

A PROJECT REPORT
on
“AI ENABLED FINTECH B2B INVOICE MANAGEMENT
APPLICATION”

Submitted to
KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of
BACHELOR’S DEGREE IN COMPUTER SCIENCE &
ENGINEERING

BY
RACHEET PRADHAN 1905112

UNDER THE GUIDANCE OF

Ashutosh Sarangi Areej Aftab

Affiliation



SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

May 2022

KIIT Deemed to be University

School of Computer Engineering

Bhubaneswar, ODISHA 751024



CERTIFICATE

This is certify that the project entitled

“AI ENABLED FINTECH B2B INVOICE
MANAGEMENTAPPLICATION “

submitted by

RACHEET PRADHAN 1905112

is a record of bonafide work carried out by him, in the partial fulfilment of the requirement for the award of **Degree of Bachelor of Engineering** (Computer Science & Engineering) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2022-2023, under our guidance.

Date: 14/04/2022

Ashutosh Sarangi Areej Aftab

Project Mentor

Acknowledgements

I am sincerely grateful to **Ashutosh Sarangi and Areej Aftab**

of **High Radius Corporation** for their continued support and constant encouragement throughout to see that this project achieves its target from its initiation to its completion. They helped me and guided me throughout the internship and provided me with the skills and knowledge required for the successful fruition of the project.

I am also thankful to my moderators who guided me throughout the course of the training with their valuable inputs, cooperation and guidance.

I am also thankful to Kalinga Institute of Industrial Technology, Bhubaneswar for providing me with this opportunity along with the technical skills and facilities which proved to be very useful for my project.

RACHEET PRADHAN

B.TECH (CSE)

1905112

ABSTRACT

The B2B world operates differently from the B2C or C2C world. Businesses work with other businesses on credit. When a buyer orders goods from the seller, the seller issues an order invoice for the same. This invoice for the goods contains several information like the details of the goods purchased and payment date.

One of the biggest problems that businesses face today is of poor and delayed payment practices where the customers fail to make payment on time. This has a very large impact on businesses like Small Medium Enterprises (SMEs) who face the problem of having limited cash flow. Late payments problem has resulted in businesses facing severe consequences and sometimes, causing them to shut down their business. In fact, many businesses have to resort to tools like invoice finance to help overcome the problems of receiving late payment.

Its responsibilities include:

- ❖ Collecting payments from customers for their past due invoices
- ❖ Sending reminders and follow ups to the customers for payments to be made
- ❖ Looking after the entire process of getting the cash inflow
- ❖ Help the company get paid for the services and products supplied

As a result, this project creates a user-friendly application with the goal of providing correct output results even when a big number of customers are involved. The responsive end- to-end application has been built using complex technologies and an elaborate framework that stands out in the market owing to its usage of new technologies such as Machine Learning, Python, Artificial Intelligence, ReactJS, among others. The application consists of two parts:- a comprehensive Data Science and Machine Learning model and an application design and user interface development methodology.

Keywords: Artificial Intelligence, Machine Learning, Python, MySQL, Java, ReactJS

Contents

1	Introduction	1
2	Background	2
2.1	Machine Learning	2
2.2	Frontend	4
2.3	Backend	4
3	Problem Statement / Requirement Specifications	5
3.1	Overview & Dataset Features	5
3.2	Machine Learning Model	6
3.3	Objective Of Java	7
3.3.1	Data Loading in the Database	7
3.3.2	Servlet Creation	8
4	Implementation	13
4.1	Results Of The Project	13
4.2	Discussion of Project	13
5	Conclusion and Future Scope	14
5.1	Conclusion	14
5.2	Future Scope	14
	References	15

Chapter 1

Introduction

In the B2B world, Account Receivables play a very major role for the business to run smoothly. A computer software package that allows firms to manage their orders and inventory is known as an invoice management system. Invoice management systems aid in more accurate inventory management by automatically entering new inventory into the system, tracking sales across various selling platforms such as Flipkart and Amazon, and alerting the business owner, when your stock of a particular item drops low enough to facilitate a re-order.

Beginning with the client order and continuing through payment reconciliation, fulfilment, and shipment, an invoice management system can automate the order-to-cash process. Invoice management software can be used by organizations of all sizes, both B2B and B2C.

Order management software is also shareable, from the customer service team to the accounting team, the warehouse staff, and you, the business owner. Effective invoice management improves the business work-flow and increases the likelihood of repeat customers

Chapter 2

Basic Concepts

The project was broadly divided into three parts:

1. Machine Learning:

To build a Machine Learning model to predict the date of payment of an invoice.

2. React:

the frontend, which is used to build an AI-Enabled FinTech B2BInvoice Management Application by having various grids for Header, footer, Data Table, buttons etc.

3. Java:

The backend which is used to build an AI-Enabled FinTech B2BInvoice Management Application by connecting to database and making frontend responsive.

3.1 MACHINE LEARNING

The objective was to build a Machine Learning Model to predict the payment date of an invoice when it gets created in the system. It would also categorize the invoice into different buckets based on the predicted payment date. Clicking on the Predict button will populate the Predicted Payment Date and the Predicted Aging Bucket column of the grid with the predicted values derived from the ML model.

3.1.1. The order is placed

The consumer places an order using a third-party sales site, personal website, or a real agent on the phone. Customers will enter their information on a standardized form online, with the option of saving a secure preferred payment method. Make all fields of your online form required to optimize the sales process and ensure you get all of the relevant contact information for the customer up front. This establishes a customer profile, which your invoicing management system may use to keep track of their purchase history, order volume, and payment and delivery preferences. It also provides their phone number and email address in case you need to contact them regarding service recovery. Once your software system accepts the charges, the payment is executed, and the order is sent to the warehouse.

