



# AI-Enabled FinTech B2B Invoice Management Application

Bibhuti Ranjan Das, GIET University, Gunupur, Odisha

Arup Kumar Padhi, GIET University, Gunupur, Odisha

## Abstract—

. The B2B world works in an unexpected way from the B2C or C2C world. Businesses work with other businesses on credit. When a buyer trade orders products from the venter commerce, the venter commerce issues a receipt for the same. This receipt for the merchandise contains different data just like the points of interest of the products acquired and when it ought to be paid. Usually known in bookkeeping wording as “Accounts Receivable” Dealer trade inter atomic with different businesses and offers goods to all of them at various times. Subsequently, the dealer commerce ought to keep track of the full sum it owes from all the buyers. This includes keeping track of all solicitations from all the buyers. However, in real-world scenarios, the solicitations are not continuously cleared i.e., paid in full sum by the due date. This finding proposes a prerequisite for a robotized framework which tracks the receipt installment from buyer.

**Keywords:** Machine Learning, React app, Java Servlet, MySQL Database, Application Programming Interface (API)

## I. INTRODUCTION

A receipt is sent by the dealer commerce when a buyer trade orders things from it. In a perfect environment, the buyer-trader transaction would be repaid within the allotted time. The buyer transaction, however, rarely makes payments within the set up time frame in the real world. With a manual receipt, preparation can often take six to ten days, can be expensive, and keeping track of installments is incredibly laborious. Therefore, sorting requests will be necessary and requires some degree of computerization; otherwise, time and effort will be lost. Acquirement and Accounts Payable can gain significant cost savings and quality improvements by switching to computerized receipt management.

### 1.1. Machine Learning

A subset of artificial intelligence (AI) called machine learning (ML) enables software applications to increase the accuracy of their predictions without having been specifically created to do so. Machine learning algorithms use historical data as input to forecast new output values.

#### 1.1.1. Python

Python is a powerful, interactive, object-oriented, and interpreted scripting language. Due to its independent platform and many built-in libraries, the Python

programming language is ideally suited for machine learning. The Python programming language is thought to be the most effective technique for automating tasks since it is more straightforward and consistent than other programming languages[1].

#### 1.1.2. Linear Regressor

A machine learning algorithm based on supervised learning is linear regression. It executes a regression operation. Regression uses independent variables to model a goal prediction value.

## II. BRIEF DESCRIPTION OF PROPOSED METHODOLOGY

The goal is to create a full-stack invoice management application that foretells when the customer will pay the invoice.

### 2.1 Project Planning

#### 2.1.1. Phase-1

In the first stage, we create a machine learning model that, given a dataset or collection of data, automates account receivables. (supervised machine learning). To predict account receivables, we must first comprehend the data, then pre-process it, do feature engineering, and then train a model.

#### 2.1.2 Phase-2

Using JS, JDBC, Java, and JSP, we construct a full-stack invoice management application in this phase that includes grid-based data visualization, search functions on the invoices, the ability to modify data in the grid's editable fields, and the ability to delete invoices as needed.

### 2.2 Project Analysis

Phase 1 is implemented first, where the data is loaded and pre-processed, in accordance with the requirements. One-hot encoding and label encoding are used to engineer the features in the data, which are then picked using lasso regression. Then, using models like linear regression, support vector regression, decision tree

regression, random forest regression, and XGB regression, the resulting features are trained. The accuracy will be determined, and the model with the highest accuracy will be chosen. The model will be kept for a future application.

### 2.3 System Architecture

The architecture of the project as shown below.

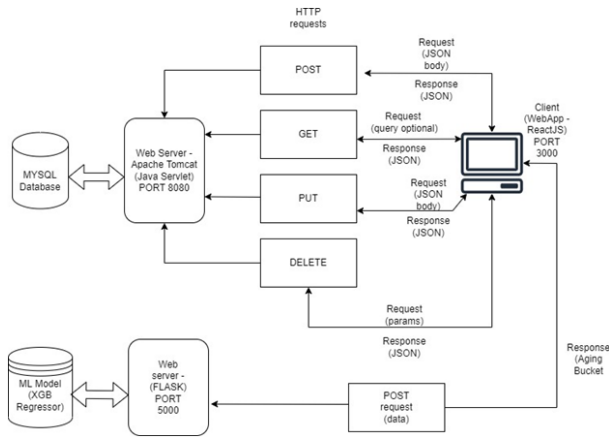


Fig. 1. Architecture of the project

## III. RESULT AND DISCUSSION

### 3.1 Implementing Machine Learning

a. Importing related libraries like

- i. Pandas
- ii. Seaborn
- iii. Matplotlib
- iv. NumPy
- v. Scikit Learn
- vi. Xgboost

b. Importing the dataset and storing it as a data frame using the pandas library and then displaying it.

