SkyNet: A BUSINESS MANAGEMENT APPLICATION FOR THE SMALL BUSINESS

Josh Krinsky
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
jkrinsky91@gmail.com

Kai Wang
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
kw87502p@pace.edu

Peter Torrente
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
pvtorrente@gmail.com

Peter McKechnie
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
mckechniep@gmail.com

Jincheng Zou
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
jzcsit@gmail.com

Houqi Zhan
Seidenberg School of Computer Science and information Systems
Pace University
New York, United States
z326316507@gmail.com

***Abstract*—This document outlines the design, implementation, and functionality of Skynet, a business management application. The application is aimed small businesses with between 15 and 500 employees. Skynet integrates vendor, customer, warehouse, and inventory management, logistics, and cash register functions, and uses a machine learning model to recognize trends and recommend sales strategy. It is a simple, affordable, cloud-based option to allow the user to access their data from anywhere on any device.**

Keywords—Business management application, small businesses, cloud-based, vendor management, customer management, warehouse management, inventory management, logistics, cash register, machine learning, trends recognition, predictive analytics, React.js, Django, MySQL, ER model, UML diagram.

I. INTRODUCTION

With most commerce shifting away from physical locations and moving online, management software has become necessary to businesses. This application is a cloud hosted business management application. The authors designed this software using python, MySQL, JavaScript, and HTML to be offer a dependable platform to manage inventory, organize contacts, track logistics, make purchases, and analyze sales trends.

II. LITERATURE SURVEY

There are several major client and business management applications on the market, but they may be overwhelming for non-technical users. Skynet aims to offer basic necessary features that are customizable to the user's needs in a lean and affordable package. The app will centralize the logistics of inventory management, vendor organization, and customer analytics while offering some of the more useful features like pricing and sale suggestions.

PROJECT REQUIREMENTS

III. PROJECT REQUIREMENTS

This section outlines the hardware and software requirements, functional requirements, technical requirements, and usability requirements for Skynet. The application is built using Django, a web application framework, and React.js, a JavaScript library for building one-page web interfaces. The database is hosted on Amazon Web Services servers in a MySQL layout.

A. Requirements

- Hardware requirements: internet connected device.
- Software requirements: internet browser
- Functional Requirements:
  - The admin will be able to add/edit/delete tables the database
  - The user will be alerted when inventory is low, incoming orders have impending delivery, and when a shipment is delivered.
  - The user will be able to pull reports, upload and edit data through the interface
- Technical requirements
  - The application will be built using Django.
  - Django is a Python framework for building Restful web applications
  - The database will be hosted in Amazon Web Services servers in a MySQL layout. This was changed to locally hosted database as the data limit for the free tier of AWS was exceeded.
  - While developing the app we are using to use an agile workflow plan
  - The interface will be built in React.js
  - React.js is a JavaScript and JSX library for building One-Page web interfaces

B. *Usability*

- The application will be available on mobile devices
- The application will work with any browser
- The app will have simple UI so anyone can use it

IV. ARCHITECTURE

Skynet's implementation includes a front-end, back-end, and database. The front-end, managed with React.js and Bootstrap, provides a graphical user interface. The back-end, managed with Django, handles the logic and interaction with the database. The database uses MySQL, a relational database structure that employs SQL to manage it. These are explained further below.

A. *Frontend*

Front end web development describes the portion of the app that end users directly interact with. It takes the raw data, in this case formatted in JSON, and converts it to a graphical user interface(GUI) through HTML, JavaScript, and CSS. We have used React to manage the design and implementation of the user/admin side using Bootstrap, an open-source design framework.

- React.JS is an open-source HTML ,CSS, and JavaScript framework for building single page applications

B. *Backend*

Back-end development describes the practice of working on the server-side software. It translated between the frontend requests and the all the working which the end user cannot see. We have used Django to manage the logic and interaction with the database.

1. Django is a free and open-source, Python-based web framework that follows the model–template–views (MTV) architectural pattern, which converts and executes python code as SQL queries.
2. Django allows programmers to use object-oriented program to design and query a database.
3. The model represents the database, the view represents the user interface, and the template defines the static portions of the desired HTML

C. *Database*

Our database uses MySQL, a relational database structure which employs SQL to manage it.

