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 and Engineering

BY

Arun Chaudhary

1905025

UNDER THE GUIDANCE OF

Harsh Mangla and Nikhil Kumar

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 MANAGEMENT APPLICATION

submitted by

Arun Chaudhary 1905025

is a record of bonafide work carried out by them, 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: 15/04/2022

Harsh Mangla
Nikhil Kumar
(House of Black-Panther)

Project Mentor

Acknowledgements

I am profoundly grateful to **Harsha Mangla** and **Nikhil Kumar** of **Affiliation** for his expert guidance and continuous encouragement throughout during the whole of the project. . I would more like to praise and express my kind appreciation towards HighRadius for admitting me for this project .I express my gratitude's towards KIIT University,Deemed to be University for giving me an opportunity to work on projects with companies and always supporting me at every step of the way.I feel thankful to the college staff for giving me such a big opportunity.

Arun Chaudhary
1905025

ABSTRACT

The B2B world operates differently from the B2C or C2C world. Businesses work with other businesses on credit. When a buyer business orders goods from the seller business, the seller business issues an order invoice for the same. This invoice for the goods containsvarious information like the details of the goods purchased and when it should be paid.

Poor payment habits, in which consumers fail to pay on time, are one of the most serious issues that businesses confront today. This has a significant impact on businesses, particularly small and medium-sized enterprises (SMEs) that confront cash flow constraints. Businesses have faced serious penalties as a result of late payments, and in some cases, have been forced to close their doors. In fact, many companies are forced to use instruments like invoice financing to help them deal with late payment issues.

Keywords: Machine Learning,ReactJS,MySQL, Servlets,Flask Framework,Java.

Contents

List of Figures

Chapter 1

Introduction

The B2B operated differently from B2C or C2C world.A computer software package that allows firms to manage their orders and inventory is known as an invoice management system. By automatically entering new inventory into the system, tracking sales across various selling platforms such as eBay and Amazon, and alerting you, the business owner, when your stock of a particular item drops low enough to facilitate a reorder, invoice management systems aid in more accurate inventory management.

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 organisations of all sizes, both B2B and B2C.

Seller business interacts with various businesses and sells goods to all of them at various times. Hence, the seller business needs to keep track of the total amount it owes from all the buyers. This involves keeping track of all invoices from all the buyers. Each invoice will have various important fields like a payment due date, invoice date, invoice amount, baseline date etc.

The buyer business needs to clear its amount due before the due date. However, in real-world scenarios , the invoices are not always cleared i.e. paid in full amount by the due date. The date on which a customer clears the payment for an invoice is called the payment date.

Account Receivables Department:

1. In the ideal world, the buyer business should payback within the stipulated time (i.e. the Payment Term). However, in the real world, the buyer business seldom pays within their established time frame, and this is where the Account receivables Department comes into picture.
2. Every business consists of a dedicated Account receivables Department to collect and track payment of invoices.
3. It consists of a Account receivables team that is responsible for:

- 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 supplies.

Chapter 2

Technologies Used :

1) Machine Learning using Python
2) Java,SQL(Back-end)
3) HTML,CSS,JavaScript,React(Front-end)
4) Flask

2.1 Technology Used

Python

Python is a high-level, general-purpose programming language that is interpreted. Python's design philosophy prioritizes code readability, as evidenced by its extensive use of indentation. Its language elements and object-oriented approach are aimed at assisting programmers in writing clear, logical code for both small and large-scale projects.

Features in Python

There are many features in Python, some of which are:

● **GUI Programming Support**: Python modules such as PyQt5, PyQt4, wxPython, and Tk can be used to create graphical user interfaces. PyQt5 is the most popular Python graphical application framework.

● **Python is Portable language**: Python is also a portable programming language. For instance, if we have python code for Windows and want to run it on other platforms like Linux, Unix, or Mac, we don't need to update it; we can execute it on any platform.

● **Interpreted Language**: Python is an Interpreted Language, which means that its code is executed line by line. Python code does not require compilation, unlike other languages such as C, C++, Java, and others, making it easier to debug. Python's source code is transformed into bytecode, which is an immediate representation of the code.

● **Large Standard Library**: Python offers a big standard library that includes a huge number of modules and methods, allowing you to avoid writing your own code for everything. Regular expressions, unit testing, web browsers, and other packages are all available in Python.

● **Dynamically Typed Language**: Python is a dynamically-typed language. That is, the type of a variable (for example, int, double, long, etc.) is determined at run time rather than in advance, thus we don't have to define the type.

MACHINE LEARNING

Machine Learning is a field of artificial intelligence built on the principle that computers can learn from data, recognize patterns, and make judgments with little or no human input. Machine learning now is not the same as machine learning in the past, thanks to advances in computer technology. It was inspired by pattern recognition and the idea that computers may learn without being taught to execute certain tasks; artificial intelligence researchers sought to investigate if computers could learn from data. The iterative feature of machine learning is crucial because models may evolve autonomously as they are exposed to fresh data. They use past computations to provide consistent, repeatable judgments and outcomes. It's a science that's not new, but it's gaining new traction. While many machine learning techniques have been known for a while, the capacity to apply difficult mathematical computations to large amounts of data automatically – again and again, quicker and quicker – is a relatively new phenomenon. Here a few well-known examples of machine learning applications to get you started :

- The much-hyped Google self-driving car? Machine learning at its most basic level.
- Offers from Amazon and Netflix for online recommendations?Applications of machine learning in everyday life.
- What do your consumers have to say about you on Twitter? Combining machine learning with the construction of language rules.
- Is it possible to identify fraud? One of the more apparent and crucial applications in today's environment.

Regression Analysis

In statistical modeling, regression analysis is a set of statistical processes for estimating the relationships between a dependent variable and one or more independent variables. The most common form of regression analysis is linear regression, in which one finds the line that most closely fits the data according to a specific mathematical criterion. For example, the method of ordinary least squares computes the unique line that minimizes the sum of squared differences between the true data and that line. For specific mathematical reasons, this allows the researcher to estimate the conditional expectation of the dependent variable when the independent variables take on a given set of values. Less common forms of regression use slightly different procedures to estimate alternative location parameters or estimate the conditional expectation across a broader collection of non-linear models.

Regression analysis is primarily used for two conceptually distinct purposes. First, regression analysis is widely used for prediction and forecasting, where its use has substantial overlap with the field of machine learning. Second, in some situations regression analysis can be used to infer causal relationships between the independent and dependent variables. Importantly, regressions by themselves only reveal relationships between a dependent variable and a collection of independent variables in a fixed dataset.