3.1.2. Warehouse processing

Once the order arrives at the warehouse, it's checked by the intake team and the item or items are "picked" from the stock. Having a Stock Keeping Unit (SKU) and barcode for every item increases the accuracy of tracking and makes it easier for pickers to simply scan the item and add it to the order. If there isn't enough of the item(s) in stock to fulfill an order, then a purchase order is automatically placed through the invoice management software. Owner and the warehouse manager will receive an alert that there may be a delay in fulfillment. The customer may receive an automatic notification of the delay, and the customer service team can follow up with the customer.

3.1.3. Reconciling the order

Now, the order is sent to the accounting department or preferably it should sync automatically with the cloud accounting software, where it's recorded in the Accounts Receivables (A/R) ledger. The sale is logged and a receipt sent to your client. Automating the sales ledgers makes it easier for auditing, inventory reconciliation, and End-of-year taxes.

3.1.4. Shipping the order

Once the order is picked from the warehouse, the packing team will double-check for accuracy, again using the barcodes and SKU. Then, the order is packed carefully and shipped via a third-party delivery system. The customer will receive a notice through the invoice management system that their order has shipped, along with a tracking number and estimated delivery time. As a store owner, you can also track the progress of shipped orders, which can be helpful if there are special needs orders, such as re-deliveries, VIP orders, or unusually large ones.

3.1.5. Post-sales follow-up

Once the order arrives, the software should generate an automatic email to follow up, asking how they liked the items and ensuring that they received everything accurately. This email should include detailed instructions on how to reach customer service if there are any issues, taking the frustration out of guessing how to obtain a refund if needed. The customer service team oversees this process, thanking the customer for their business or working with them for a refund or replacement.

3.1.6. Special order oversight

Another aspect of good Invoice management system is the ability to flag a special order. This may be a return/replacement or it could be a VIP order that includes a free thank-you gift or special coupon. When these orders are placed through the system, the software can flag them with a code, allowing the owner or customer retention team to personally monitor the order for accuracy.

2.2.2 FRONTEND

2.2.2. ReactJs

ReactJs is an open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality. In this project ReactJs is the frontend technology used. It is used to design the User Interface which is the dashboard of the Invoice Management Application.

The app was powerful and included several concepts such as working with concepts, hooks, redux and more, including endless scrolling, which is a web design that loads continuous content as a user scrolls down the page, eliminating the need for pagination. Redux is a state management tool. With redux the state of the application is kept in a store, and each component can access any state that it needs from this store. It allows React components to read data from a Redux Store, and dispatch Actions to the Store to update data. Redux helps apps to scale by providing a sensible way to manage state through a unidirectional data flow model.

2.2.3. Java Servlets

A Java Servlet is a Java software component that enhances a server's functionality. Although servlets can reply to a wide range of requests, they are most typically used to host web applications on web servers, making them a server-side servlet web API. Other dynamic online content platforms, such as PHP and ASP.NET, have Java counterparts in the form of web servlets. Java Servlets are utilised in the project to create an Application Programming Interface (API) that connects the backend and frontend properly.

Chapter 3

Problem Statement / Requirement Specifications

In this given project we were required to make a full stack Invoice Management application and also to build a Machine Learning Model to predict the date of payment and aging bucket of an invoice. The technologies used are:-

- 1) Machine Learning using Python
- 2) Java, SQL (Backend)
- 3) HTML, CSS, JavaScript, ReactJS (Frontend)

3.1 The dataset that used had following features:

- i) business_code - company code of the account
- ii) cust_number - customer number given to all the customers of the Account
- iii) name_customer - name of the customer.
- iv) cust_number - Each customer has a number that uniquely identifies it.
- v) document_create_date - The date on which the invoice document was created
- vi) document_create_date_1 - Normalized date of the invoice document
- vii) posting_id - key indicator to identify whether an AR item is an invoice.
- viii) due_in_date - The date on which the customer is expected to clear an invoice
- ix) invoice_id - Unique number assigned when a seller creates an Invoice.
- x) baseline_create_date - The date on which the Invoice was created.
- xi) total_open_amount - The amount that is yet to be paid for that invoice
- xii) invoice_amount - The total amount for that invoice.
- xiii) cust_payment_terms - Business terms and agreements between customers and accounts on discounts and days of payment
- xiv) area_business - Business area in sap is defined as an organizational area within the financial accounting module.
- xv) clear_date - The date on which the customer clears an invoice, or in simple terms, they make the full payment.
- xvi) is_open - indicator of whether an invoice is open or closed.
- xvii) invoice_currency - The currency of the invoice amount in the document for the invoice.
- xviii) doc_id - It is also a unique identifier of an invoice and is a primary key.
- xix) total_open_amount - open amount of an invoice

