2.3 Requirements Specifications:

2.3.1 FUNCTIONAL REQUIREMENTS:

- Sales Tracking
- Admin access control
- Authentication via mail
- Wrong password lockouts
- Sales report

2.3.2 NON-FUNCTIONAL REQUIREMENTS:

- Performance
- Scalability
- Capacity
- Availability
- Reliability
- Recoverability
- Maintainability
- Serviceability

2.3.3 HARDWARE REQUIREMENTS:

Processor: Minimum 1 GHz

Network: Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

Hard Drive: Minimum 32 GB.

Memory (RAM): Minimum 1 GB

2.3.4 SOFTWARE REQUIREMENTS:

The system will be provided through a web browser, and it will be compatible with most of
the widely used ones if not all. Thus, it will be independent of the operating system of the
computer on which it runs.

Technologies used-

1. Django framework for Python - Django is a Python-based free and open-source web framework that follows the MTV architectural pattern used to create web applications. architectural pattern.
2. MySQL for storing Database – MySQL is an open-source relational database management system.

Tools used-

1. Visual Studio Code for HTML5, CSS3, JAVASCRIPT,BOOTSTRAP
2. Python IDE

Operating System: Windows

Front-End : HTML, CSS, Bootstrap

Back-End: Python

Database: MYSQL

Code Editor: Visual

Chapter-3

System Design

3.1 Project Main Focus

- Effective sales management plays a crucial role in the growth of the organization.
- This project mainly focuses on CRM and Sales acceleration.

3.1.1 CRM

- CRM stands for "Customer Relationship Management" and refers to all strategies, techniques, tools, and technologies used by enterprises for developing, retaining, and acquiring customers.
- This software ensures that every step of the interaction with consumers goes smoothly and efficiently to increase the overall profits. The software gathers customer data from multiple channels.
- These are just a few capabilities that CRM features - Storing all customer information in one place, recording service issues, identifying sales opportunities, managing marketing campaigns

3.1.2 Sales Acceleration

- Sales acceleration tools let one track the entire sales process from lead generation to closing. This provides important data beyond basics like conversion rates.
- It provides insights about how the products are performing in the market and also about customers.
- It shows which members of your sales team have the greatest success, but other metrics give you deeper insight into the way your team functions.
- All of this leads to increased productivity. Work that is high in value and requires a high touch should be accelerated.
- Team members save time since they don't have to engage in repetitive tasks or waste time interacting with unqualified leads who have no intention to buy.

3.2 Implementation of Different Modules

- Two modules have been implemented in this project:
  1. Admin Panel:
     - This admin panel is for the managers of the organization. To check on sales and expenses.

  2. Super Admin Panel:
     - Super admin panel is a track of all the organizations that are subscribed to the sales tool.

3.2.1 Admin Panel:

Dashboard:

- Sales dashboards translate data into easily consumable pieces of information, providing you with instant business insight.
- This module includes information like sales of the current month, total sales amount, Purchase of the current month, total purchase amount.
- It shows a graphical representation of how sales and purchases are going.
- There are different charts to display information like sales, expenses, and present stock of a product.
- A calendar is included to help the staff providing a glance at upcoming activities.
- And also a to-do list is provided to help the staff to write down the tasks to be performed and can prioritize them easily.
- By this module staff will have the information on how the sales are performing and the future activities, events they have to plan and can prioritize.

Landing Page Panel:

- This page shows the basic information about the tool.
- It consists of four sections, in the first one shows brief information about the tool.
- In the second one, features of the tool are displayed.
- In the third section, screenshots of the tool are displayed. It gives a basic idea for the organization on how the tool works and features.
- Then in the last section, pricing details are displayed.

Staff Panel:

- It shows details of users like what role they playing in the tool.
- From here admin can create a user or can modify the roles they are playing.
- Admin has rights to create a user and can specify they can access the module or not.
- Admin can categories them using the search field based on their roles.
- The second module shows the details of the roles and permissions assigned to each role.
- Admin can create a new role for the users or he/she edit different permission of roles by clicking the edit button and can also delete the role entirely.
- The permission of each role includes creating, managing, or deleting Profiles, customers, vendors, etc.

Customer Panel:

- It shows the list of customers to the admin.
- Basic details of customer like email Id are shown. So that the staff can be able to contact them immediately in case of any issue.
- It also shows when the customer is added i.e. the date of the placement of an order by the particular customer.
- Admin has the right to manage the customer list. Can be able to add new customers and can delete the old ones.

Vendor Panel:

- This panel is similar to the Customers panel but it shows the details of different vendors and their contact details.
- Admin can create, delete or edit vendor details in case of any miscommunication.

Products Panel:

- The products panel shows detailed information about the products.
- From here Admin will be able to know how much stock is left for each product.
- All the products are categorized based on their type and brand.
- Admin can include brands of the products if they have subbrands.
- Admin can easily be able to filter the products based on the branding, type of category, or name.
- If the stock is zero, the indication is easily shown in the table with a red label. So that admin can be easily able to find which stock is low.
- Admin has permission to add a new product, fix a tax percentage, or can edit the prices of products if their demand gets low.
- In this, there are other modules that Admin can access that are categories panel and tax pannel.
- In the categories panel, Admin can create new categories based on the products they are launching.

- In the tax panel, all the details about different tax percentages are shown.
- Admin can set if the tax is the default he can select yes or if the tax is based on the region can select no.
- Admin can also add new tax if they are expanding to different markets where tax may vary.

Sales Panel:

- Here the basic details of the products that are currently in the sale are shown like product picture, price, and category they belong to.
- Then in the sales list tab details of various sales will be shown.
- Here admin is provided with information like the invoice id of the payment, to whom the item is sold, the number of items sold, and the payment status.
- If the vendor's payment has been declined then the admin can immediately send a notification to the vendor through the contact details provided in the vendor tab.
- Here additional information is also inserted so that it can be used to show the reports for the admin.
- These sales data are also used in charts.

Reports Panel:

- This panel provides a detailed view of products and customers.
-

  Stock Analysis Panel:
  - Here the stock details of the products are shown to the admin.
  - If a product that is performing well is low in stock, from here admin can be easily known and he will be able to manage more resources to the product.

  Customer Analysis Panel:
  - Here the contact details of the customer will be shown.
  - It also provides a view of total sales done by customer or vendor

  Tax Panel:
  - In the tax panel, the tax amount of the sales done previously is shown.
  - In addition to that, the total purchase amount of the month and year are shown here.
  - In the table total amount of the sale is shown after the tax inclusion.

Expenses Panel:

- Details of company expenses are shown here

Expense List Panel:

- In the expense list panel, all the expenses of the company are shown in one place.
- All the expenses are categorized based on type and branch.
- This panel is used for the business which sells their products in a large market.
- All the expenses of various branches are shown here.

Expense Category Panel:

- In this panel, all the expense categories are listed.
- Admin has the privilege to edit, create or delete a category based on their expenses.

Admin Profile Panel:

- Here the admin details are shown like mobile number, email id, and location.
- Admin will be provided an option to change passwords and personal details in this tab.

3.2.2 Super Admin Panel:

Dashboard:

- It shows the sales of the tool and shows how it is competing in the market.
- It shows the no of subscriptions in the month and year.
- This module includes information like sales of the current month, total sales amount, Subscriptions of the current month, total purchase amount.
- It shows a graphical representation of how sales are going.
- A calendar is included to help the staff providing a glance at upcoming activities.
- And also a to-do list is provided to help the staff to write down the tasks to be performed and can prioritize them easily.
- By this module staff will have the information on how the sales are performing and the future activities, events they have to plan and can prioritize.

Owners List Panel:

- Here the list of the tool subscriber is shown.
- It also includes the contact details of the company that subscribed to the tool.
- No users using the tool in the subscribed organization are also shown here.
- It also gives a detail of the plan that the company subscribed to and the expiry date of the plan.
- Here according to the organization's request, the plan can be canceled.
- From here admin can create a user or can modify the roles they are playing.
- Admin has rights to create a user and can specify they can access the module or not.
- Admin can categories them using the search field based on their roles.

Landing Page Panel:

- This page shows the basic information about the tool.
- It consists of four sections, in the first one shows brief information about the tool.
- In the second one, features of the tool are displayed.
- In the third section, screenshots of the tool are displayed. It gives a basic idea for the organization on how the tool works and features.
- Then in the last section, pricing details are displayed.

Plans Panel:

- Here a brief view of all the plans that are offered by the tool is displayed.
- Organizations can select the plan based on the number of customers they are having.
- There are four different plans offered here ranging from the organization having 20 to n number of customers.

Coupon Panel:

- Here the list of coupons provided by the tool is displayed.
- Coupons are provided to the organization who is a subscriber for more than 3 years and also for the new customers.
- All the coupon details including discount percentage and no of organizations used the coupon and also the limit.

Orders List:

- Subscription history, payment status, the type of plan, date of purchased all the details are displayed here.

3.3 Software's and Technologies used in the Project:

- **HTML-Hypertext Markup Language (HTML) :**

    - The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.
    - It is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.
    - HTML is heavily used for creating pages that are displayed on the world wide web. Every page contains a set of HTML tags including hyperlinks which are used for connecting to other pages.
    - HTML is parsed by the browser which renders the webpage to display. It isn't a programming language.