To use regressions for prediction or to infer causal relationships, respectively, a researcher must carefully justify why existing relationships have predictive power for a new context or why a relationship between two variables has a causal interpretation. The latter is especially important when researchers hope to estimate causal relationships using observational data.

Various regression techniques were used to find out which technique best fit our model. Some of them are

Linear Regression

In machine learning, linear regression is one of the most fundamental forms of regression. A predictor variable and a dependent variable are connected linearly in the linear regression model. When there are numerous independent variables in the data, linear regression is referred to as multiple linear regression models. The linear regression model is represented by the equation below:
y=mx+c+e
where m denotes the line's slope, c denotes an intercept, and e is the model's error.

Changing the values of m and c yields the best fit line. The difference between the observed and expected values is the prediction error. The values of m and c are chosen such that the prediction error is as low as possible. It's vital to remember that outliers might occur in a simple linear regression model. As a result, it should not be utilized when dealing with large amounts of data.

Support Vector Regression

Both linear and non-linear models may be solved using support vector regression. For non-linear models, SVM employs non-linear kernel functions (such as polynomial) to identify the best solution.SVR's major goal is to reduce error by customizing the hyperplane to maximize the margin.

Random Forest Regression

Random forests, also known as random decision forests, are an ensemble learning approach for classification, regression, and other tasks that works by training a large number of decision trees and then outputting the class that is the mode of the classes (classification) or the mean/average prediction (regression) of the individual trees. Random decision forests address the problem of decision trees overfitting their training set. Random forests outperform decision trees in most cases, though they are less accurate than gradient enhanced tree.
Data features, on the other hand, might have an impact on their performance.

Decision Tree Regression

Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision.

Integration of ML using Python

While building this model we had gone through many process:
Step 1. Firstly we need to import the dataset and also import various data manipulation libraries.
Step 2 . We visualized the dataset in the form of a data frame to get a brief idea about the dataset.
Step 3. Identified the target variable which is clear_date.
Step 4. Independent and dependent variables identification and extraction. Step 5. Handling Missing Values using Null Imputation Techniques.
Step 6. Encoding Categorical Variables to training purposes.
Step 7. Splitting the dataset for train/test and validation.
Step 8. Feature Scaling for improved training using Normalization and Standardization.
Step 9. Training and Validating ML model.
Step 10. Predict clear_date using test data.
Step 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

REACT

React is a JavaScript library for building interactive web or mobile applications. It is an open source and front-end based library for building user interface or UI components maintained primarily by Facebook. Some features of React are that it is-

● **Declarative** - This is a reason for the developer's ease and simplicity in regard to the output that is viewed. The UI is updated as the developer changes the code and simultaneously renders it to the output screen. Due to this, it becomes easier to design as well as debug the code. The developer can change the code and components to see how the UI will look at that particular time.This feature results in painless development and makes the code more predictable.In simpler terms, declarative means "telling what to do instead of how to do it".

Let us take an example: If we want a button at any position in the application, we can simply instruct it to render a button there with the help of the code, React will make it happen without us specifying details as to 'how to render.

- **Component-based** - React consists of specific elements in the application which have different functionalities like- a header, side panel, footer, etc., these are called components. These components represent classes or functions that accept input and render different HTML components to the UI accordingly.

Components in React are of two types-
1. Class-based components 2. Function-based components

- **JSX** - JSX stands for JavaScript XML. It is a JavaScript syntax extension wherein the developer can write HTML codes in JavaScript. React processes this extended syntax into JavaScript calls. This enables the functionalities of HTML in the application to be used with JavaScript. This syntax extension makes the code easier to write and debug as it.

Implementation of React in the project-

React was used to build the user interface of the web application.
Two main components of the application were-
1. Header 2. Grid Panel section

Section 1: Header

The header consists of the following elements:
i. Account name logo on the left,
ii. ii. In the middle, the HighRadius logo.

Section 2: Grid Panel

The segment of the Grid panel will be divided into four parts:

- A Predict button will be in the top left corner of the grid's header, followed by a View Correspondence Button, an Add Button, an Edit Button, a Delete Button, and a Search Bar.
- In the top left hand corner of the row, the name of the grid, Invoice List, will be stated.
- The table with customer invoice data as rows and the following columns is the second part.

The columns are-
1. Checkbox
2. Customer Name
3. Customer Number (Customer #)
4. Invoice Number (Invoice #)
5. Invoice Amount
6. Due Date
7. Predicted Payment Date
8. Predicted Aging Bucket
9. Notes

Functionalities -
1. Add button
2. Edit button
3. Delete
4. View correspondence
5. Predict
6. Search bar

Every client-side application needs data from the server to view in the user interface. To communicate with our server and perform data transactions, we make API calls to it. This is something also done by React. Callbacks, Promises, and async await all aid us in this endeavour.They play the same function, but they are syntactic sugar.

Concept

1.**HTTP Methods :-**

a. GET request: The GET method is used to get a representation of a resource. Requests made with the GET method should only return data.

b. Post request: The POST method is used to submit an object to the defined resource, which often causes the server's state to alter or has side effects.

c. PUT request: All current representations of the target resource are replaced with the request payload when using the PUT process.

d. DELETE request: The DELETE method deletes the specified resources.

2. Callbacks :-

Callbacks are functions that are transferred to another function as an argument. once an incident has occurred or a specific mission has been completed, In asynchronous code, it's frequently used. Later on, callback functions are named. a piece of code that can be declared on initialization but does not have to be executed invocation. 3. Promises -: A promise is an entity that has the potential to generate a single value in the future: either a resolved value or a reason why it hasn't been resolved yet (e.g., a network error occurred). A promise may be fulfilled, refused, or pending in one of three ways. Users may use callbacks to manage the fulfilled value or the reason for rejection when using Promise.

A promise has 3 stages-

- Fulfilled- if the request is fulfilled
- Rejected- if the request is rejected
- Pending- the request is not yet fulfilled

These were the functions that were used to make the API request to get required data from the database.

JAVA-

A specification defines Java, which includes a programming language, a compiler, core libraries, and a runtime (Java virtual machine) The Java runtime enables software developers to write programme code in languages other than Java, which is also supported by the Java virtual machine. The Java virtual machine and the Java core libraries are commonly associated with the Java platform.

The backend work is done using servlets which are implemented in Java.