business_code	cust_number	name_customer	cle_date	business_doc_id	posting_date	document	document_due_in	di_invoice	documentposting	ic_area_busi	total_ope	baseline_cust_payn	invoice_id	isOpen	
1	U001	200769623 WAL-MAR corp	11-02-2020 00:00	2020 1930438491	26-01-2020	20200125	20200126	20200210	USD	RV	1	54273.28	20200126 NAH4	1930438491	0
2	U001	200980828 BEN E	08-08-2019 00:00	2019 1929646410	22-07-2019	20190722	20190722	20190811	USD	RV	1	79656.6	20190722 NAD1	1929646410	0
3	U001	200792734 MDV/ trust	30-12-2019 00:00	2019 1929873765	14-09-2019	20190914	20190914	20190929	USD	RV	1	2253.86	20190914 NAA8	1929873765	0
4	CA02	140105686 SYSC llc		2020 2960623488	30-03-2020	20200330	20200330	20200410	CAD	RV	1	3299.7	20200331 CA10	2960623488	1
5	U001	200769623 WAL-MAR foundation	25-11-2019 00:00	2019 1930147974	13-11-2019	20191113	20191113	20191128	USD	RV	1	33133.29	20191113 NAH4	1930147974	0
6	CA02	140106181 THE corporation	04-12-2019 00:00	2019 2960581231	20-09-2019	20190920	20190920	20191004	CAD	RV	1	22225.84	20190924 CA10	2960581231	0
7	U001	200769623 WAL-MAR in	12-11-2019 00:00	2019 1930083373	01-11-2019	20191031	20191101	20191116	USD	RV	1	7358.49	20191101 NAH4	1930083373	0
8	U001	200744019 TARG us		2020 1930659387	19-03-2020	20200318	20200319	20200403	USD	RV	1	11173.02	20200319 NAA8	1930659387	1
9	U001	200769623 WAL-MAR corp	18-06-2019 00:00	2019 1929439637	07-06-2019	20190605	20190607	20190622	USD	RV	1	15995.04	20190607 NAH4	1929439637	0
10	U001	200762301 C&S WH systems	06-03-2019 00:00	2019 1928819386	20-02-2019	20190219	20190220	20190307	USD	RV	1	28.63	20190220 NAC6	1928819386	0
11	U001	200418007 AM		2020 1930610806	11-03-2020	20200306	20200311	20200326	USD	RV	1	3525.59	20200311 NAA8	1930610806	1
12	U001	200743129 BROOKS co	22-01-2019 00:00	2019 1928550622	02-01-2019	20190102	20190102	20190117	USD	RV	1	103147.4	20190102 NAA8	1928550622	0
13	U001	200186937 SYSC corporation	06-05-2019 00:00	2019 1929151655	15-04-2019	20190415	20190415	20190430	USD	RV	1	16381.45	20190415 NAA8	1929151655	0
14	U001	200721222 GO corporation	01-11-2019 00:00	2019 1930022117	17-10-2019	20191017	20191017	20191101	USD	RV	1	19581.57	20191017 NAA8	1930022117	0
15	U001	200739534 OK systems		2020 1930788296	15-04-2020	20200415	20200415	20200430	USD	RV	1	121105.7	20200415 NAA8	1930788296	1
16	U001	200353024 DECA corporation		2020 1930817482	23-04-2020	20200423	20200423	20200426	USD	RV	1	3726.06	20200416 NAM2	1930817482	1
17	U001	200794332 COST in	12-11-2019 00:00	2019 1930052739	25-10-2019	20191025	20191025	20191109	USD	RV	1	5181.51	20191025 NAA8	1930052739	0
18	U001	200881076 ALBERT corp	17-12-2019 00:00	2019 1920209407	02-12-2019	20191202	20191202	20191217	USD	RV	1	19082.14	20191202 NAA8	1920209407	0
19	U001	200769623 WAL-MAR trust	26-11-2019 00:00	2019 1930153511	15-11-2019	20191114	20191115	20191130	USD	RV	1	12819.5	20191115 NAH4	1930153511	0
20	U001	200769623 WAL-MAR systems	05-02-2020 00:00	2020 1930438462	24-01-2020	20200124	20200124	20200208	USD	RV	1	53349.54	20200124 NAH4	1930438462	0
21	U013	100053554 SYSTEMS systems	11-02-2020 00:00	2020 1991837617	11-01-2020	20200107	20200111	20200210	USD	RV	1	5683.5	20200111 NAVE	1991837617	0
22	U001	200787334 FAREW us	04-09-2019 00:00	2019 1929773400	21-08-2019	20190822	20190821	20190905	USD	RV	1	11845.19	20190821 NAA8	1929773400	0
23	U001	200744019 TARG associates		2020 1930676042	21-03-2020	20200320	20200321	20200405	USD	RV	1	5893.01	20200321 NAA8	1930676042	1
24	U001	100006311 QUALITY C	30-07-2019 00:00	2019 1929626925	17-07-2019	20190717	20190717	20190801	USD	RV	1	20079.58	20190717 NAA8	1929626925	0
25	U001	200769623 WAL-MAR co	04-02-2020 00:00	2020 1930431304	24-01-2020	20200123	20200124	20200208	USD	RV	1	11307.78	20200124 NAH4	1930431304	0
26	CA02	140106408 WAL-M corp		2020 2960618790	06-03-2020	20200306	20200306	20200316	CAD	RV	1	92832.27	20200306 CA10	2960618790	1
27	U001	200769623 WAL-MAR foundation	30-01-2019 00:00	2019 1928620435	14-01-2019	20190114	20190114	20190129	USD	RV	1	36188.11	20190114 NAH4	1928620435	0
28	U001	200337148 COAS llc		2020 1930592246	02-03-2020	20200302	20200302	20200317	USD	RV	1	11380.83	20200302 NAA8	1930592246	1
29	U001	200769623 WAL-MAR foundation	09-05-2019 00:00	2019 1929194820	23-04-2019	20190423	20190423	20190508	USD	RV	1	412.09	20190423 NAH4	1929194820	0
30	CA02	140106408 WAL-M associates	06-05-2019 00:00	2019 2960542344	16-04-2019	20190416	20190416	20190426	CAD	RV	1	48413.28	20190416 CA10	2960542344	0
31	U001	200750051 ALBER systems	03-05-2019 00:00	2019 1929170780	18-04-2019	20190418	20190418	20190503	USD	RV	1	6023.53	20190418 NAA8	1929170780	0
32	U001	200721222 GO trust	07-10-2019 00:00	2019 1929907681	22-09-2019	20190921	20190922	20191007	USD	RV	1	17307.51	20190922 NAA8	1929907681	0
33	U001	200769623 WAL-MAR in	23-09-2019 00:00	2019 1929847863	10-09-2019	20190909	20190910	20190925	USD	RV	1	341.14	20190910 NAH4	1929847863	0
34	U001	200726979 B/S systems	01-07-2019 00:00	2019 1929541405	15-06-2019	20190625	20190615	20190630	USD	RV	1	365.87	20190615 NAA8	1929541405	0
35	U001	CCCA02 KRAFT systems	30-08-2019 00:00	2019 1929535888	05-07-2019	20190701	20190705	20190809	USD	RV	1	32751.88	20190705 NAG2	1929535888	0
36	U001	200794332 COST associates		2020 1930731070	07-04-2020	20200401	20200407	20200422	USD	RV	1	3863.93	20200407 NAA8	1930731070	1
37	U001	200700919 US corp	13-02-2019 00:00	2019 1928656691	23-01-2019	20190121	20190123	20190122	USD	RV	1	39221.31	20190123 NAD1	1928656691	0