- **CSS-Cascading Style Sheets (CSS):**

    - CSS is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.
    - It is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers**.**
    - CSS is independent of HTML and can be used with any XML-based markup language.
    - There are three ways you can use to implement CSS: internal, external, and inline styles.

- **JAVASCRIPT:**

    - JavaScript is a scripting language. It is different from the Java language. It is an object-based, lightweight, cross-platform translated language.

- It is widely used for client-side validation. The JavaScript Translator (embedded in the browser) is responsible for translating the JavaScript code for the web browser.

- It is used when a webpage is to be made dynamic and add special effects on pages like rollover, roll out and many types of graphics.

- To execute JavaScript in a browser you have two options either put it inside a script element anywhere inside an HTML document, or put it inside an external JavaScript file

- **BOOTSTRAP:**
  - Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development.
  - It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.
  - Bootstrap is great for quickly throwing together a prototype or internal tool, but it is not intended for use on a production website.
  - Bootstrap 4 is the newest version of Bootstrap; with new components, a faster stylesheet, and more responsiveness. Bootstrap 4 supports the latest, stable releases of all major browsers and platforms.

- **PYTHON:**
  - Python was created by Guido van Rossum, and released in 1991. It is a general-purpose computer programming language.
  - It is a high-level, object-oriented language that can run equally on different platforms such as Windows, Linux, UNIX, and Macintosh.
  - It is widely used in data science, machine learning, and the artificial intelligence domain.
  - It's often used as a “scripting language” for web applications. This means that it can automate specific series of tasks, making it more efficient.

- **Virtual Studio Code:**
  - Visual Studio Code is a freeware source-code editor made by Microsoft. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.
  - Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.
  - It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.
  - The extensions make VS Code an excellent Python editor and work on any operating system with a variety of Python interpreters.

- **DJANGO:**
  - Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.
  - Built by experienced developers, it takes care of much of the hassle of Web development, so one can focus on writing an app without needing to reinvent the wheel. It's free and open source.
  - It is maintained by the Django Software Foundation, an American independent organization established as a 501 non-profit.
  - While PHP is specifically designed for web designs, Django bases on a more robust language. So, to write a good code, it is easier to do it in python compared to PHP.
  - Django provides security features like
    - Clickjacking protection
    - Cross-site scripting protection
    - CSRF protection
    - SQL injection protection

- **MySQL:**
  - MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database.
  - MySQL is open-source and free software under the GNU license. It is supported by Oracle Company.
  - It is a fast, scalable, and easy-to-use database management system in comparison with Microsoft SQL Server and Oracle Database.

3.3 Database Design:

- The database consists of 12 tables. The main tables are the staff, customer, vendors, and sales list tables. The remaining tables all point to the sales list, customers, vendors, expenses tables.

Structure of Database:

Table 3.3.1 Structure of Database

Database Table	Description
userslists	Contains information about the staff
todolists	Contains information about the upcoming events
roleslists	Contains information about specific roles and their permission list.
customerlist	Contains basic customer information and contact details.
vendorlists	Contains basic vendor information and contact details.
productslists	Contains information about different products sold by the organization.
categorieslists	Contains information about different product categories
catagorieslists	Contains information about different categories of the products.
taxlists	Contains information about all the taxes percentages imposed on different products
saleslists	Contains purchase information of different products.
expenceslists	Contains expenses information of the company
exp_categorieslists	Contains information about categories in which the company spends.

ER Diagram:

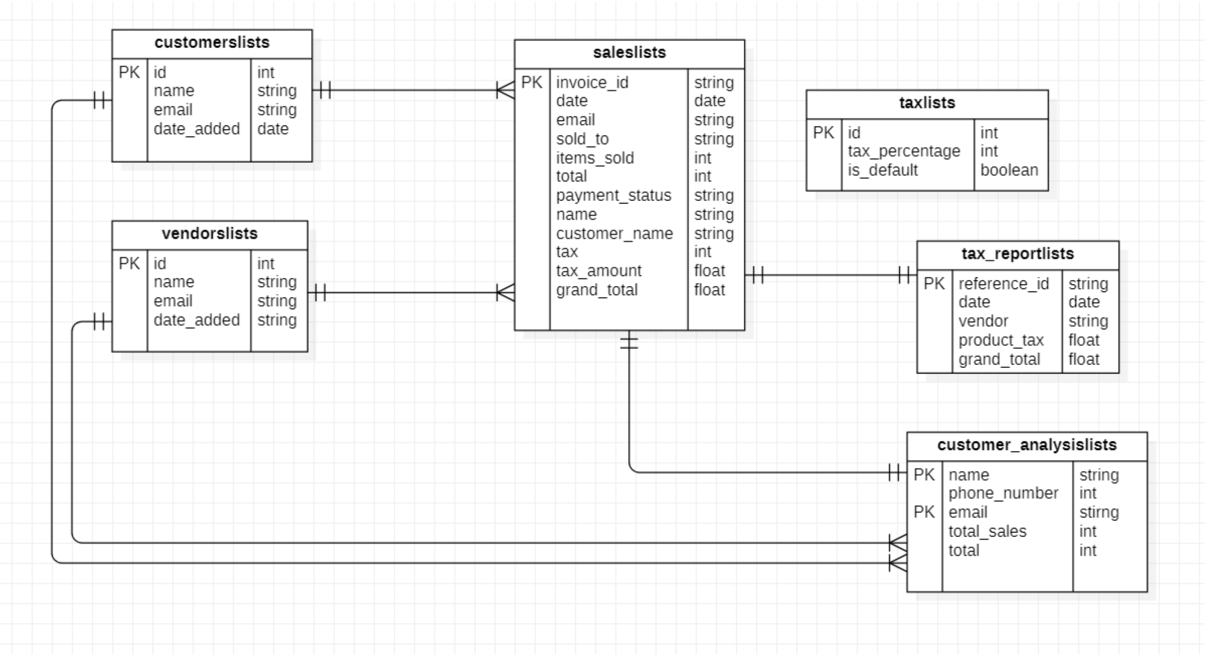


FIG 3.4.1 ER DIAGRAM FOR SALES AND REPORT TABLE

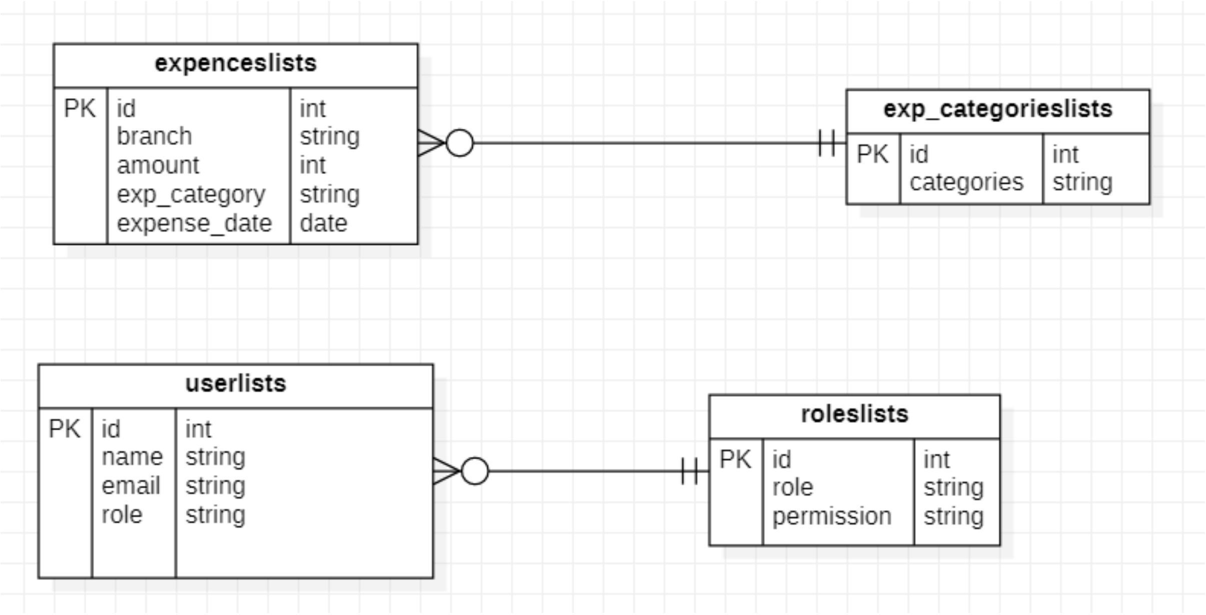


FIG 3.4.2 ER DIAGRAM FOR EXPENSES AND USER'S TABLES

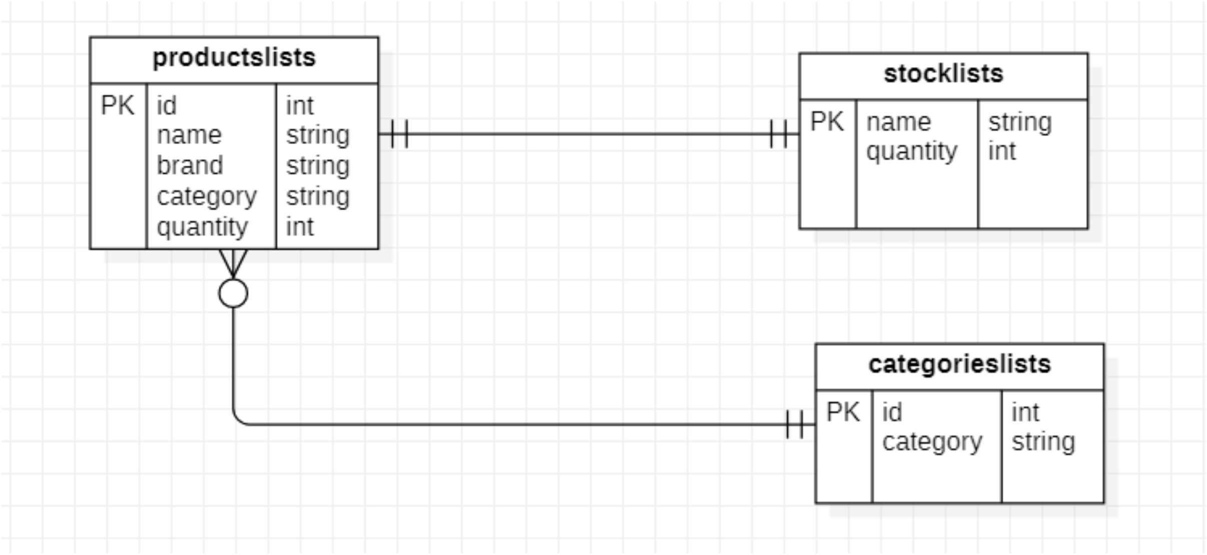


FIG 3.4.3 ER DIAGRAM FOR PRODUCTS TABLE

3.5 Organization benefits by using workssale tool:

- Get to know which products/services, geographies, and salespersons bring in the maximum revenue
- Learn when something goes wrong and avoid risks.
- Have better communications with your target audience by monitoring their behavior.
- It facilitates the organizations in achieving sales targets through easy monitoring.
- Resource management can also be improved by the organization by prioritizing the resources to the most demanding products first.
- It can be used by organizations of any scale.

Chapter-4

Project Implementation

Module Implementation:

- **Login Module:**
    1. The login module is implemented such that no user will be able to login into the tool without proper authentication.
    2. An employee is given an option to sign up and once an employee is successfully signed up he/she will receive a link to the specified mail id to activate their account in the tool.
    3. Employees should activate their account to use the tool from the link sent to their mail id.
    4. Employees cannot access web pages by specifying the URL in the address bar.
    5. All the web pages are secured such that no page is accessible without proper authentication.
    6. Once an employee is logged in he/she can view their profile in the profile tab section.
    7. An employee is permitted to change their password, first name, last name, and their email address.
    8. There are two levels of access to the tool, employee access, and super access.
    9. One with the super access role can sign into the admin module and can import or export the data into the database.

- **Email Module:**
    1. An email feature is integrated into the tool so that organizations can easily send emails to the customer easily and quickly.
    2. For the login module no need to create an activation link and send it to the newly registered customer, the tool will automatically create a link and will push the email to the customer.
    3. For this automated email to send, organizations should create a new support mail and should include the login details in the .env file.
    4. This email feature can also be used to send payment remainder notifications also.

- **Import and Export Module:**
    1. Organizations can upload the previous sales data into the tool easily by login into the tool as a superuser in the admin module.
    2. Employees can also download the data in the database in different formats if needed.
    3. A download feature is also built into the system so that employees can download a single report or wholesales data into a pdf format.

- **Charts Module:**
    1. There are line charts in the dashboard that shows information like sales and expenses of the organization.
    2. The stock chart shows the details of products and their stock quantity in a pie chart style.
    3. The information displayed in the charts is taken from the sales, expenses table in the database.
    4. Charts type can also easily be changed by the organization.

- **To-do and Calendar module:**
    1. A to-do list tab is provided to the organization so that their employees can easily track the future task to be completed easily.
    2. Employees can easily add, delete, and can mark as completed.
    3. The calendar tab shows the future events of the organization.
    4. For now, the calendar shows the account that sends the activation mails.
    5. Once the organizations give their employees workspace accounts, they can be integrated so that they show only their calendar in the dashboard section.

- **Payment Module:**
    1. On the sales page while inserting the sale data into the system there is a column included to display the payment status.
    2. If the payment is successful, it will be shown in green colour.
    3. If the payment is unsuccessful a pending red-coloured button is shown.
    4. The employee can click on the button to send a payment reminder to the customer

- **Profile Module:**
  1. There are two tabs in this module.
  2. The first one shows the details of the employee.
  3. In the second tab, the employee can change his/her details.
  4. Employees cannot change others' emails, passwords, or any other data.
  5. Only the employee with superuser access can change or delete an employee login access to the module.

- **New User Module:**
  1. A new user can be created in two ways one is through the signup process.
  2. Another is through the staff section in the tool. Admin has permission to add staff to the module and can specify access to the module.
  3. If access to the module to be provided to the staff, the admin can create a username and password and can send an authentication mail to the staff.

- **Report Module:**
  1. There is no option provided to insert data into the report module, it takes the data from the sales module and shows an analysis report of the sales.
  2. In reports tabs, the tool shows the analysis report of the products, tax, and customers.
  3. All the details are filtered and shown to the employee in the reports module.

- **Tax Module:**
  1. Different taxes are imposed on different products.
  2. So the tool is implemented with a built-in tax page so that organizations can keep track of all the tax percentages that are imposed on different products.
  3. There is an option to set the tax as default. So that the tax will be automatically imposed.

- **Sales Module:**
    1. All the sale payment details are updated here so that they can be further used in another module.
    2. In this module, a download feature is implemented so that employees can download the receipt of the payment.
    3. Single payment receipts can be downloaded to submit to the customer or all the sales reports can be downloaded for analysis purposes.
    4. It can also be used as a receipt and can be given to the customer.
    5. The report will be downloaded in pdf format.
    6. The information in the report can be changed according to the organization instructions, can also add additional information if they want to.
    7. Email remainder option is also provided to the employee. So that he/she no need to send the payment reminder to every customer.
    8. If payment is pending, the employee can click on the pending button to send a reminder mail to the customer.

Chapter-5

Conclusion and Scope of Future Work

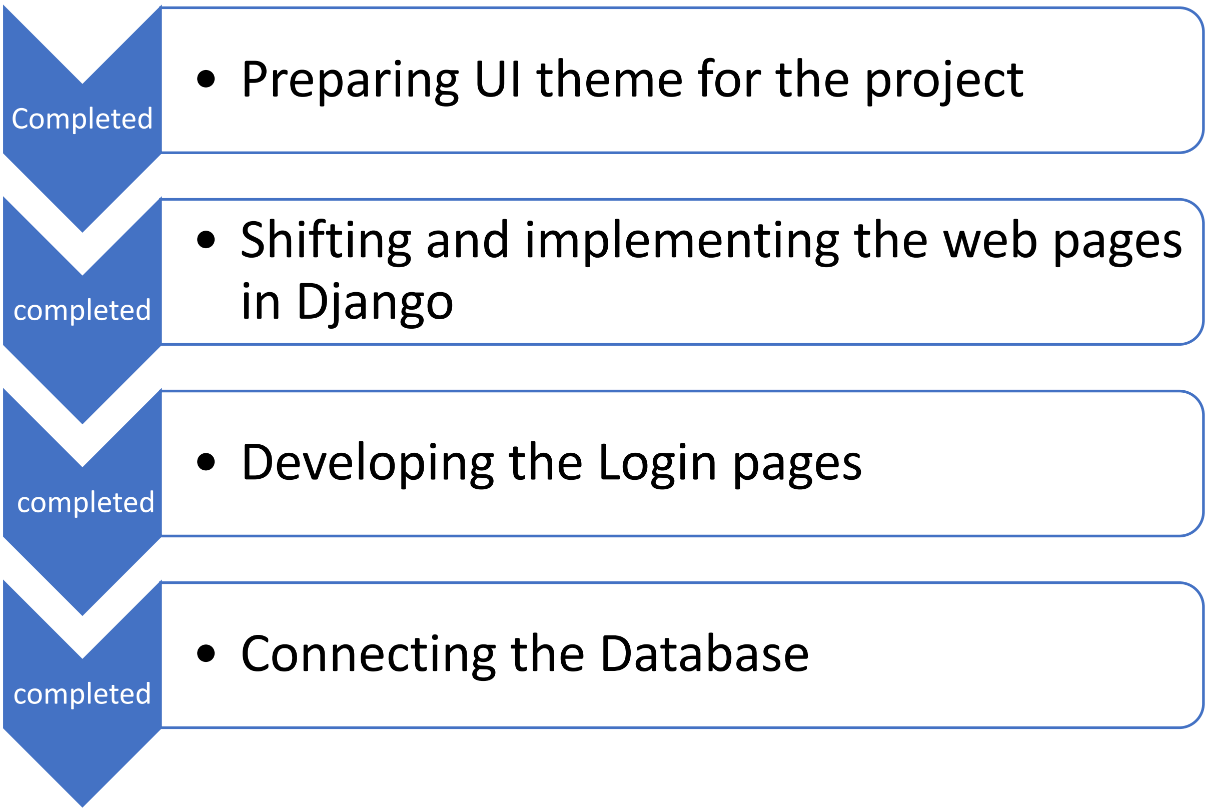
Conclusion and Scope of Future work:

- Rather than using the traditional system to monitor tools which consume a lot of time to gather all the necessary information to monitor a particular product workssale tool provides a detailed view of the sales and expenses of a company in one place.
- Not only monitoring sales but through the contact details of the customers and vendors organizations can also push new offers or product details.
- For medium-sized businesses who cannot include sales staff to monitor each product they are launching workssale is a great option to take advantage of.