Servlets:

It is a technology that resides at the server side and is responsible for generating dynamic web pages is known as a servlet, in order words it helps create web applications.

Since it is written in Java this makes it robust and scalable technology. Prior to servlets, Common Gateway Interface (CGI) scripting language was common as a server-side programming language. But due to its many drawbacks and flaws as discussed below, we do not use CGI anymore.

There are many interfaces and classes in the Servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, etc.

A servlet is a Java class which is used to extend the capabilities of servers that host applications accessed by means of a request-response model. Servlets are mainly used to extend the applications hosted by webs servers, however, they can respond to other types of requests too. For such applications, HTTP- specific servlet classes are defined by Java Servlet technology.

A servlet is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model. Servlets are commonly used to extend the applications hosted by web servers.

Servlet Life Cycle

The Servlet life cycle mainly includes the following four stages,

- Loading a Servlet
- Initializing the Servlet
- Request handling
- Destroying the Servlet

JSP:

A JSP is a text document which contains two types of text: static data and dynamic data. The static data can be expressed in any text-based format (like HTML, XML, SVG and WML), and the dynamic content can be expressed by JSP elements.

JavaServer Pages often serve the same purpose as programs implemented using the Common Gateway Interface (CGI). But JSP offers several advantages in comparison with the CGI.

Performance is significantly better because JSP allows embedding Dynamic Elements in HTML Pages itself instead of having separate CGI files.

JSP are always compiled before they are processed by the server unlike CGI/Perl which requires the server to load an interpreter and the target script each time the page is requested.

JavaServer Pages are built on top of the Java Servlets API, so like Servlets, JSP also has access to all the powerful Enterprise Java APIs, including JDBC, JNDI, EJB, JAXP, etc. JSP pages can be used in combination with servlets that handle the business logic, the model supported by Java servlet template engines.

JavaServer Pages (JSP) is a technology used for developing Web Pages that supports dynamic content. This allows developers to insert java code in HTML pages.

STEPS TO FOLLOW TO CREATE A JSP PROGRAM:

- Create a html page from where request will be sent to server eg try.html.

- To handle the request of the user next is to create .jsp file Eg.new.jsp

- Create project folder structure.

- Add Tomcat server to your Workspace in eclipse IDE / Create a WAR file

- Start the Tomcat Server

Chapter 3

Problem Statement / Requirement Specifications

3.1 PROJECT PLANNING

Build an AI-Enabled FinTech B2B Invoice Management Application to find whether bill payment will be delayed or not.

"Accounts Receivable represents money owed by entities to the firm on the sale of products or services on credit. In most business entities, accounts receivable is typically executed by generating an invoice and either mailing or electronically delivering it to the customer, who, in turn, must pay it within an established time frame, called credit terms or payment terms."

Machine Learning model to predict the date of payment of an invoice:

- To build a Machine Learning Model to predict the payment date of an invoice when it gets created in the system.
- Categorize the invoice into different buckets based on predicted.

Problem statement for Application Development:

- To build a full stack Sales Management Application using ReactJs, JDBC, Java and JSP
- Build a responsive Receivables Dashboard.
- Visualize Data in the form of grids.
- Perform Searching operations on the invoices.
- Edit data in the editable fields of the grid.
- Download data of selected rows in predefined templates.

3.2 PROJECT ANALYSIS / SYSTEM DESIGN

REQUIREMENT

The backend of the UI is primarily built using Java and JSP, and the database used is a MySQL database hosted using SQLyog community. We also make use of Tomcat, made by the Apache Software Foundation to be able to render the JSP pages which make up most of the front end of the product. To summarize, you would need the following minimum software components in your system to be able to use the product :

- Anaconda (Python 3.7) : To pre process the dataset and build the prediction model.

- Java Development Kit (JDK 1.8): To process data from the MySQL database using Java.

- Visual Studio Code (editor) : To code the frontend using HTML, CSS, REACT JS.

- Tomcat8/9: To render JSP webpages.

- Web Browser: Google Chrome recommended, to access the GUI.

High Level Requirements of Application

Specifically, below are the major aspects of the application that needs to be developed.

1. Data Loading in DB:

a. You will be provided with a invoices dataset which you need to parse, process and load in the provided database schemas.

2. UI Representation of the data:

a. Build a responsive UI which can display the invoice data loaded from the database.

b. The UI should support searching and infinite scrolling operations.

c. The UI should support editing of some editable fields, adding a new row to the grid, deleting rows from the grid and downloading selected records from the grid in predefined template(s)

TOOLS USED

The tools and programming languages used for the implementation of this project are described are-

1. **PYTHON :** Python is an interpreted high-level general purpose programming language. To build the machine learning prediction model we used the following libraries- Pandas, Numpy, Seaborn, Scikit-learn.

2 HTML : Hypertext Markup Language is the fundamental markup language that helped us build the website for this project. It allowed us to create and structure sections for the web page. Being a markup language, HTML does not provide dynamic functionality instead its salient features are its ability to format and organize documents.

3. CSS : Cascading Style Sheets is a design language that allows users to present web pages in a beautiful manner using different colors and styles for texts, backgrounds, headlines, and paragraphs.

4. JavaScript : JavaScript, being a dynamic computer programming language, allows interaction between client-side script and user which makes the webpage dynamic. JavaScript also has object-oriented programming capabilities. When used for the frontend, Javascript provides a platform to make the webpage more interactive such as image carousels and converting menus to dropdowns, and making any page responsive.

5. JSP: Java Server Pages belong to Java web technologies and are server-side technologies that help in creating dynamic and platform-independent web pages. Using this, Java code can be inserted into HTML.

6. JDK: The Java Development Kit (JDK) is a software development environment used for developing Java applications and applets. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

7. JRE: The Java Runtime Environment, or JRE, is a software layer that runs on top of a computer's operating system software and provides the class libraries and other resources that a specific Java program needs to run.

8. Apache Tomcat: Apache Tomcat is and is an open-source software where Java Servlet, JavaServer Pages, Java Expression Language, and WebSocket technologies can be implemented. Tomcat provided us a pure Java system environment where Java code could be run.

9. SQLyog: SQLyog Ultimate is the most powerful manager, admin and GUI tool for MySQL, combining the features of MySQL Query Browser and MySQL GUI tools in a single intuitive interface. SQLyog is a fast, easy to use and compact graphical tool for managing your MySQL databases.

10. Eclipse: Eclipse is an integrated development environment (IDE) used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications.