Fig 3.1 Dataset Of The Invoice Management

3.2. Machine Learning Model

While building this model we had gone through many process:

1. Firstly we needed to import the dataset and also import various data manipulation libraries.
2. We visualized the dataset in the form of a data frame to get a brief idea about the dataset.
3. Identified the target variable which is clear_date.
4. Independent and dependent variables identification and extraction.
5. Handling Missing Values using Null Imputation Techniques.
6. Encoding Categorical Variables to training purposes.
7. Splitting the dataset for train/test and validation.
8. Feature Scaling for improved training using Normalization and Standardization.
9. Training and Validating ML model.
10. Predict clear_date using test data.
11. Prepare aging bucket by subtracting invoice creation date from predicted clear date.

The different buckets were:-

1. 0-15 days
2. 16-30 days
3. 31-45 days
4. 46-60 days
5. Greater than 60 days

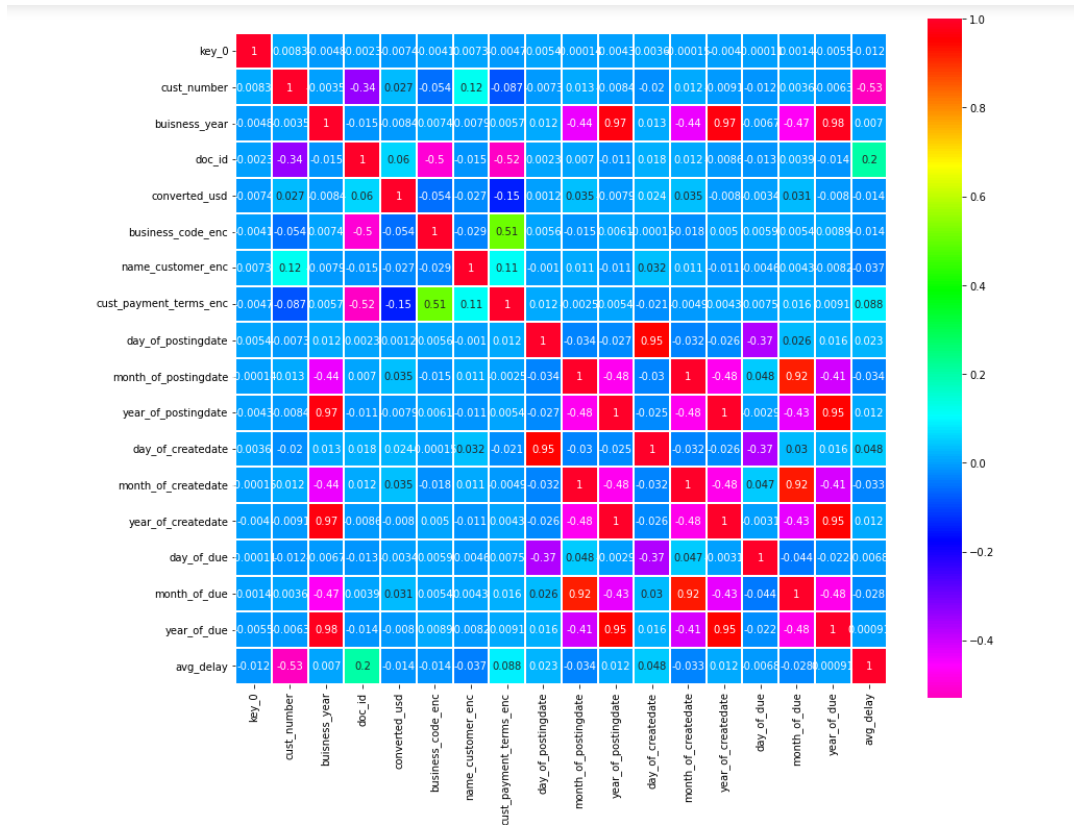


Fig 3.2 Heat map of ML model

3.3. Objective of Java:

3.3.1. Data Loading in the Database:

1. Execute the SQL script for the creation of table.
2. Read the csv datasheet using a CSV reader and stored information.
3. We used a JDBC driver and also made a POJO class which helped us load the datasheet into the database in batches.