c. Finding out the specifics of the data frame's shape, including its mean, median, maximum, and minimum values as well as the standard deviation, mean, and maximum values of the numerical variables.

d. Data cleaning:

- i. Checking for the null values in the data frame and checking for their percentage.
- ii. Checking for any abnormalities in the dataset and cleaning those data.
- iii. Checking for the unique values of the columns.
- iv. Dropping the columns those are irrelevant or those that are quite similar

e. Splitting of the dataset

i. Splitting the dataset into two parts

1. Maindata - It consists of the data from the actual data frame where the 'clear\_date' column does not contain null values.

2. Nulldata - It consists of the data from the actual data frame where the 'clear\_date' column contains null values.

3. Train\_test\_split function from sklearn has been used for splitting.

### 3.2. Backend API

a. An SQL dump has been provided as a script which was runned on the MYSQL workbench on the local host connection.

b. Java Servlet

- i. Creating a 'Dynamic Web Project' and attaching 'Apache Tomcat 10.0.6' as the runtime environment
- c. Flask API: This API is responsible for prediction and using the model saved from ML.

### 3.3. Frontend Application (ReactJS app)

- a. Creating a new project with 'npx create-react-app.' command to create a react project in that folder.
- b. Installing the dependencies/modules from npm using 'npm i' command.
- c. Cleaning the pre-built code
- d. Adding new folders like 'components' to store the components that will be created later

Sl No.	Business Code	Customer Number	Clear Date	Business Year	Document Id	Posting Date	Document Create Date	Due Date	Invoice Currency	Document Type	Posting Id	Total Open Amount
1	U001	20078623	2020-02-11	2020	182048861	2020-01-26	2020-01-25	2020-02-10	USD	RV	1	54273.28
2	U001	20096029	2018-08-08	2018	182064410	2018-07-22	2018-07-22	2018-08-11	USD	RV	1	7869.68
3	U001	20076274	2018-12-20	2018	182087793	2018-01-14	2018-08-14	2018-08-29	USD	RV	1	2261.98
4	CAG2	14018698	2020-02-02	2020	290102469	2020-02-01	2020-02-01	2020-04-10	CAD	RV	1	3263.7
5	U001	20078623	2018-11-23	2018	182047676	2018-11-12	2018-11-13	2018-11-23	USD	RV	1	3913.29
6	CAG2	14026181	2018-12-04	2018	280208721	2018-09-26	2018-09-30	2018-10-04	CAD	RV	1	2222.91
7	U001	20078623	2018-11-12	2018	182008873	2018-11-01	2018-10-31	2018-11-16	USD	RV	1	2358.46
8	U001	200744018	2020-01-08	2020	182069887	2020-01-19	2020-01-18	2020-04-03	USD	RV	1	11173.30
9	U001	20078623	2018-06-18	2018	182043807	2018-06-07	2018-06-05	2018-06-22	USD	RV	1	15865.94
10	U001	200780204	2018-01-04	2018	182081888	2018-01-26	2018-03-19	2018-03-07	USD	RV	1	28.83

Fig2. Web Application UI – Landing Page

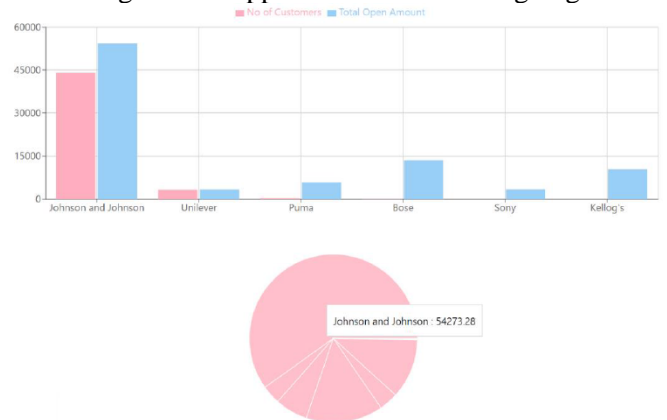


Fig3. Web Application UI – Chart View modal.