11.Visual Studio Code: Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python and C++.Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse.

11. JSX : JSX stands for JavaScript XML. It is a JavaScript syntax extension wherein the developer can write HTML codes in JavaScript. React processes this extended syntax into JavaScript calls. This enables the functionalities of HTML in the application to be used with JavaScript. This syntax extension makes the code easier to write and debug as it.

12. Java : Java is a high-level, class based, object oriented programming language that is designed to have as few implementation dependencies as possible. The back-end work isdone using servlets which are implemented in Java.

Chapter 4

Implementation

4.1 Methodology OR Proposal

About Dataset

For our project, I used an invoices dataset that contains the past payment information and behaviour of various buyers. The dataset was in csv format and the dataset contained the following columns :-

- - Company ID
- - Document Number
- - Business Code
- - Create Year
- - Document type
- - Customer Number
- - Customer Name
- - Document Create Date
- - Posting date
- - Due In Date
- - Order Create Date
- - Invoice ID
- - Baseline Create Date
- - Total Open Amount

Feature Engineering

All the features are categorized in parts, then all the features which comes under the category will be sequenced and followed by the reason of using it. Now we have to represent the categories for which categorical columns need to be created. We use three methods to create categorical column :-

1. Label Encoding: Map each category/ unique value to a numeric value such as A:1,B:2,C:3.

2.One hot encoding: Here, we map each category to a vector that contains 1 and 0 denoting the presence of the feature or not. The number of vectors depends on the categories which we have in our dataset. For high cardinality features, this method produces a lot of columns that slows down the learning of the model significantly.

3. Target Encoding: Here, features are replaced with a blend of the posterior probability of the target for the given particular categorical value and the prior probability of the target over all the training data. Also, they are not generated For the test data. We usually save the target encodings obtained from the training data set and use the same encodings to encode features in the test data set.

Feature selection

Feature selection is one of the most important processes in machine learning algorithms. It uses a greedy approach and evaluates all possible combinations of features to give us the best score. There are different approaches to go for feature selection i.e manually, wrapper method and embedded method.

A correlation matrix was used to find the best possible features manually, and business_code, cust_number, cust_payment_terms and the difference between due_in_date and the baseline_create_date were chosen as the appropriate features.

After that the model was evaluated. Three main metrics are used for model evaluation :-
1. Accuracy

2. Precision

3. Recall

For the test set feature calculations we predict one prediction at a time and then that prediction was used to calculate the delay

Modelling

Decision Tree, Linear Regression, Support vector machines(SVMs) and Random Forest Regressor were used with different feature sets and the models were evaluated on the basis of R2_score and RMSE_score. After the evaluation the models were used to predict when the payments were cleared by the customers.

Objective of Java and React :

Concept Details

1. HTTP Methods are as following :

• GET request: The GET method is used to get a representation of a resource. Requests made with the GET method should only return data.

• Post request: The POST method is used to submit an object to the defined resource, which often causes the server's state to alter or has side effects.

• PUT request: All current representations of the target resource are replaced with the request payload when using the PUT process.

• DELETE request: The DELETE method deletes the specified resources.

2. Callbacks :-

Callbacks are functions that are transferred to another function as an argument. Once an incident has occurred or a specific mission has been completed. In asynchronous code, it's frequently used. Later on, callback functions are named.

3. Promises -:A promise is an entity that has the potential to generate a single value in the future: either a resolved value or a reason why it hasn't been resolved yet (e.g., a network error occurred). A promise may be fulfilled, refused, or pending in one of three ways. Users may use callbacks to manage the fulfilled value or the reason for rejection when using Promise.

A promise has 3 stages-

● Fulfilled- if the request is fulfilled
● Rejected- if the request is rejected
● Pending- the request is not yet fulfilled

These were the functions that were used to make the API request to get required data from the database.

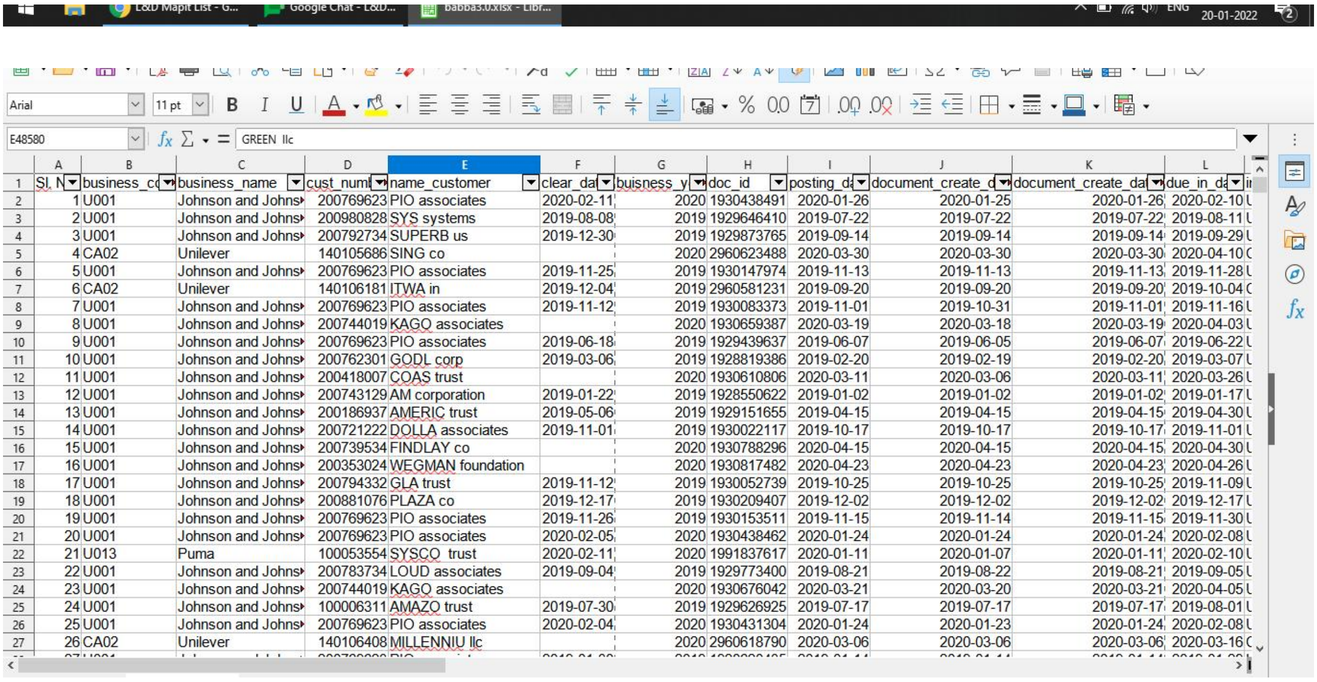
Backend :

Data Loading in the Database:

Step 1: Execute the SQL script for the creation of table.