id	business_code	cust_number	clear_date	business_year	doc_id	posting_date	document_create_date	document_create_date1	due_in_date	invoice_currency	document_type	posting_id	area_business	total_open_amount	baseline_create_da
1	20001	200769423	2020-02-11	2020	1920439491	2020-01-26	2020-01-25	2020-01-26	2020-02-10	USD	INV		1	54273.20	2020-01-26
2	20001	20080828	2019-08-08	2019	1920644410	2019-07-22	2019-07-22	2019-07-22	2019-08-11	USD	INV		1	79656.4	2019-07-22
3	20001	200782784	2019-12-30	2019	19208773765	2019-09-14	2019-09-14	2019-09-14	2019-09-29	USD	INV		1	2253.64	2019-09-14
4	20001	140106048	2000-00-00	2020	2040622489	2020-02-30	2020-02-30	2020-02-30	2020-04-10	CAD	INV		1	2299.7	2020-02-30
5	20001	200769423	2019-11-25	2019	1930147974	2019-11-13	2019-11-13	2019-11-13	2019-11-29	USD	INV		1	3313.29	2019-11-13
6	20001	140106181	2019-12-04	2019	2040551231	2019-09-20	2019-09-20	2019-09-20	2019-10-04	CAD	INV		1	22225.64	2019-09-24
7	20001	200769423	2019-11-12	2019	1930089373	2019-11-01	2019-10-31	2019-11-01	2019-11-14	USD	INV		1	7888.49	2019-11-01
8	20001	200744019	2000-00-00	2020	1930658387	2020-03-19	2020-03-18	2020-03-19	2020-04-03	USD	INV		1	11173.62	2020-03-19
9	20001	200769423	2019-04-18	2019	1920439491	2019-04-07	2019-04-05	2019-04-07	2019-04-22	USD	INV		1	15995.64	2019-04-07
10	20001	200742301	2019-03-06	2019	1920819386	2019-02-20	2019-02-19	2019-02-20	2019-03-07	USD	INV		1	28.43	2019-02-20
11	20001	200410007	2000-00-00	2020	1930618006	2020-03-11	2020-03-11	2020-03-11	2020-03-24	USD	INV		1	9325.99	2020-03-11
12	20001	200743129	2019-01-22	2019	1920550422	2019-01-02	2019-01-02	2019-01-02	2019-01-17	USD	INV		1	103147.37	2019-01-02
13	20001	200184937	2019-05-04	2019	1920151655	2019-04-15	2019-04-15	2019-04-15	2019-04-30	USD	INV		1	14381.45	2019-04-15
14	20001	200712122	2019-11-01	2019	1930022117	2019-10-17	2019-10-17	2019-10-17	2019-11-01	USD	INV		1	19861.87	2019-10-17
15	20001	200738534	2000-00-00	2020	1930788294	2020-04-15	2020-04-15	2020-04-15	2020-04-30	USD	INV		1	121105.45	2020-04-15
16	20001	200333024	2000-00-00	2020	1930817482	2020-04-23	2020-04-23	2020-04-23	2020-04-26	USD	INV		1	3724.64	2020-04-16
17	20001	200744332	2019-11-12	2019	1930052739	2019-10-25	2019-10-25	2019-10-25	2019-11-09	USD	INV		1	5181.51	2019-10-25
18	20001	200810740	2019-12-17	2019	1930209497	2019-12-02	2019-12-02	2019-12-02	2019-12-17	USD	INV		1	34902.14	2019-12-02
19	20001	200769423	2019-11-26	2019	1930155511	2019-11-15	2019-11-14	2019-11-15	2019-11-30	USD	INV		1	12019.5	2019-11-15
20	20001	200769423	2020-02-05	2020	1930438462	2020-01-24	2020-01-24	2020-01-24	2020-02-08	USD	INV		1	53349.54	2020-01-24
21	20001	10003584	2020-02-11	2020	1930371617	2020-01-11	2020-01-07	2020-01-11	2020-02-10	USD	INV		1	5483.9	2020-01-11
22	20001	200737314	2019-08-04	2019	1920773405	2019-08-21	2019-08-21	2019-08-21	2019-08-25	USD	INV		1	11845.19	2019-08-21
23	20001	200744019	2000-00-00	2020	1930476042	2020-03-21	2020-03-20	2020-03-21	2020-04-05	USD	INV		1	5893.61	2020-03-21
24	20001	100006311	2019-07-30	2019	1920424828	2019-07-17	2019-07-17	2019-07-17	2019-08-01	USD	INV		1	20079.58	2019-07-17
25	20001	200769423	2020-02-04	2020	1930431374	2020-01-24	2020-01-23	2020-01-24	2020-02-08	USD	INV		1	11307.76	2020-01-24
26	20001	140106048	2000-00-00	2020	2040618790	2020-03-06	2020-03-06	2020-03-06	2020-03-16	CAD	INV		1	92032.27	2020-03-06
27	20001	200769423	2019-01-30	2019	1920820439	2019-01-14	2019-01-14	2019-01-14	2019-01-29	USD	INV		1	36188.11	2019-01-14
28	20001	200337148	2000-00-00	2020	1930592246	2020-03-02	2020-03-02	2020-03-02	2020-03-17	USD	INV		1	11390.63	2020-03-02
29	20001	200769423	2019-08-09	2019	1920134625	2019-08-23	2019-08-23	2019-08-23	2019-08-25	USD	INV		1	412.09	2019-08-23
30	20001	140106048	2019-05-06	2019	2040542344	2019-04-16	2019-04-16	2019-04-16	2019-04-26	CAD	INV		1	49433.28	2019-04-16
31	20001	20075001	2019-05-03	2019	1920170780	2019-04-18	2019-04-18	2019-04-18	2019-05-03	USD	INV		1	6023.53	2019-04-18
32	20001	200712122	2019-10-07	2019	1920907891	2019-09-22	2019-09-21	2019-09-22	2019-10-07	USD	INV		1	17307.51	2019-09-22
33	20001	200769423	2019-09-23	2019	1920847863	2019-09-10	2019-09-09	2019-09-10	2019-09-25	USD	INV		1	341.14	2019-09-10
34	20001	200726979	2019-07-01	2019	1920841405	2019-06-15	2019-06-15	2019-06-15	2019-06-30	USD	INV		1	365.87	2019-06-15
35	20001	200744332	2000-00-00	2020	1930731070	2020-04-07	2020-04-01	2020-04-07	2020-04-22	USD	INV		1	3863.93	2020-04-07
36	20001	200700919	2019-02-13	2019	1920856491	2019-01-23	2019-01-21	2019-01-23	2019-02-12	USD	INV		1	59221.31	2019-01-23