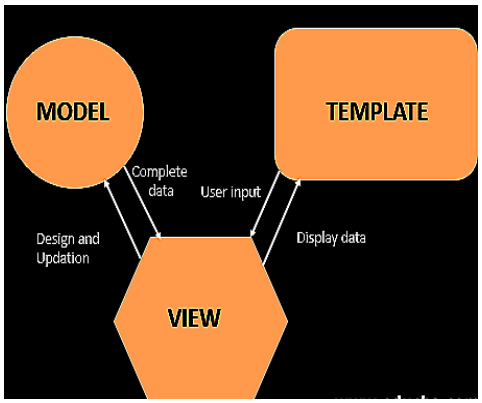

Fig. 1 outlines the design

V. GOALS

- Small business owners and managers (our target users) will have an easy to implement and use management system to increase their efficiency. They will be able to design their database to suit their product.
- The application will benefit users with non-integrated systems, little to now technical knowledge, and users with small technology budgets.
- The main goal of the application is to reduce busy work, eliminate communication errors and breakdown, integrate all aspects of a business into a centralized interface.
- The secondary goal is to reduce the busy work of employees to allow focus on value-add tasks.
- Users can directly communicate with supplier and customers by redirects from the GUI

VI. DIAGRAMS

A. *Entity Relationship Diagram*

An entity–relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity types).[2]

In Database design, the ER diagram is an abstract representation of the functions of a business.

The relationships between entities (often referred to as tables) describes how the entities interact, leading to the relational database implementation

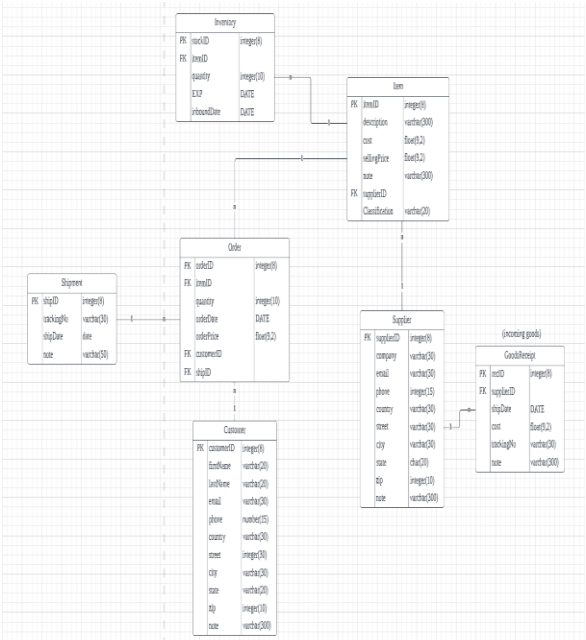

Fig. 2 Entity Relationship Diagram

B. *UML Diagram*

A UML diagram is a visual representation of a system's structure and behavior using standardized symbols and notation. It is a powerful tool for software developers and system architects to communicate complex ideas in a clear and concise manner. They are used to model everything from software applications to business processes, and can help identify issues and inefficiencies in a system's design. Below is the Skynet UML diagram

Fig. 3 UML Diagram

C. *Workflow*

The following workflow chart represents the interaction of the application with all its separate parts.

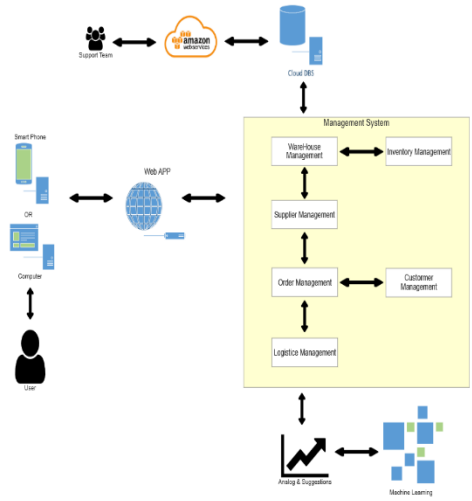

Fig. 4 Workflow Diagram

VII. MACHINE LEARNING

Machine learning involves developing algorithms and statistical models that enable computers to learn from data and improve their performance on a specific task over time. Pandas and MLxtend, along with the Apriori algorithm, have been implemented to run predictive analytics and trend recognition.

MLxtend: a powerful Python library for machine learning, which provides a wide range of tools to support data analysis, feature selection, and model selection. The library is built on top of popular data science and machine learning packages, such as, pandas, which is also used by Skynet.

Apriori Algorithm: a popular algorithm used to relating find meaning in large datasets . It identifies items that appear together in transactions above a defined threshold. These itemsets are then used to generate association rules, which are statements that describe the relationships between different items. The Apriori algorithm is particularly useful for finding associations in transactional data, such as retail sales data, where it can help identify patterns and trends that can be used for market analysis and other business purposes. Skynet implements the Apriori algorithm with MLxtend to analyze sales trends, and can be customized to fit user's needs.

VIII. CONCLUSION

We have outlined the main features and design of the Skynet application. In this project we aim to offer simple business management software for small business. Skynet is a tool that can help our users simplify, organize, and centralize their core functions. The app is not designed for large companies with the need and finances for more complex management.

REFERENCES

[1] "Entity–relationship model," Wikipedia, 26-Apr-2023. [Online]. Available: https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model. [Accessed: 08-May-2023].].

[2] "Django Project MVT structure," GeeksforGeeks, 16-Aug-2021. [Online]. Available: https://www.geeksforgeeks.org/django-project-mvt-structure/. [Accessed: 08-May-2023].