Step 2: Read the csv datasheet using a CSV reader and stored information.

Step 3: We used a JDBC driver and also made a POJO class which helped us load the datasheet into the database in batches.

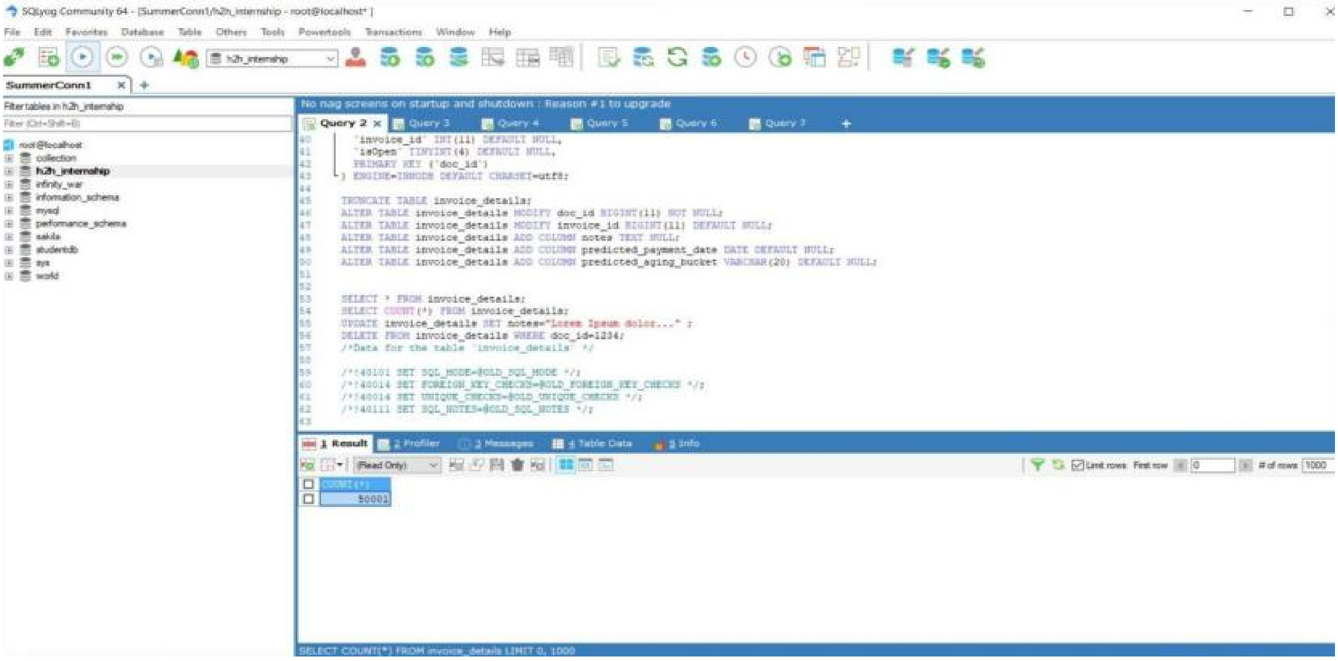


E48580

GREEN llc

	A	B	C	D	E	F	G	H	I	J	K	L	
1	SI. N	business_co	business_name	cust_numb	name_customer	clear_dat	buisness_y	doc_id	posting_da	document_create_d	document_create_dat	due_in_da	i
2	1	U001	Johnson and Johns	200769623	PIO associates	2020-02-11	2020	1930438491	2020-01-26	2020-01-25	2020-01-26	2020-02-10	U
3	2	U001	Johnson and Johns	200980828	SYS systems	2019-08-08	2019	1929646410	2019-07-22	2019-07-22	2019-07-22	2019-08-11	U
4	3	U001	Johnson and Johns	200792734	SUPERB us	2019-12-30	2019	1929873765	2019-09-14	2019-09-14	2019-09-14	2019-09-29	U
5	4	CA02	Unilever	140105686	SING co		2020	2960623488	2020-03-30	2020-03-30	2020-03-30	2020-04-10	C
6	5	U001	Johnson and Johns	200769623	PIO associates	2019-11-25	2019	1930147974	2019-11-13	2019-11-13	2019-11-13	2019-11-28	U
7	6	CA02	Unilever	140106181	ITWA in	2019-12-04	2019	2960581231	2019-09-20	2019-09-20	2019-09-20	2019-10-04	C
8	7	U001	Johnson and Johns	200769623	PIO associates	2019-11-12	2019	1930083373	2019-11-01	2019-10-31	2019-11-01	2019-11-16	U
9	8	U001	Johnson and Johns	200744019	KAGO associates		2020	1930659387	2020-03-19	2020-03-18	2020-03-19	2020-04-03	U
10	9	U001	Johnson and Johns	200769623	PIO associates	2019-06-18	2019	1929439637	2019-06-07	2019-06-05	2019-06-07	2019-06-22	U
11	10	U001	Johnson and Johns	200762301	GODL corp	2019-03-06	2019	1928819386	2019-02-20	2019-02-19	2019-02-20	2019-03-07	U
12	11	U001	Johnson and Johns	200418007	COAS trust		2020	1930610806	2020-03-11	2020-03-06	2020-03-11	2020-03-26	U
13	12	U001	Johnson and Johns	200743129	AM corporation	2019-01-22	2019	1928550622	2019-01-02	2019-01-02	2019-01-02	2019-01-17	U
14	13	U001	Johnson and Johns	200186937	AMERIC trust	2019-05-06	2019	1929151655	2019-04-15	2019-04-15	2019-04-15	2019-04-30	U
15	14	U001	Johnson and Johns	200721222	DOLLA associates	2019-11-01	2019	1930022117	2019-10-17	2019-10-17	2019-10-17	2019-11-01	U
16	15	U001	Johnson and Johns	200739534	FINDLAY co		2020	1930788296	2020-04-15	2020-04-15	2020-04-15	2020-04-30	U
17	16	U001	Johnson and Johns	200353024	WEGMAN foundation		2020	1930817482	2020-04-23	2020-04-23	2020-04-23	2020-04-26	U
18	17	U001	Johnson and Johns	200794332	GLA trust	2019-11-12	2019	1930052739	2019-10-25	2019-10-25	2019-10-25	2019-11-09	U
19	18	U001	Johnson and Johns	200881076	PLAZA co	2019-12-17	2019	1930209407	2019-12-02	2019-12-02	2019-12-02	2019-12-17	U
20	19	U001	Johnson and Johns	200769623	PIO associates	2019-11-26	2019	1930153511	2019-11-15	2019-11-14	2019-11-15	2019-11-30	U
21	20	U001	Johnson and Johns	200769623	PIO associates	2020-02-05	2020	1930438462	2020-01-24	2020-01-24	2020-01-24	2020-02-08	U
22	21	U013	Puma	100053554	SYSCO trust	2020-02-11	2020	1991837617	2020-01-11	2020-01-07	2020-01-11	2020-02-10	U
23	22	U001	Johnson and Johns	200783734	LOUD associates	2019-09-04	2019	1929773400	2019-08-21	2019-08-22	2019-08-21	2019-09-05	U
24	23	U001	Johnson and Johns	200744019	KAGO associates		2020	1930676042	2020-03-21	2020-03-20	2020-03-21	2020-04-05	U
25	24	U001	Johnson and Johns	100006311	AMAZO trust	2019-07-30	2019	1929626925	2019-07-17	2019-07-17	2019-07-17	2019-08-01	U
26	25	U001	Johnson and Johns	200769623	PIO associates	2020-02-04	2020	1930431304	2020-01-24	2020-01-23	2020-01-24	2020-02-08	U
27	26	CA02	Unilever	140106408	MILLENNIU llc		2020	2960618790	2020-03-06	2020-03-06	2020-03-06	2020-03-16	C