Fig 3.3 Database connection of MySQL

3.3.2 Servlet Creation

So after the UI is made some actions (add, edit, delete etc.) needs to perform. So using the help of servlets given below we can make those actions happen.

1. Add servlet - Get a POST request from the frontend with parameters such as invoice amount, notes, date, etc. and pass them to the SQL database.
2. Edit Servlet - GET a POST request from the frontend with parameters such as doc_id to identify the invoice in addition to the parameters which need to be changed.
3. Delete Servlet - Delete the selected invoices from the database by passing their respective doc_id's to identify them in the database.
4. Search Servlet - Get the invoice number from the frontend and pass them as a http request using axios to the backend and search through the database and return it to the frontend again.
5. Data Display Servlet - Display the table of invoices to the frontend.

3.4. Objective of React (UI Representation)

3.4.1. Header Section

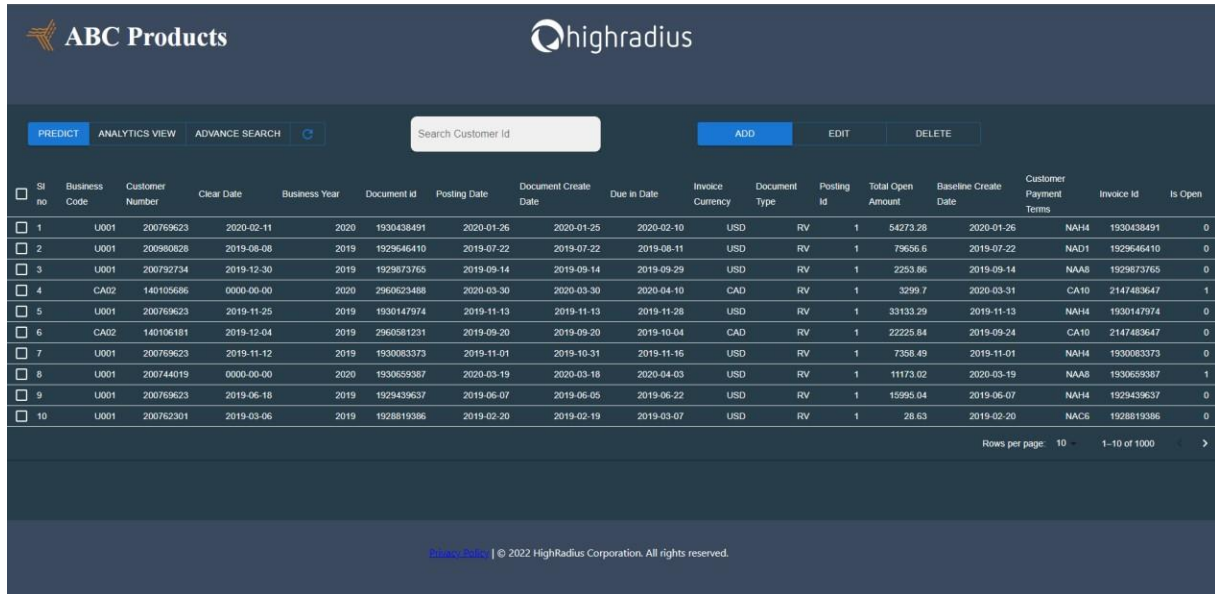
The header consists of:-

1. Account name logo on the left
2. The HighRadius Logo in the center.

3.4.2. Grid Panel Section

The Grid panel section will be divided into 4 portions:

1. The header of the grid will have a Predict button on the top left corner followed by a View Correspondence Button, an Add Button, an Edit Button, a Delete Button and a Search Bar.
2. The name of the grid that is Invoice List will be mentioned in the top left corner of the grid.
3. The second portion is the table with customer invoice data as rows and the columns



