SUPER MARKET AUTOMATION SOFTWARE

405 FOUND

Yasaswani Rongali - 21CS10082
Shamitha Ambati - 21CS30004
Gaddam Sree Harshitha - 21CS10024

PROBLEM STATEMENT

- Super Market Automation Software for digitising supermarket billing, accounts and purchases.
- The manual process is time consuming, cost ineffective and prone to human error.
- The aim is to develop software to help supermarkets and the relevant users.
- SAS reduces the dependency of supermarkets on large teams to manage supermarket processes.

PROBLEM STATEMENT

- Manager is at the top of the employee hierarchy.
- The Manager can view the inventory,change the price of an item and can even access the statistics of a product in a given period of time.
- The salesclerk performs new transaction and prints the bill containing the items,quantity sold and total price payable.
- An employee(IM staff) can update the inventory when a new supply arrives.

Manager

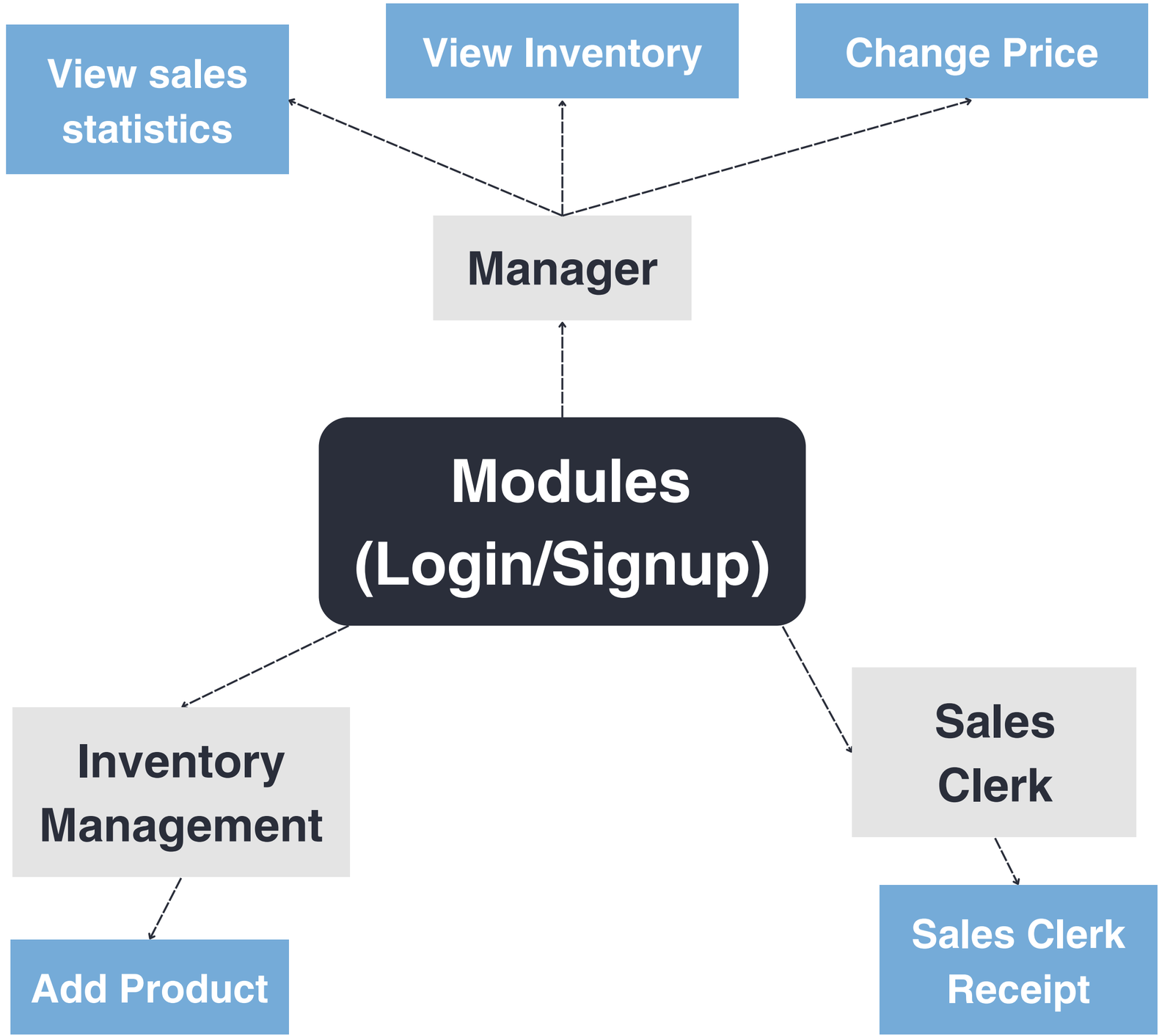
Inventory staff Management

Sales Clerk

SOLUTION

We have developed a Web-based application for resolving these issues using the modern full stack development by designing and implementing the frontend and the backend with easy to use UI and robust backend database.

Salient Features of SRS and Software Design



Software Architecture

Software Architecture

- Programming Language: Python.
- Backend software architecture: Flask Framework.
- Frontend software architecture: HTML, CSS.
- Database: SQLalchemy,SQLite.

Software Architecture

- Python, Flask, SQLAlchemy, and SQLite are all open-source technologies, which means they are freely available and widely used. This makes it easier to find documentation, support, and examples online.
- Flask has a reputation for being easy to learn and use, which made it easier for us to build and maintain the SAS software.

Challenges Faced

- Integration of the database with software was one of the major challenges we faced during the initial stages of development of the SAS
- Creating the New transaction and View sales statistics functionalities was one of the difficulties that came in our way because of the interlinking of the item and item transaction history classes which we were unaware of in the beginning

Scope of Improvement

Improvements

- Frontend can be made more responsive.
- Add layers of data verification using OTP verification.
- Displaying of flash messages in the same page when invalid data entered in any of the functionalities
- Keep a track of the number of transactions and adding the id of a transaction as an attribute of item transaction history

THANK YOU