Fig 4.1: CSV file



```
  `invoice_id` INT(11) DEFAULT NULL,
  `isOpen` TINYINT(4) DEFAULT NULL,
  PRIMARY KEY (`doc_id`)
) ENGINE=INNODB DEFAULT CHARSET=utf8;

TRUNCATE TABLE invoice_details;
ALTER TABLE invoice_details MODIFY doc_id BIGINT(11) NOT NULL;
ALTER TABLE invoice_details MODIFY invoice_id BIGINT(11) DEFAULT NULL;
ALTER TABLE invoice_details ADD COLUMN notes TEXT NULL;
ALTER TABLE invoice_details ADD COLUMN predicted_payment_date DATE DEFAULT NULL;
ALTER TABLE invoice_details ADD COLUMN predicted_aging_bucket VARCHAR(20) DEFAULT NULL;


SELECT * FROM invoice_details;
SELECT COUNT(*) FROM invoice_details;
UPDATE invoice_details SET notes="Lorem Ipsum dolor..." ;
DELETE FROM invoice_details WHERE doc_id=1234;
/*Data for the table `invoice_details` */

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
```

□	COUNT(*)
□	50001

Limit rows: 0, # of rows: 1000

SELECT COUNT(*) FROM invoice_details LIMIT 0, 1000

Fig 4.2: Data Inserted In SQLYog

Servlets:

It is a technology that resides at the server side and is responsible for generating dynamic web pages is known as a servlet, in order words it helps create web applications.

There are many interfaces and classes in the Servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, etc.

A servlet is used to extend the capabilities of servers that host applications accessed by means of a request-response programming model.Servlets are mainly used to extend the applications hosted by webs servers, however, they can respond to other types of requests too. For such applications, HTTP- specific servlet classes are defined by Java Servlet technology.

Servlet Life Cycle

The Servlet life cycle mainly includes the following four stages,

- Loading a Servlet
- Initializing the Servlet
- Request handling.
- Destroying the Servlet

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.

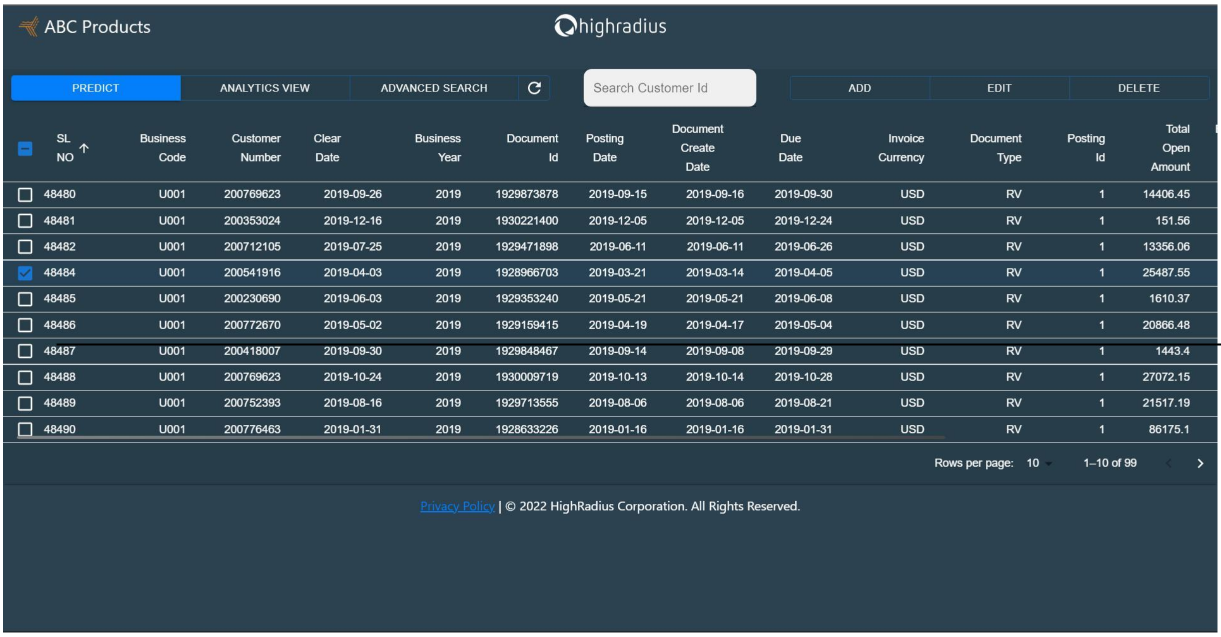
4) Search Servlet - Get the invoice number from the frontend and pass them as a http request using axios to the backend and search through the database and return it to the frontend again.