SI no	Business Code	Customer Number	Clear Date	Business Year	Document Id	Posting Date	Document Create Date	Due In Date	Invoice Currency	Document Type	Posting Id	Total Open Amount	Baseline Create Date	Customer Payment Terms	Invoice Id	Is Open
1	U001	200769623	2020-02-11	2020	1930438491	2020-01-26	2020-01-25	2020-02-10	USD	RV	1	54273.28	2020-01-26	NAH4	1930438491	0
2	U001	200980828	2019-08-08	2019	1929646410	2019-07-22	2019-07-22	2019-08-11	USD	RV	1	79656.6	2019-07-22	NAD1	1929646410	0
3	U001	200792734	2019-12-30	2019	1929873765	2019-09-14	2019-09-14	2019-09-29	USD	RV	1	2253.86	2019-09-14	NAA8	1929873765	0
4	CA02	140105686	0000-00-00	2020	2960623488	2020-03-30	2020-03-30	2020-04-10	CAD	RV	1	3299.7	2020-03-31	CA10	2147483647	1
5	U001	200769623	2019-11-25	2019	1930147974	2019-11-13	2019-11-13	2019-11-28	USD	RV	1	33133.29	2019-11-13	NAH4	1930147974	0
6	CA02	140106181	2019-12-04	2019	2960581231	2019-09-20	2019-09-20	2019-10-04	CAD	RV	1	22225.84	2019-09-24	CA10	2147483647	0
7	U001	200769623	2019-11-12	2019	1930083373	2019-11-01	2019-10-31	2019-11-16	USD	RV	1	7368.49	2019-11-01	NAH4	1930083373	0
8	U001	200744019	0000-00-00	2020	1930659387	2020-03-19	2020-03-18	2020-04-03	USD	RV	1	11173.02	2020-03-19	NAA8	1930659387	1
9	U001	200769623	2019-06-18	2019	1929439637	2019-06-07	2019-06-05	2019-06-22	USD	RV	1	15995.04	2019-06-07	NAH4	1929439637	0
10	U001	200762301	2019-03-06	2019	1928819386	2019-02-20	2019-02-19	2019-03-07	USD	RV	1	28.63	2019-02-20	NAC6	1928819386	0

Fig 3.4 UI of Invoice management Application

List of all the buttons present in the Grid Panel with their functionalities are as follows:

- 1. Predict Button** - The Predict button will remain in disabled state if no rows are selected. Whenever one or more rows are selected, the Predict button will be activated. After clicking on the Predict button, the Predicted Payment Date and Predicted Aging Bucket will be populated for the respective records.
- 2. View Analytics Button** - Helps in viewing an invoice or a group of invoices in detail and exporting it.

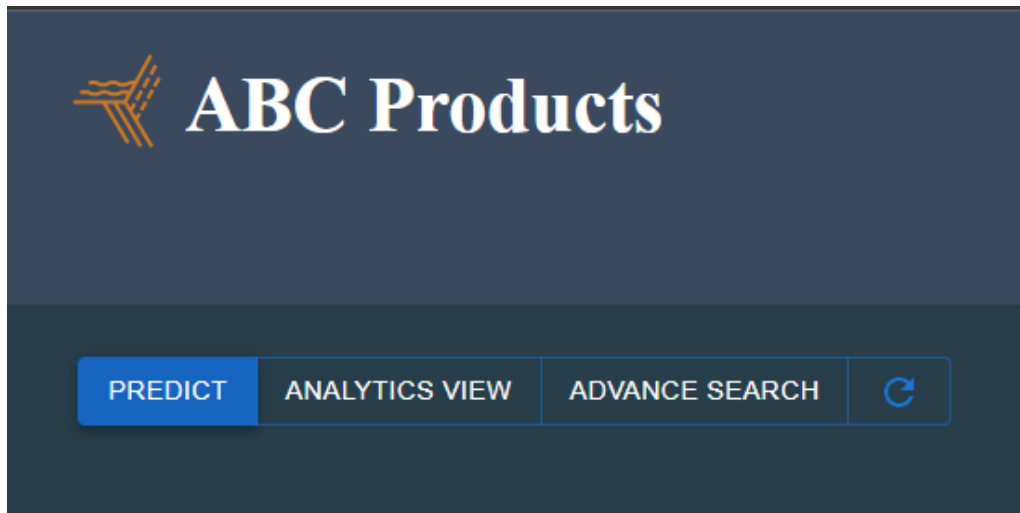


Fig 3.5 Prediction and Analytics button

3. Add Button - Adds an Invoice to the existing database

The screenshot shows the 'Add' form in the ABC Products application. The form is titled 'Add' and contains several input fields for adding a new invoice. The fields are arranged in a grid-like structure. On the left, there is a sidebar with a list of business codes and their corresponding invoice IDs. On the right, there is a table showing the list of invoices with columns for 'INVOICE ID' and 'Is Open'.

Form Fields:

- 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
- Baseline Clear Date
- Customer Payment Terms
- Invoice Id

Buttons: ADD, CANCEL

Table:

INVOICE ID	Is Open
10204308491	0
10276494410	0
10256737305	0
21474036447	1
10201479774	0
21474036447	0
10200033373	0
10200033307	1
10294308337	0
10208193066	0