APPENDIX - A

Project Status and Source Code

Internship Project Steps:



Completed
- Preparing UI theme for the project

completed
- Shifting and implementing the web pages in Django

completed
- Developing the Login pages

completed
- Connecting the Database

SAMPLE SOURCE CODE:

Manage.py

```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys


def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'main_part.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)


if __name__ == '__main__':
    main()
```

urls.py

```
from django.urls import path
from . import views

urlpatterns=[
    path('payment_pending/<int:id>',views.payment_pending,name='payment_pending'),
path('pdf/<pk>/',views.sales_render_pdf_view,name="sales_render_pdf_view"),
path('sales_list_pdf',views.sales_list_pdf,name="sales_list_pdf"),

path('',views.landing_page,name="landing_page"),
path('home',views.home,name='home'),
path('signup', views.signup, name="signup"),
path('login', views.login, name="login"),
path('logout', views.logout, name="logout"),
path('activate/<uidb64>/<token>',views.VerificationView.as_view(), name="activate"),

path('index',views.index,name="index"),
path('todo_add',views.todo_add,name='todo_add'),
path('delete_todo/<int:id>', views.delete_todo),
path('categories.html',views.categories,name="categories"),
path('customer_analysis.html',views.customer_analysis,name="customer_analysis"),
path('customer.html',views.customer,name="customer"),
path('exp_categories.html',views.exp_categories,name="exp_categories"),
path('expences.html',views.expences,name="expences"),
path('products.html',views.products,name="products"),
path('profile.html',views.profile,name="profile"),
path('roles.html',views.roles,name="roles"),
path('sales_list.html',views.sales_list,name="sales_list"),
path('sales.html',views.sales,name="sales"),
path('stock_analysis.html',views.stock_analysis,name="stock_analysis"),
path('tax_report.html',views.tax_report,name="tax_report"),
path('tax.html',views.tax,name="tax"),
path('users.html',views.users,name="users"),
path('vendors.html',views.vendors,name="vendors"),

path('delete_users/<int:id>', views.delete_users),
path('delete_roles/<int:id>', views.delete_roles),
path('delete_customer/<int:id>', views.delete_customer),
path('delete_vendors/<int:id>', views.delete_vendors),
path('delete_products/<int:id>', views.delete_products),
    path('delete_categories/<int:id>', views.delete_categories),
    path('delete_tax/<int:id>', views.delete_tax),
    path('delete_sales_list/<int:id>', views.delete_sales_list),
    path('delete_expences/<int:id>', views.delete_expences),
    path('delete_exp_categories/<int:id>', views.delete_exp_categories),

    path('sales_list_view/<int:id>',views.sales_list_view,name='sales_list_view'),

    path('users_edit/<int:id>',views.users_edit,name='users_edit'),
    path('roles_edit/<int:id>',views.roles_edit,name='roles_edit'),
    path('customer_edit/<int:id>',views.customer_edit,name='customer_edit'),
    path('vendors_edit/<int:id>',views.vendors_edit,name='vendors_edit'),
    path('products_edit/<int:id>',views.products_edit,name='products_edit'),
    path('categories_edit/<int:id>',views.categories_edit,name='categories_edit'),
    path('tax_edit/<int:id>',views.tax_edit,name='tax_edit'),
    path('sales_list_edit/<int:id>',views.sales_list_edit,name='sales_list_edit'),
    path('expences_edit/<int:id>',views.expences_edit,name='expences_edit'),

path('exp_categories_edit/<int:id>',views.exp_categories_edit,name='exp_categories_edit'),

    path('users_update/<int:id>',views.users_update,name='users_update'),
    path('roles_update/<int:id>',views.roles_update,name='roles_update'),
    path('customer_update/<int:id>',views.customer_update,name='customer_update'),
    path('vendors_update/<int:id>',views.vendors_update,name='vendors_update'),
    path('products_update/<int:id>',views.products_update,name='products_update'),
    path('categories_update/<int:id>',views.categories_update,name='categories_update'),
    path('tax_update/<int:id>',views.tax_update,name='tax_update'),
    path('sales_list_update/<int:id>',views.sales_list_update,name='sales_list_update'),
    path('expences_update/<int:id>',views.expences_update,name='expences_update'),

path('exp_categories_update/<int:id>',views.exp_categories_update,name='exp_categories_u
pdate'),

]
```

Models.py

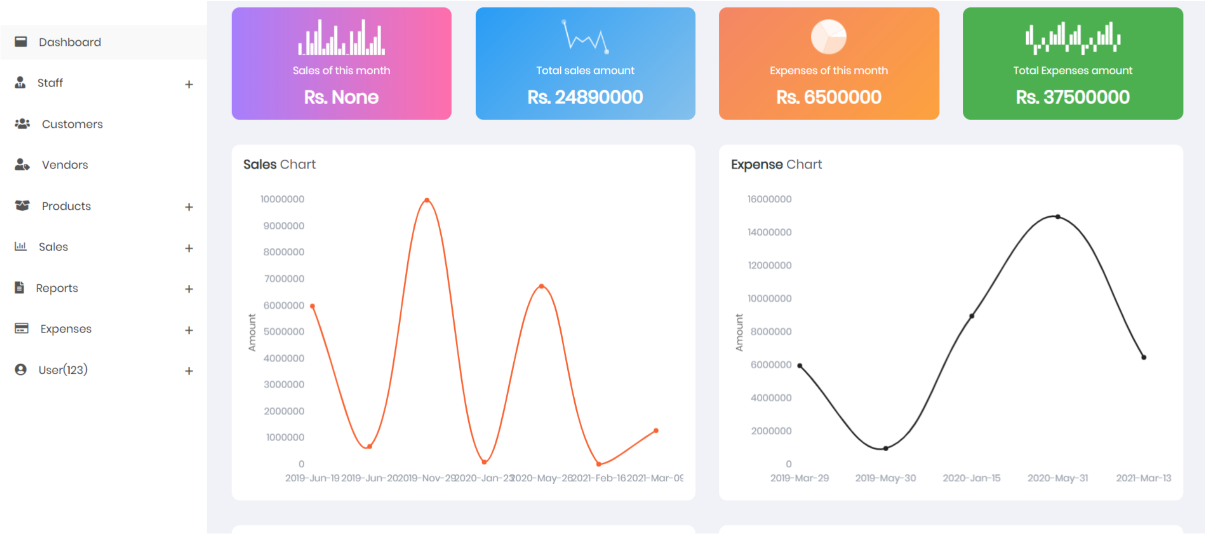
```
from django.db import models
from multiselectfield import MultiSelectField
# Create your models here.

class todolists(models.Model):
    sentense=models.CharField(max_length=100)
    def __str__(self):
        return self.sentense

class userslists(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField()
    user_role = models.CharField(max_length=100)
    def __str__(self):
        return '{} - {} - {}'.format(self.name,self.email,self.user_role)

class roleslists(models.Model):
    role = models.CharField(max_length=100)
    Profile_box= models.CharField(max_length=200)
    Vendor_box= models.CharField(max_length=200)
    Customer_box= models.CharField(max_length=200)
    user_box= models.CharField(max_length=200)
    roles_list = (
        ('edit hello','edit hello'),
        ('edit bolo','edit bolo'),
    )
    title=MultiSelectField(choices= roles_list)
    def __str__(self):
        return '{} - {} - {} - {} -
{}'.format(self.role,self.Profile_box,self.Vendor_box,self.user_box,self.Customer_box)

class customerlists(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField()
    # phone number is used on cusomer analysis page
    phone_number = models.IntegerField()
    date = models.DateTimeField()
    # these fields should not be given by user this will be handled in views assigned zero they will be used in customer analysis page
    total_sales = models.IntegerField()
    total = models.IntegerField()
    def __str__(self):
        return '{} - {} - {} - {}'.format(self.name, self.email, self.phone_number, self.date)

class vendorlists(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField()
    # phone number is used on cusomer analysis page
    phone_number = models.IntegerField()
    date = models.DateTimeField()
    # these fields should not be given by user this will be handled in views assigned zero they will be used in customer analysis page
    total_sales = models.IntegerField()
    total = models.IntegerField()
    def __str__(self):
        return '{} - {} - {} - {}'.format(self.name, self.email, self.phone_number, self.date)

class productslists(models.Model):
    name = models.CharField(max_length=100)
    brand = models.CharField(max_length=100)
    category = models.CharField(max_length=100)
    quantity = models.IntegerField()
    # these fields should not be given by user they are assgned to default in view.productslist
    stock = models.IntegerField()
    def __str__(self):
        return '{} - {} - {} - {}'.format(self.name,self.brand,self.category,self.quantity)
```

```
class categorieslists(models.Model):
    name = models.CharField(max_length=100)
    def __str__(self):
        return self.name



class taxlists(models.Model):
    tax_percentage = models.IntegerField()
    is_default = models.BooleanField(default=False)
    def __str__(self):
        return '{} - {}'.format(self.tax_percentage, self.is_default)

class saleslists(models.Model):
    invoice_id = models.CharField(max_length=100)
    date = models.DateTimeField()
    email = models.EmailField()
    sold = (
        ('Walk in Customer','Walk in Customer'),
        ('Vendor','vendor'),
    )
    sold_to = models.CharField(max_length = 100, choices = sold)
    items_sold = models.IntegerField()
    total = models.IntegerField()
    payment = (
        ('Success','Success'),
        ('Pending','Pending'),
    )
    payment_status = models.CharField(max_length = 100, choices = payment)
    # name(product name) is used in stock_analysis page
    # it is used in views.stock_analysis to compare product names
    name = models.CharField(max_length=100)
    # customer_name is used in views.customer_analysis page to compare customer names
    customer_name = models.CharField(max_length=100)
    # these three are used for tax analysis calculation
    tax=models.IntegerField()
    # these fields should not be given by user they are assgned to default in view.tax_report
    tax_amount=models.IntegerField()
    grand_total=models.IntegerField()
    def __str__(self):
        return '{} - {} - {} - {} - {} - {} - {} - {} - {}'.format(self.invoice_id, self.date,
self.sold_to, self.items_sold, self.total, self.payment_status, self.name, self.customer_name,
self.tax)

class stock_analysislists(models.Model):
    name = models.CharField(max_length=100)
    quantity = models.IntegerField()
    def __str__(self):
        return self.name

class customer_analysislists(models.Model):
    name = models.CharField(max_length=100)
    phone_number = models.IntegerField()
    email = models.EmailField()
    total_sales = models.IntegerField()
    total = models.IntegerField()
    def __str__(self):
        return self.name

class tax_reportslists(models.Model):
    reference_no = models.CharField(max_length=100)
    date = models.DateTimeField()
    sold = (
        ('Walk in Customer','Walk in Customer'),
        ('Vendor','vendor'),
    )
    vendor = models.CharField(max_length = 100, choices = sold)
    product_tax = models.IntegerField()
    grand_total = models.IntegerField()
    def __str__(self):
        return self.reference_no

class expenseslists(models.Model):
    branch = models.CharField(max_length=100)
    amount = models.IntegerField()
    expense_category = models.CharField(max_length=100)
    expense_date = models.DateTimeField()
    def __str__(self):
        return '{} - {} - {} - {}'.format(self.branch, self.amount, self.expense_category,
self.expense_date)

class exp_categorieslists(models.Model):
    name = models.CharField(max_length=100)
    def __str__(self):
        return self.name
```