5) Data Display Servlet - Display the table of invoices to the frontend.

Frontend:

UI Representation of the Data:

Receivables Dashboard Page



▬	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
☐	48480	U001	200769623	2019-09-26	2019	1929873878	2019-09-15	2019-09-16	2019-09-30	USD	RV	1	14406.45
☐	48481	U001	200353024	2019-12-16	2019	1930221400	2019-12-05	2019-12-05	2019-12-24	USD	RV	1	151.56
☐	48482	U001	200712105	2019-07-25	2019	1929471898	2019-06-11	2019-06-11	2019-06-26	USD	RV	1	13356.06
☑	48484	U001	200541916	2019-04-03	2019	1928966703	2019-03-21	2019-03-14	2019-04-05	USD	RV	1	25487.55
☐	48485	U001	200230690	2019-06-03	2019	1929353240	2019-05-21	2019-05-21	2019-06-08	USD	RV	1	1610.37
☐	48486	U001	200772670	2019-05-02	2019	1929159415	2019-04-19	2019-04-17	2019-05-04	USD	RV	1	20866.48
☐	48487	U001	200418007	2019-09-30	2019	1929848467	2019-09-14	2019-09-08	2019-09-29	USD	RV	1	1443.4
☐	48488	U001	200769623	2019-10-24	2019	1930009719	2019-10-13	2019-10-14	2019-10-28	USD	RV	1	27072.15
☐	48489	U001	200752393	2019-08-16	2019	1929713555	2019-08-06	2019-08-06	2019-08-21	USD	RV	1	21517.19
☐	48490	U001	200776463	2019-01-31	2019	1928633226	2019-01-16	2019-01-16	2019-01-31	USD	RV	1	86175.1

Rows per page: 10 1–10 of 99

Privacy Policy | © 2022 HighRadius Corporation. All Rights Reserved.

Fig 4.3: Receivables Dashboard Page

It consists of 2 sections:
1. Header

2. Grid Panel Section

1. **Header Section**

The header consists of:

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

2. Grid Panel Section

The Grid panel section will be divided into 4 portions:

- 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.

- The name of the grid that is Invoice List will be mentioned in the top left corner of the grid.

- The second portion is the table with customer invoice data as rows and the columns hello bro!!

List of all the columns to be represented on the UI are as follows:
1. Checkbox
2. Customer Name
3. Customer Number (Customer #)
4. Invoice Number (Invoice #)

5.Invoice Amount
6. Due Date
7. Predicted Payment Date
8. Predicted Aging Bucket
9.Notes

Functionalities implemented in React in Detail:

Add Button Functionality - Adds an Invoice to the existing database

Edit Button Functionality- Edits an invoice in the database

Delete Button Functionality - Deletes an invoice in the database

Analytics View - Helps to view chart of details according to delay date.

Predict Button Functionality - Predict the clearing date of an invoice. So after clicking on the predict button it will populate the Predicted Payment Date and Predicted Aging Bucket.

Search Bar Functionality - Search an invoice by the invoice number.

Infinite Scrolling Functionality - As we know that loading 50000 data at same time could cause problems in the UI. To remove that problem we used the concept of Infinite Scrolling where the data would be loaded in batches.

Problems faced while doing the project:

1) In the machine learning model it was tough to choose or make the features which would increase the accuracy of the model.

2) Implementing Redux in React was tough.

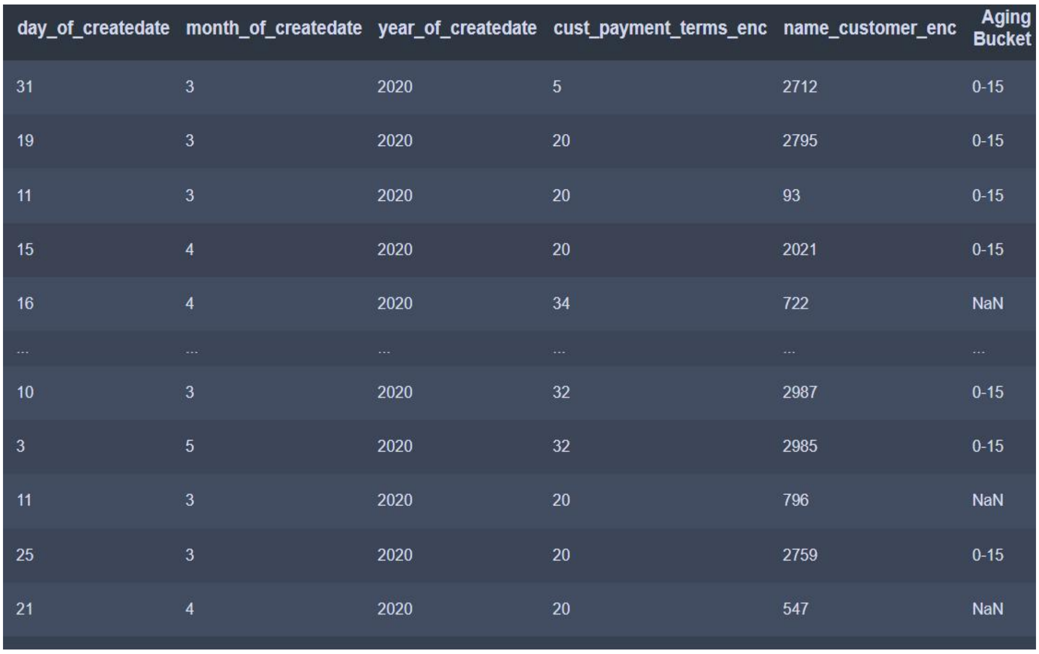
3) Faced much difficulty while implementing the various functionalities in the UI .

4) Infinite Loading was quite challenging. UI design was also difficult to implement since I am very new to JavaScript. Logical interfacing of the functionalities was the most challenging work.

4.3 Result Analysis and Screenshots

After the completion of the project, I was able to build the predictive model as well as the B2B application for the use of the company.
For the machine learning model, I was able to acquire an R2 score of 0.69 for XG Boost Regressor. On comparing it with the other regression models, it was found that the XG Boost Regressor performed the best among all of them. Hence the XG Boost Regression model was applied to the validation set and the result which was obtained was found to be 0.69 for the R2 score.