1-10 of 1000

Fig 3.6 Add button

4. Edit Button - Edits an invoice in the database

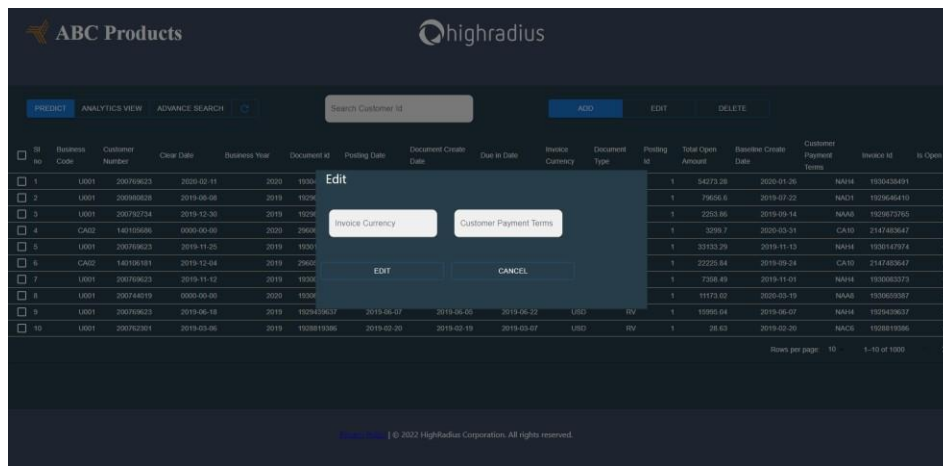


Fig 3.7 Edit button view

5. Delete Button - Deletes an invoice in the database

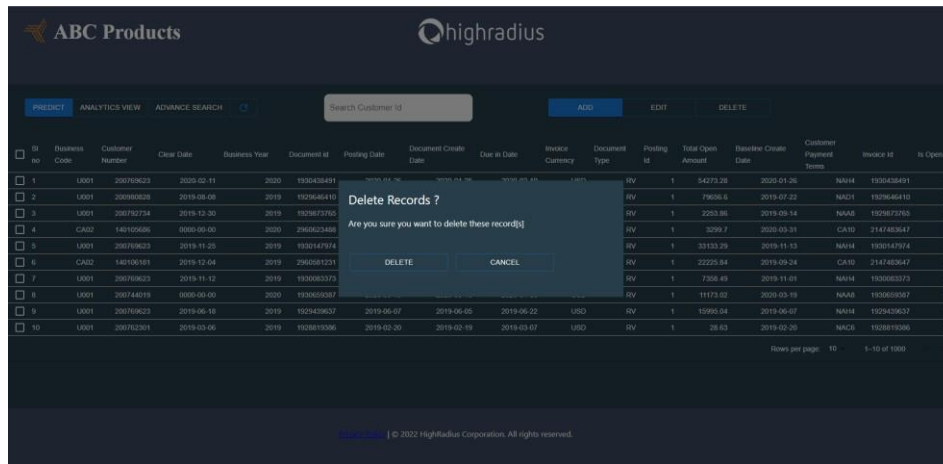


Fig 3.8 Delete records view

6. Search For Invoices - Search an invoice by the invoice number

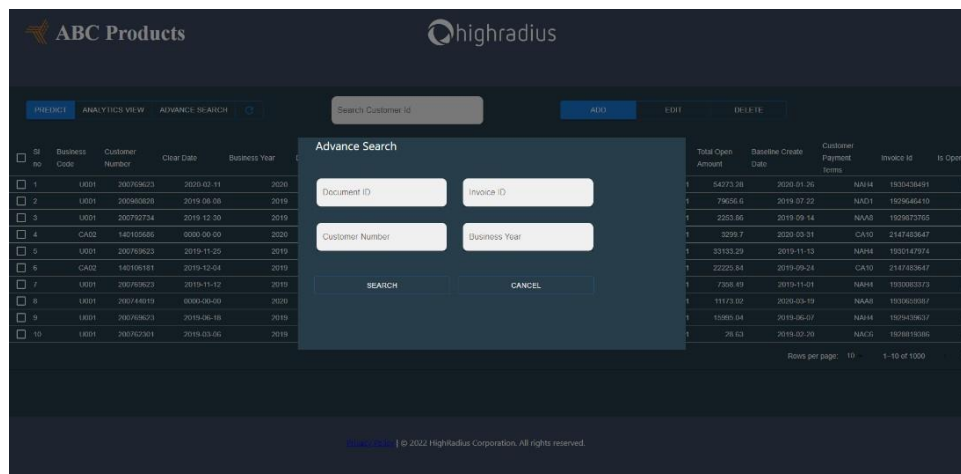


Fig 3.9 Advanced search

Chapter 4

4.1. Results of the Project:

On our data set, we examined how the various features interact to produce findings that allow us to draw a number of inferences. Area business is an empty feature in the ML model and does not support the modelling process. Both doc_id and Invoice_id are exact duplicates of one another. Subtracting clear_date from due_in_date yields the delay date. Servlets facilitate the connection between our database and our user interface. The servlets created are for data retrieval, record addition, record editing, record deletion, record searching, and correspondence viewing in various templates.

4.2 Discussion of Project:

Considering the findings, the Invoice Management Application appears to be a more efficient method of managing invoices than manually reviewing each customer's invoice. The system considers thousands of invoices and then does regression analysis on them.

Chapter 5

5.1 CONCLUSION:

Experiences from various industries show that companies may have problems collecting customer invoice payments. Our aim was to understand customer behavior regarding invoice payments, and propose an analytical approach to learning and predicting payment behavior. Our logic can then be embedded into a system where decision makers can make predictions regarding future payments, and take actions as necessary toward the collection of potentially unpaid debt, or adjusting their financial plans based on the expected invoice-to-cash amount.

It is critical to identify the system requirements by appropriately gathering required data to interact with the system in order for the software to perform successfully. We gained a lot of practical information from this project, which we believe will help us to be in a better position in the future.

The project helped me to build a complete understanding of a working full-stack application. Also by integrating a machine learning model to my project, I was able to enhance the capabilities of my project and make it as industrial as possible

5.2 FUTURE SCOPE:

Organizations, firms, and even individuals may keep track of the payment of their orders with the aid of AI ENABLED FINTECH B2B INVOICE MANAGEMENT APPLICATION. This will not only save time, but will also provide a functional environment for an organization's and/or workplace's proper functioning.

The accuracy of the model can be increased as well and the same can be deployed in cloud to make it available to a larger audience.

References

- [1] <https://www.python.org/>
- [2] <https://www.geeksforgeeks.org/>
- [3] <https://reactjs.org/>
- [4] <https://mui.com/>
- [5] <https://www.w3schools.com/java/>