## CONCLUSION

The project offers a comprehensive method to receipt management that allows sellers' receipt information to be properly stacked, looked at, changed, and wiped. The internet application can predict the time frame in which the customer will pay the receipt amount thanks to AI features.

By putting into practice a computerized system for handling receipts now, you may support future trade growth. In case a single payment isn't paid, we are prepared to provide updates and notices to clients

by mail. Our business is currently only available in USD/CAD, but we are prepared to change the billable currency so that any client can accept a variety of currencies in the future. RPA-based arrangements can help automate the entire process. For customers, we can provide a self-service membership administration entrance. Customers can include, modify, delay, and cancel memberships, download prior solicitations, and manage their shipping and billing addresses and installment plans by using these entrances.

## REFERENCES

- [1]. M. E. Ropp and S. Gonzalez, "Development of a MATLAB/simulink model of a single-phase grid-connected photovoltaic system," *IEEE Trans. Energy Conv.*, vol. 24, no. 1, pp. 195–202, Mar. 2009.
- [2]. Sahu, R., Panigrahi, P. K., & Lal, D. K. (2020, December). Control and protection of hybrid smart-grid power system: A review. In 2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC) (pp. 1-6). IEEE.
- [3]. J. M. Carrasco, L. G. Franquelo, J. T. Bialasiewicz, E. Galvan, R. C. PortilloGuisado, M. A. M. Prats, J. I. Leon, and N. Moreno-Alfonso, "Power-electronic systems for the grid integration of renewable energy sources: A survey," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1002–1016, Aug. 2006.
- [4]. Sahu, R., Misra, S., Sandhya, Y. B., Panigrahi, P. K., & Prasad, G. S. (2023). Smart controller PV system for stochastic environmental condition. *Materials Today: Proceedings*, 80, 2553-2556.
- [5]. V. G. Agelidis, D. M. Baker, W. B. Lawrance, and C. V. Nayar, "A multilevel PWM inverter topology for photovoltaic applications," in *Proc. IEEE ISIE, Guimarães, Portugal, 1997*, pp. 589–594.
- [6]. S. Kouro, J. Rebolledo, and J. Rodriguez, "Reduced switching-frequency modulation algorithm for high-power multilevel inverters," *IEEE Trans. Ind. Electron.*, vol. 54, no. 5, pp. 2894–2901, Oct. 2007.
- [7]. Rakesh, S., Digbijoy, M., Tanushree, M., & Nishant, P. (2016). Designing and study standalone hybrid energy system: for technical institutes. *Int J Inf Res Rev*, 3(6).
- [8]. S. J. Park, F. S. Kang, M. H. Lee, and C. U. Kim, "A new single-phase five level PWM inverter employing a deadbeat control scheme," *IEEE Trans. Power Electron.*, vol. 18, no. 18, pp. 831–843, May 2003.
- [9]. Sahu, Rakesh, et al. "Smart Operations Of Smart Grids Integrated With Fuel Cell With Controlling Strategies" *International Journal of Engineering and Advanced Technology*, Volume-8 Issue-6, August, 2019. DOI: 10.35940/ijeat.F8635.088619.
- [10]. L. M. Tolbert and T. G. Habetler, "Novel multilevel inverter carrier-based PWM method," *IEEE Trans. Ind. Appl.*, vol. 35, no. 5, pp. 1098–1107, Sep./Oct. 1999.
- [11]. Sahu, R., Das, B., Sabat, R. R., & Swain, M. K. (2013). Hybrid Power Model of Renewable Energy Sources For On Grid Power Supply. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2(11), 5466-5473.
- [12]. [https://en.wikipedia.org/wiki/Python\\_\(Programming\\_language\)](https://en.wikipedia.org/wiki/Python_(Programming_language))
- [13]. <https://scikit-learn.org/>
- [14]. <https://pandas.pydata.org/>
- [15]. <https://numpy.org/doc/stable/>
- [16]. <https://seaborn.pydata.org/>
- [17]. <https://matplotlib.org/>
- [18]. <https://nodejs.org/en/>
- [19]. <https://reactjs.org/>
- [20]. <https://mui.com/>
- [21]. <https://www.javatpoint.com/servlet-tutorial>