Finally, the features were mapped onto the final test set and were fed to the predictive model. A final dataframe with the predicted dates and the aging buckets was created. A snippet of which is shown below.



day_of_createdate	month_of_createdate	year_of_createdate	cust_payment_terms_enc	name_customer_enc	Aging Bucket
31	3	2020	5	2712	0-15
19	3	2020	20	2795	0-15
11	3	2020	20	93	0-15
15	4	2020	20	2021	0-15
16	4	2020	34	722	NaN
...	...	...	...	...	...
10	3	2020	32	2987	0-15
3	5	2020	32	2985	0-15
11	3	2020	20	796	NaN
25	3	2020	20	2759	0-15
21	4	2020	20	547	NaN

Fig 4.5: Final DataFrame

After the successful building of the model, it was required to create the full stack invoice management application using Java and React.

Chapter 5

Standards Adopted

5.1 Design/ Standards

In all the engineering streams, there are predefined design standards are present such as IEEE, ISO etc.

- Design and development stages
- Design and development inputs
- Design and development controls
- Design and development outputs
- Design and development changes

5.2 Coding Standards

Coding standards are collections of coding rules, guidelines, and best practices. Few of the coding standards are:

1. Write as few lines as possible.
2. Use appropriate naming conventions.
3. Segment blocks of code in the same section into paragraphs.
4. Use indentation to marks the beginning and end of control structures. Clearly specify the code between them.
5. Don’t use lengthy functions. Ideally, a single function should carry out a single task.

5.3 Testing/Verification Standards

There are some ISO and IEEE standards for quality assurance and testing of the product.

- Design complies with and is traceable to inputs
- Design complies with client requirements
- Design complies with standards, guides and codes
- Design complies with local and statutory requirements
- Constructability and maintainability review has been completed
- Design is economical and value-for-money

Chapter 6

Conclusion & Future Scope

6.1 Conclusion

A Sales Management system is any tool or platform that tracks sales, orders, inventory, and fulfillment as well as enables the people, processes, and partnerships necessary for products to find their way to the customers who bought them. In this project, I have built an AI-Enabled FinTech B2B Order Management Application using a regression model to predict whether payment of order will be delayed or not. The same has been deployed using Flask Framework, React JS (for frontend) and Java (for backend).

This Project dealt with the design and development process of context-adaptive Web-application. During the whole process, I was able to learn and implement many new technologies. I also got the knowledge of how Business to Business work operates. 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.

6.2 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, regardless of the big number of clients. This will not only save time, but will also provide a functional environment for an organization's and/or workplace's proper functioning.
We can work on making the application more user friendly and try to build new features for the machine learning model to improve the accuracy. We can also work on making the site responsive for every device or resolution.

References

[1] https://www.geeksforgeeks.org/data-preprocessing-machine-learning-python/

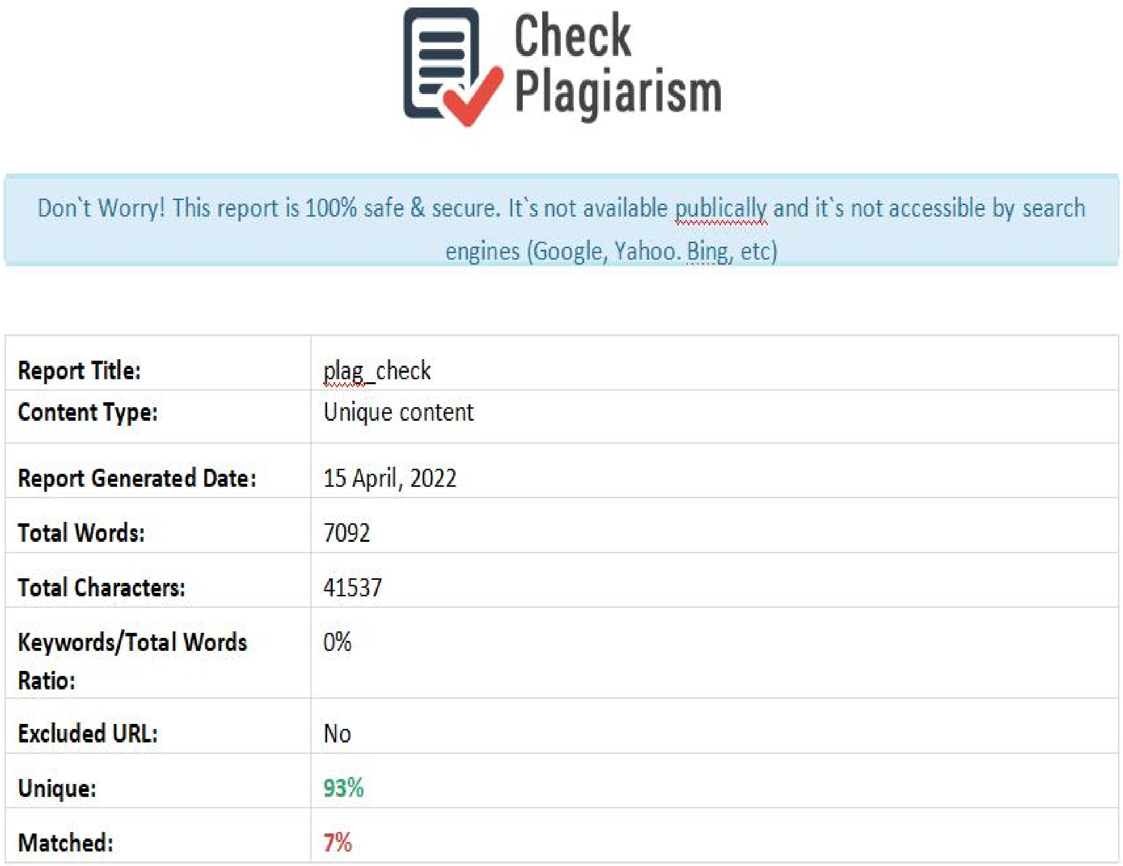
[2] https://reactjs.org/

[3] https://www.javatpoint.com/servlet-tutorial

[4] How to Choose a Feature Selection Method For Machine Learning (machinelearningmastery.com)

[5] Java Database Connectivity(JDBC) (knowledgehut.com)

PLAGIARISM REPORT



Check Plagiarism

Don`t Worry! This report is 100% safe & secure. It`s not available publically and it`s not accessible by search engines (Google, Yahoo. Bing, etc)

Report Title:	plag_check
Content Type:	Unique content
Report Generated Date:	15 April, 2022
Total Words:	7092
Total Characters:	41537
Keywords/Total Words Ratio:	0%
Excluded URL:	No
Unique:	**93%**
Matched:	**7%**