APPENDIX - B

Screenshots

Admin Dashboard:

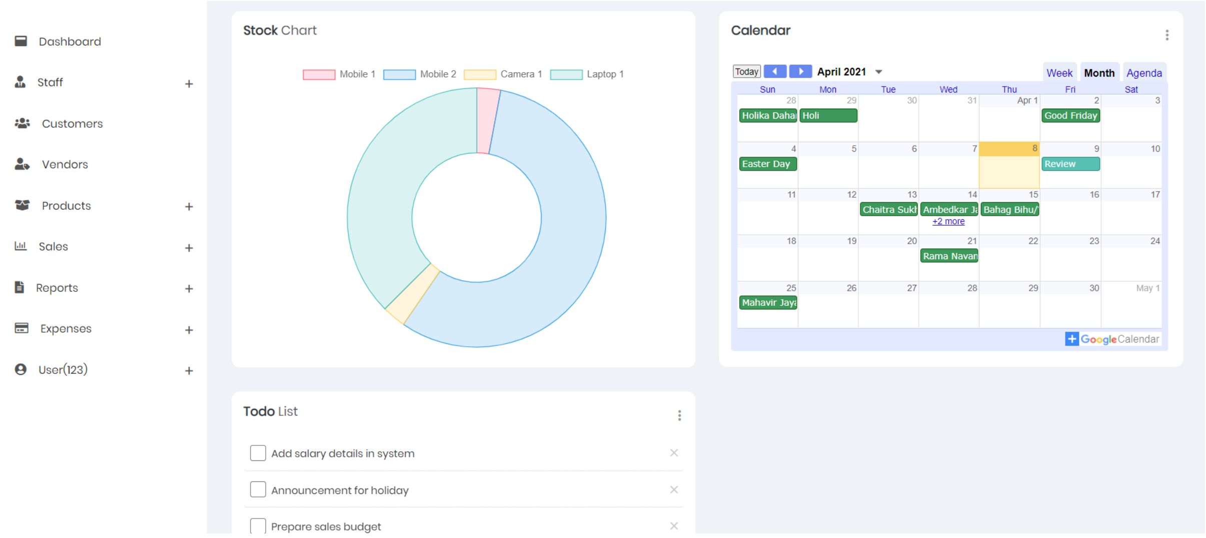


Sales of this month
Rs. None

Total sales amount
Rs. 24890000

Expenses of this month
Rs. 6500000

Total Expenses amount
Rs. 37500000



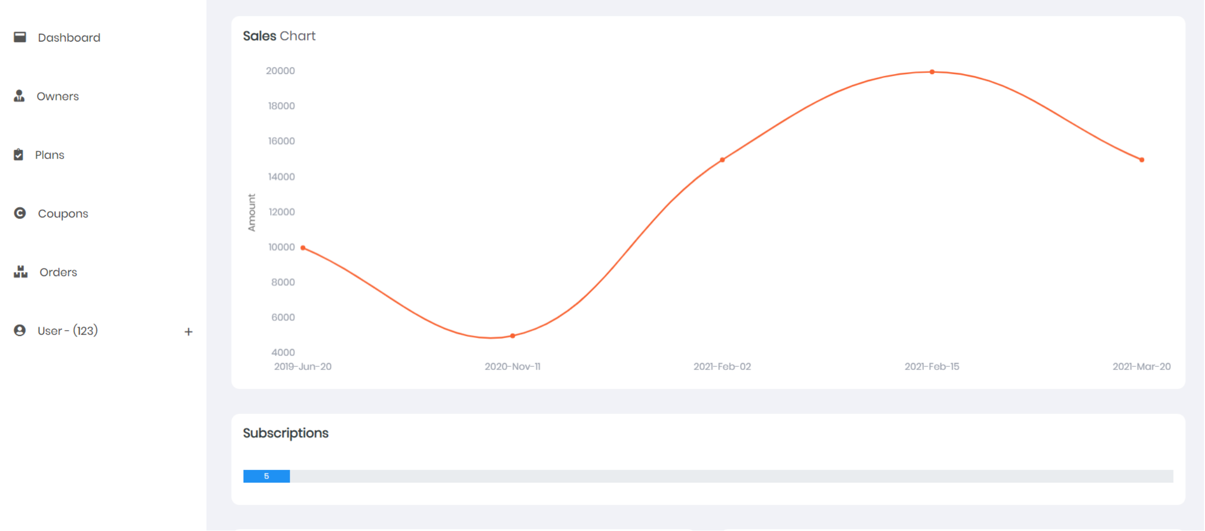
□ Add salary details in system ×

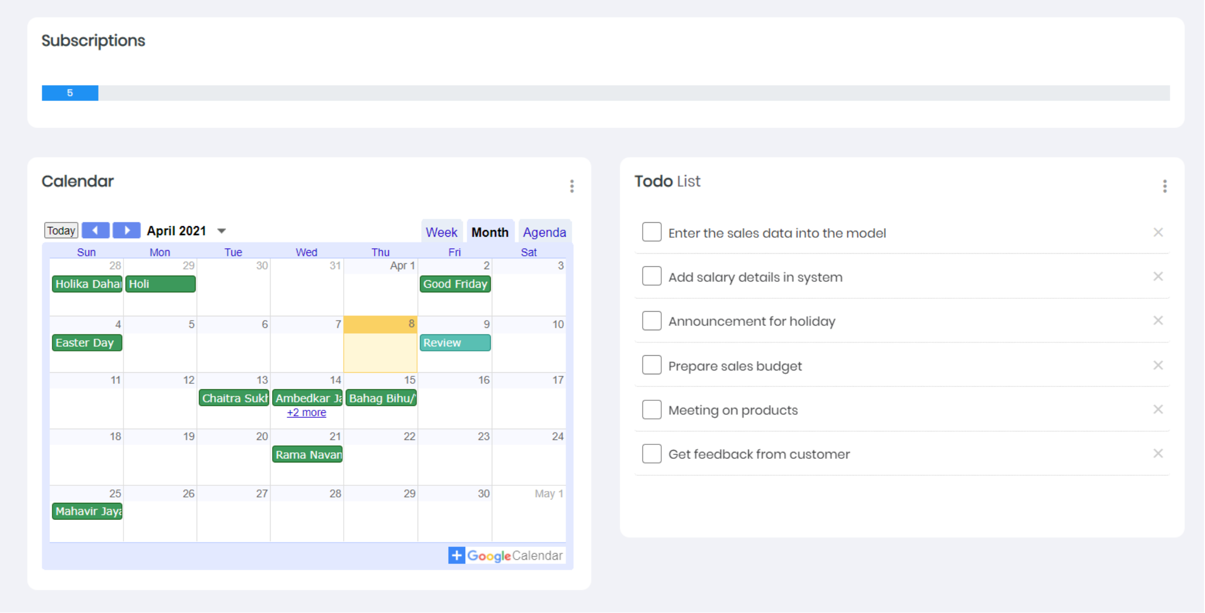
□ Announcement for holiday ×

□ Prepare sales budget ×

Fig 7.1 and 7.2 Dashboard

Super Admin Dashboard:





Sun	Mon	Tue	Wed	Thu	Fri	Sat
28 Holika Dahai	29 Holi	30	31	Apr 1	2 Good Friday	3
4 Easter Day	5	6	7	8	9 Review	10
11	12	13 Chaitra Sukl	14 Ambedkar Ja +2 more	15 Bahag Bihu/	16	17
18	19	20	21 Rama Navan	22	23	24
25 Mahavir Jaya	26	27	28	29	30	May 1

□ Enter the sales data into the model ×

□ Add salary details in system ×

□ Announcement for holiday ×

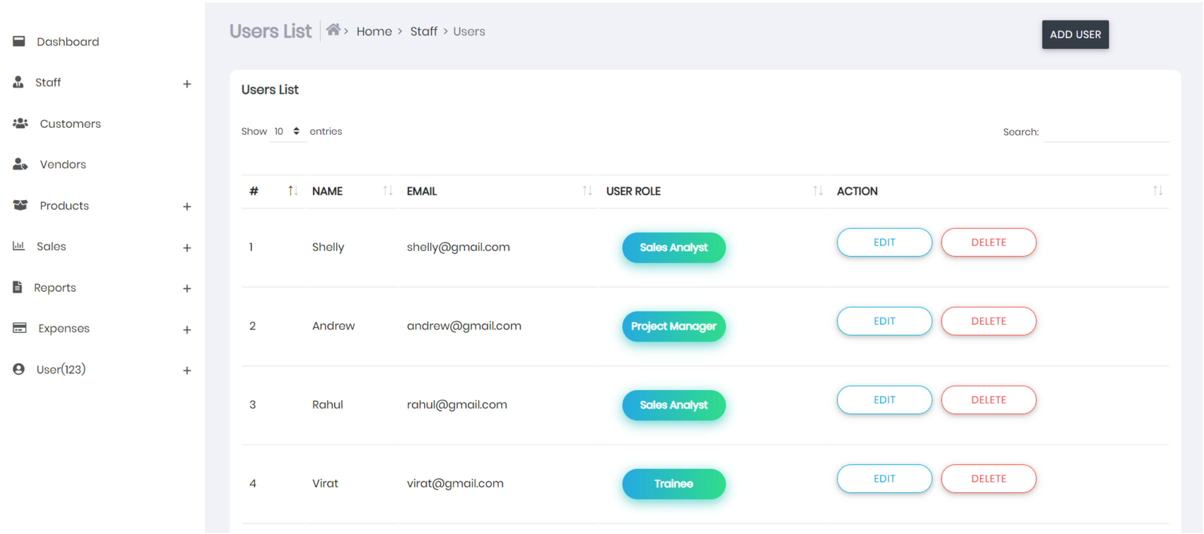
□ Prepare sales budget ×

□ Meeting on products ×

□ Get feedback from customer ×

Fig 7.3 & 7.4 Super Admin Dashboard

Staff Page:



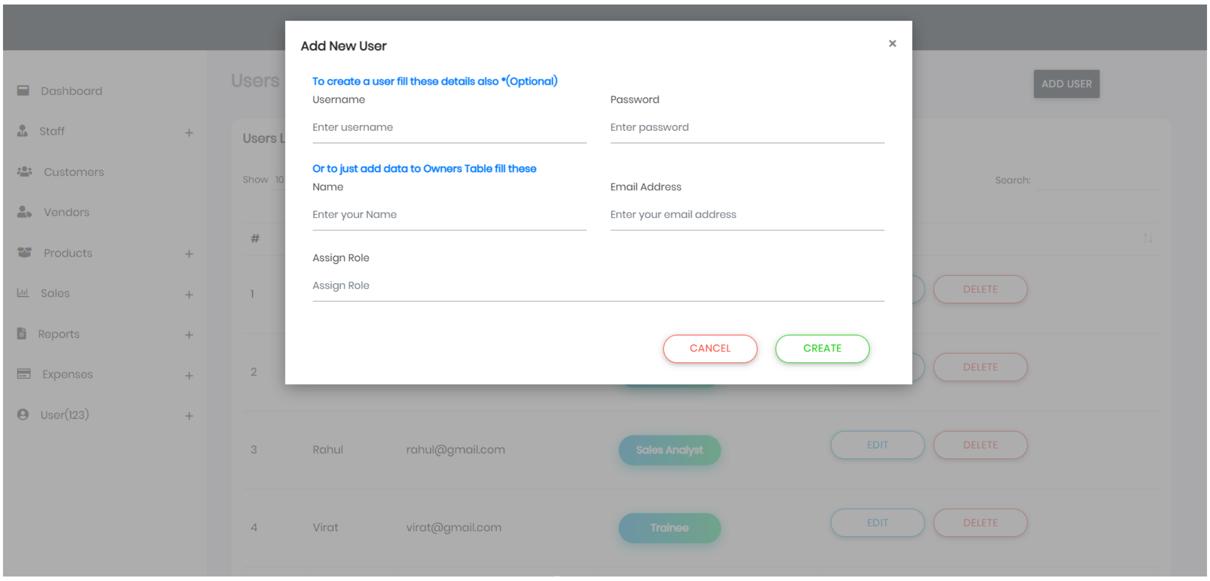
Users List

Show 10 entries Search:

#	NAME	EMAIL	USER ROLE	ACTION
1	Shelly	shelly@gmail.com	Sales Analyst	EDIT DELETE
2	Andrew	andrew@gmail.com	Project Manager	EDIT DELETE
3	Rahul	rahul@gmail.com	Sales Analyst	EDIT DELETE
4	Virat	virat@gmail.com	Trainee	EDIT DELETE

Fig 7.5 Staff Page

Add New Staff:



Add New User

To create a user fill these details also *(Optional)

Username: Enter username

Password: Enter password

Or to just add data to Owners Table fill these

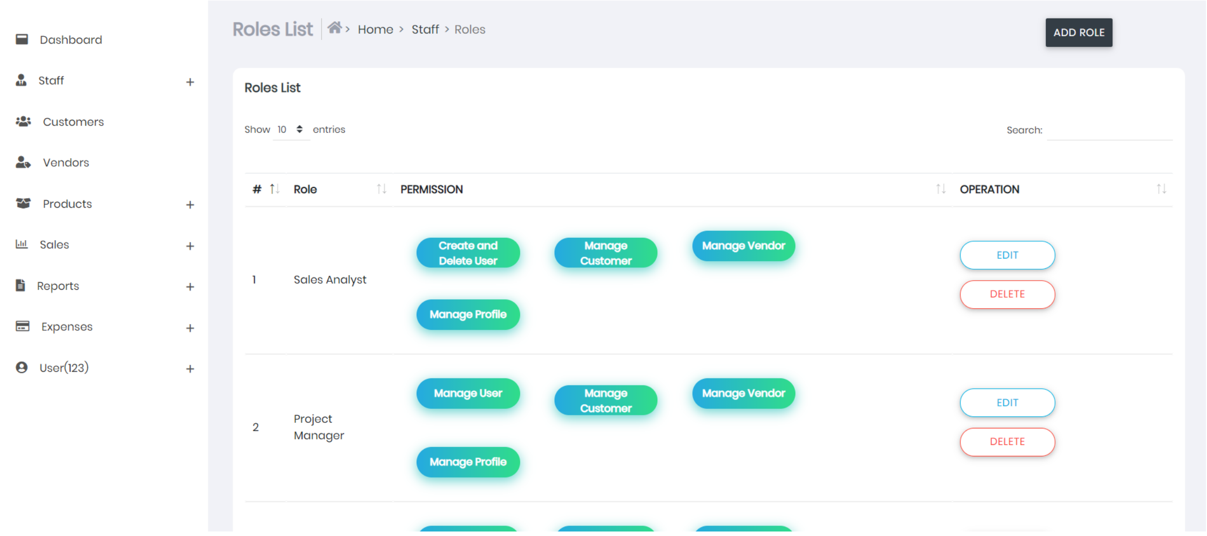
Name: Enter your Name

Email Address: Enter your email address

Assign Role: Assign Role

Fig 7.6 Add New Staff

Staff Roles and Permission List:



Roles List

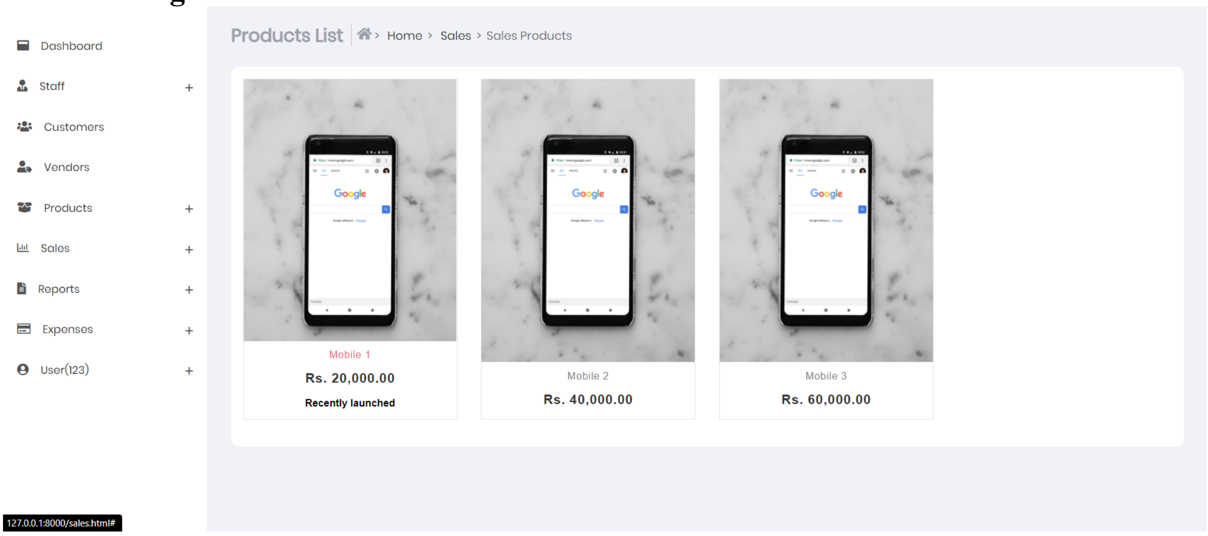
Show 10 entries

Search:

#	Role	PERMISSION	OPERATION
1	Sales Analyst	Create and Delete User Manage Customer Manage Vendor Manage Profile	EDIT DELETE
2	Project Manager	Manage User Manage Customer Manage Vendor Manage Profile	EDIT DELETE

Fig 7.7 Staff Roles and Permission List

Products Page:



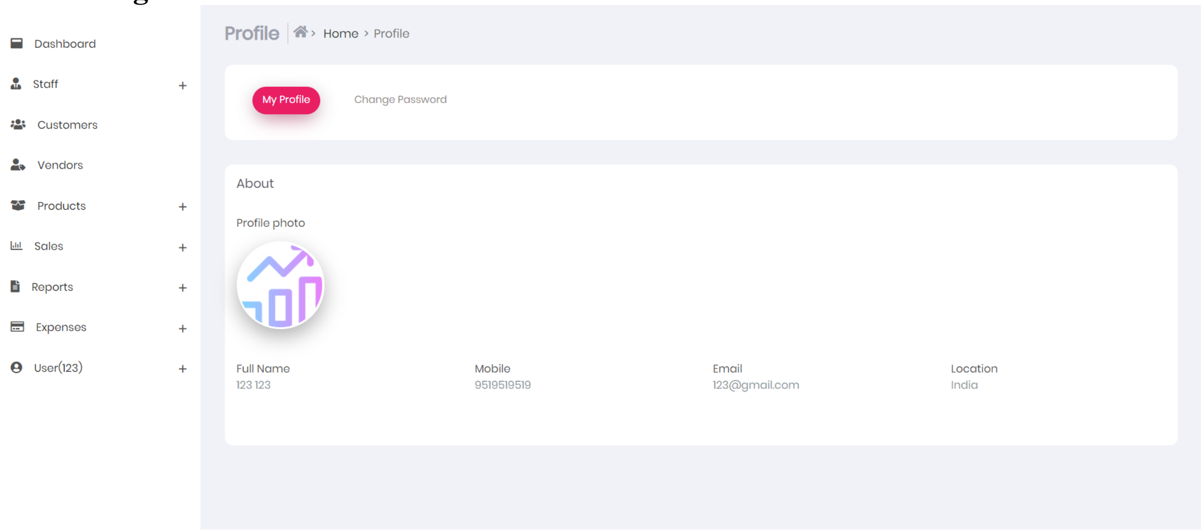
Mobile 1
Rs. 20,000.00
Recently launched

Mobile 2
Rs. 40,000.00

Mobile 3
Rs. 60,000.00

Fig 7.8 Products Page

Profile Page:



Full Name
123 123

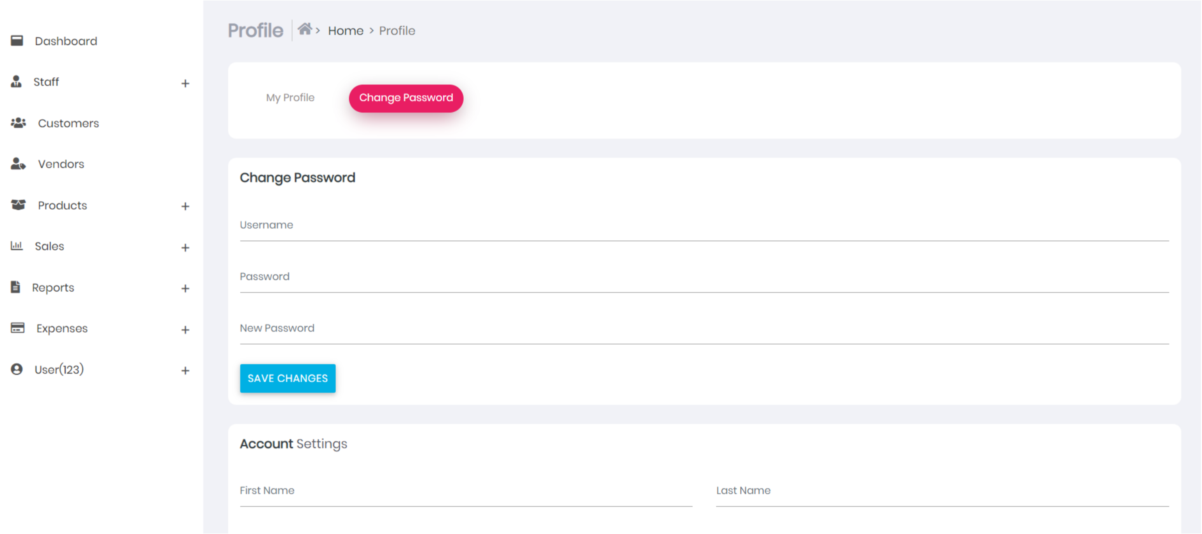
Mobile
9519519519

Email
123@gmail.com

Location
India

Fig 7.9 Profile Page

Profile page:



Change Password

Username

Password

New Password

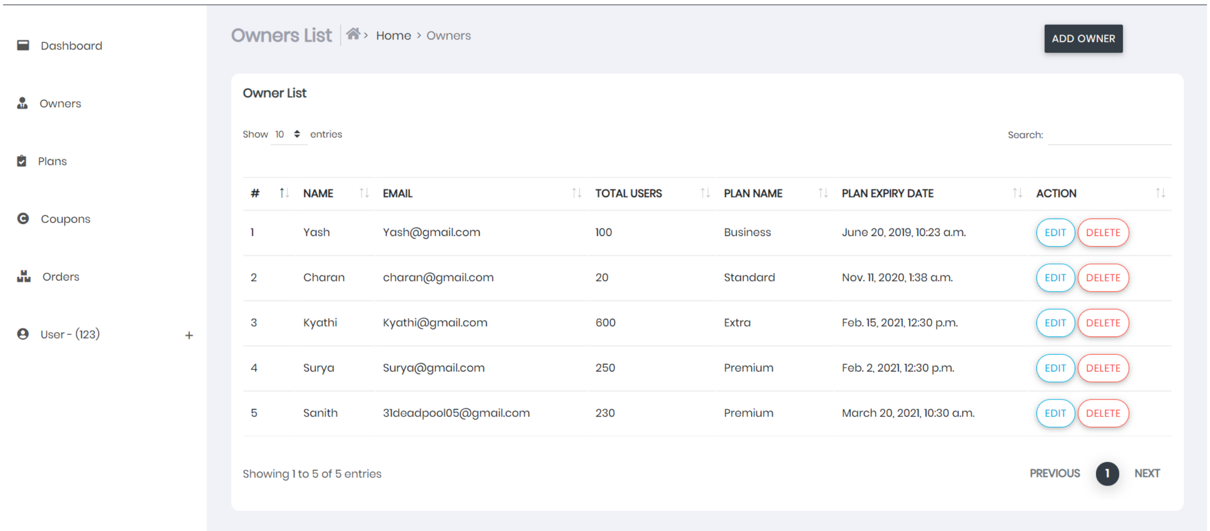
Account Settings

First Name

Last Name

Fig 7.10 Profile edit page

Super Admin Owners Page:



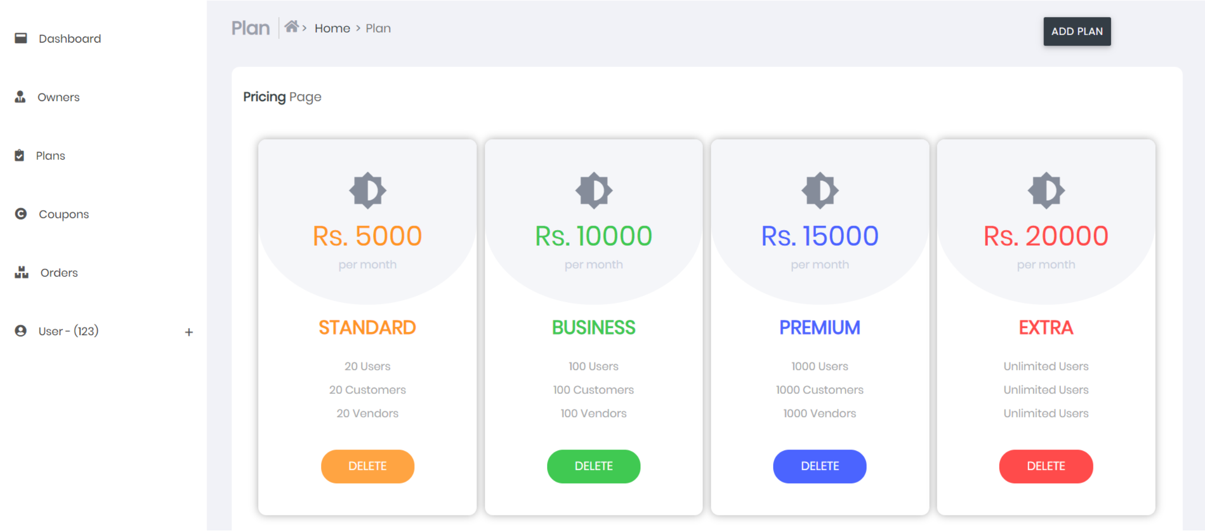
Owner List

#	NAME	EMAIL	TOTAL USERS	PLAN NAME	PLAN EXPIRY DATE	ACTION
1	Yash	Yash@gmail.com	100	Business	June 20, 2019, 10:23 a.m.	EDIT DELETE
2	Charan	charan@gmail.com	20	Standard	Nov. 11, 2020, 1:38 a.m.	EDIT DELETE
3	Kyathi	Kyathi@gmail.com	600	Extra	Feb. 15, 2021, 12:30 p.m.	EDIT DELETE
4	Surya	Surya@gmail.com	250	Premium	Feb. 2, 2021, 12:30 p.m.	EDIT DELETE
5	Sanith	31deadpool05@gmail.com	230	Premium	March 20, 2021, 10:30 a.m.	EDIT DELETE

Showing 1 to 5 of 5 entries

Fig 7.11 Super Admin Owners Page

Plan Subscriptions Pages:



Pricing Page

Rs. 5000
per month
STANDARD
20 Users
20 Customers
20 Vendors

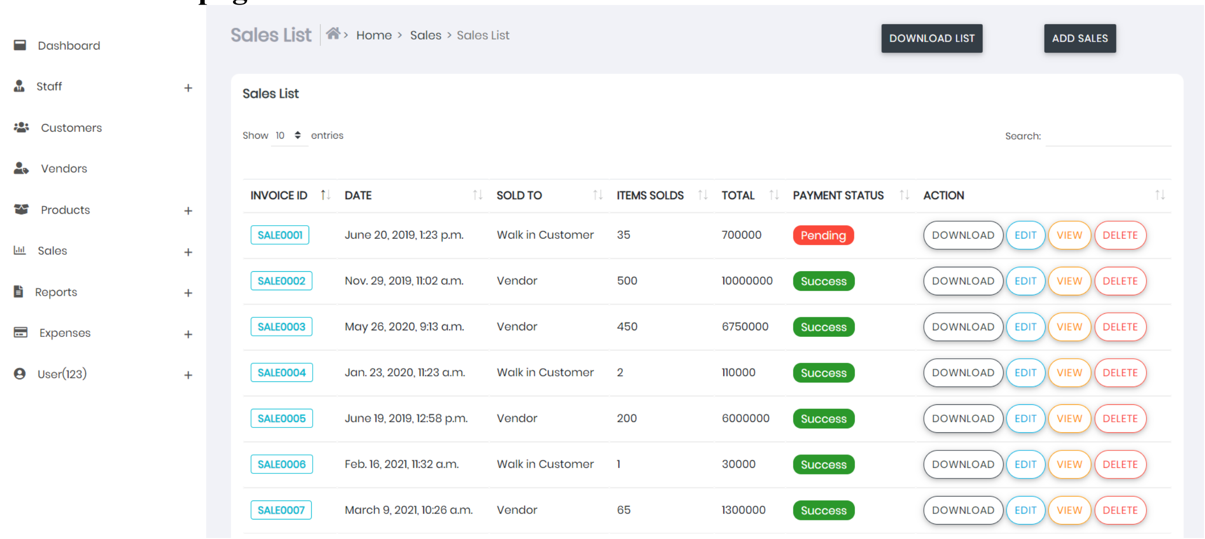
Rs. 10000
per month
BUSINESS
100 Users
100 Customers
100 Vendors

Rs. 15000
per month
PREMIUM
1000 Users
1000 Customers
1000 Vendors

Rs. 20000
per month
EXTRA
Unlimited Users
Unlimited Users
Unlimited Users

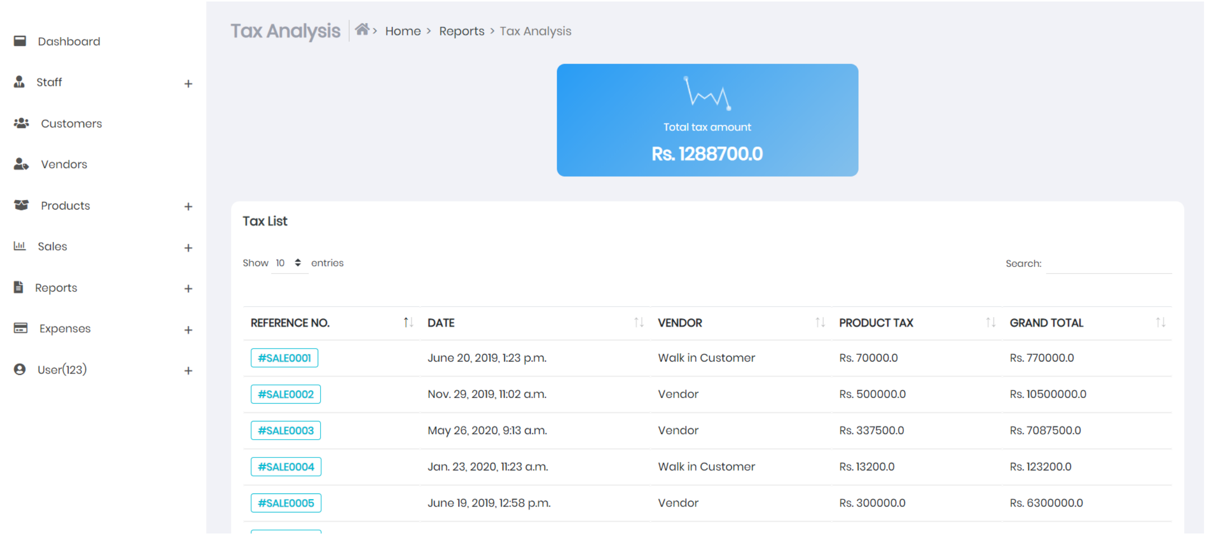
Fig 7.12 Plan Subscriptions Page

Sales and Tax page:



INVOICE ID	DATE	SOLD TO	ITEMS SOLDS	TOTAL	PAYMENT STATUS	ACTION
SALE0001	June 20, 2019, 1:23 p.m.	Walk in Customer	35	700000	Pending	DOWNLOAD EDIT VIEW DELETE
SALE0002	Nov. 29, 2019, 11:02 a.m.	Vendor	500	10000000	Success	DOWNLOAD EDIT VIEW DELETE
SALE0003	May 26, 2020, 9:13 a.m.	Vendor	450	6750000	Success	DOWNLOAD EDIT VIEW DELETE
SALE0004	Jan. 23, 2020, 11:23 a.m.	Walk in Customer	2	110000	Success	DOWNLOAD EDIT VIEW DELETE
SALE0005	June 19, 2019, 12:58 p.m.	Vendor	200	6000000	Success	DOWNLOAD EDIT VIEW DELETE
SALE0006	Feb. 16, 2021, 11:32 a.m.	Walk in Customer	1	30000	Success	DOWNLOAD EDIT VIEW DELETE
SALE0007	March 9, 2021, 10:26 a.m.	Vendor	65	1300000	Success	DOWNLOAD EDIT VIEW DELETE

Fig 7.13 Sales Page



REFERENCE NO.	DATE	VENDOR	PRODUCT TAX	GRAND TOTAL
#SALE0001	June 20, 2019, 1:23 p.m.	Walk in Customer	Rs. 70000.0	Rs. 770000.0
#SALE0002	Nov. 29, 2019, 11:02 a.m.	Vendor	Rs. 500000.0	Rs. 10500000.0
#SALE0003	May 26, 2020, 9:13 a.m.	Vendor	Rs. 337500.0	Rs. 7087500.0
#SALE0004	Jan. 23, 2020, 11:23 a.m.	Walk in Customer	Rs. 13200.0	Rs. 123200.0
#SALE0005	June 19, 2019, 12:58 p.m.	Vendor	Rs. 300000.0	Rs. 6300000.0

Fig 7.14 Tax Page

References

1. https://www.w3schools.com/html/
2. https://docs.djangoproject.com/en/3.1/topics/auth/default/
3. https://docs.djangoproject.com/en/3.1/ref/databases/
4. https://pypi.org/project/django-multiselectfield/
5. https://pypi.org/project/xhtml2pdf/
6. https://www.chartjs.org/docs/latest/samples/other-charts/doughnut.html