Objective of Highway to Highradius Internship Program

Build an AI-Enabled FinTech B2B Invoice Management Application

Business Overview

Introduction to B2B Operations:

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 invoice for the same**. This invoice for the goods contains various information like the details of the goods purchased and when it should be paid. This is known in accounting terminology as “Accounts Receivable”

> ***“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 timeframe, called credit terms or payment terms.”***

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 ie. 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 (ie 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 supplied.

Problem Statement for Application Development:

The objective of the HTHIP internship project is:

- Data Preprocessing of invoices dataset
- Feature Engineering on the dataset
- To build a **Machine Learning Model** to predict the payment date of an invoice when it gets created in the system
- To build **a full stack Invoice Management Application** using HTML, CSS, JS, Java and JSP.
- Build a **Receivables Dashboard.**
- **Visualize Data** in the form of grids.
- Perform **Searching** operations on the invoices.
- **Edit data** in the editable fields of the grid.

MILESTONES:

Machine Learning:

Milestone 1: Basics of Python & Data Preprocessing
Milestone 2: Exploratory Data Analysis, Data Manipulation & Feature Engg
Milestone 3: Feature Selection, Model Preparation & Prediction

Application:

Milestone 4: UI Creation
Milestone 5: SQL Basics, Java Basics, Servlet Basics, JDBC Connection
Milestone 6: Feature Functionality - Servlets
Optional Milestone : Search Functionality

HIGH LEVEL REQUIREMENTS OF APPLICATION

Specifically, below are the major aspects of the application that needs to be developed. The details for each of the below is provided in the functional overview section.

1. **UI Representation of the data:**
    a. Build a UI which can display the invoice data loaded from the database.
    b. The UI should support searching operations.
    c. The UI should support editing of some editable fields, adding a new row to the grid, deleting rows from the grid.
2. **Data Loading in DB:**
    a. You will be provided with an invoices **dataset** which you need to parse, process and load in the provided database schemas.

FUNCTIONAL OVERVIEW

Milestone 1: Data Preprocessing

HighRadius will be providing you an **invoices dataset** which you need to parse and process.

account_id	document_numb	company_code	fiscal_year	branch	customer_numbe	fk_customer_ma	document_date_	baseline_date_n	due_date_
60	20000950	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000950	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000950	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000950	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000870	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000870	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000870	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000870	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20001004	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20001004	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20001004	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20001004	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000831	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000831	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000831	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20000831	\N	\N	\N	999888	-1	05-04-16	\N	\N
60	20103019	\N	\N	\N	999888	-1	11-04-16	\N	\N
60	20103019	\N	\N	\N	999888	-1	11-04-16	\N	\N
60	20103019	\N	\N	\N	999888	-1	11-04-16	\N	\N
60	20103019	\N	\N	\N	999888	-1	11-04-16	\N	\N
60	20000972	\N	\N	\N	999888	-1	05-04-16	\N	\N

List of all the fields part of dataset are as follows:

- Acct Doc ID
- Company ID
- Document Number
- Document Number Norm
- Business Code

- Create Year
- Document type
- Customer Number
- Customer Number Norm
- Customer Map ID
- Customer Name
- Document Create Date
- Document Create Date Norm
- Posting date
- Posting date norm
- Posting ID
- Due In Date
- Due In Date Norm
- Order Create Date
- Order Create Date Norm
- Invoice ID
- Invoice ID Norm
- Baseline Create Date
- Invoice Date norm
- Total Open Amount
- Customer Payment Terms
- Shipping date
- Shipping to
- Clear Date
- Clear Date Norm
- Is Open Invoice
- Doc Id
- Actual Outstanding amount
- Invoice Amount
- Dispute Valid Status
- Dispute Amount

Refer to this link for detailed description of column headers.

Milestone 2: Feature Engineering

Objective of this milestone is to apply feature engineering techniques to extract patterns and identify features for efficient modelling.

Milestone 3: Developing ML Model

An Invoices dataset that contains the past payment information and behaviour of various buyers. Based on the previous payment patterns, the ML model needs to predict what will be the date a payment is made by the customer for an invoice.
The model also needs to predict which aging bucket the invoice falls into based on the predicted payment date.

For example:

Invoice No	Due Date	Actual Open Amt($)	Payment Date
1001	03/12/2020	9000	14/02/2021
1002	10/12/2020	5000	23/12/2020
1003	17/12/2020	3000	16/01/2021

In this example, the payment for the first invoice (1001) is due on 3rd December 2020. The payment for the invoice was actually made on 14th February 2021 so it was 73 days past due. Similarly for the second invoice(1002) which was due on 10th December 2020, the payment was actually made on 23rd December 2020, 13 days after the original due date.
So the objective is to build a model predict the payment date for each of the invoices.

Milestone 4: UI Representation of the Data:

The UI consists of a single screen :

Receivables Dashboard Page

It consists of 2 sections:

1. Header
2. Grid Panel Section

1. Header Section

The header consists of :

i. **Account name logo** <ABC Products>on the left,
ii. The HighRadius Logo in the center.

2. Grid Panel Section

The Grid panel section will be divided into 3 portions:

- The name of the grid i.e **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 following columns:

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. Notes

- The **Predicted Payment Date values**, exported out as a CSV from ML model, need to be loaded along with the rest of the data unto the UI.
- The grid will also have a **Select all and Deselect All functionality** to select one or more records.

- The past due invoices should have the due date marked in red in the grid i.e any invoices where due date is smaller than current date, should be in red.
- To support pagination, we will have nav buttons on the screen

Milestone 5: Data Loading to DataBase:

- **All the Columns of the CSV file need to be loaded into the DB.**
- **The Predicted Payment Date from ML Model also needs to be loaded into the Database and then to UI.**

List of all the fields part of dataset are as follows:

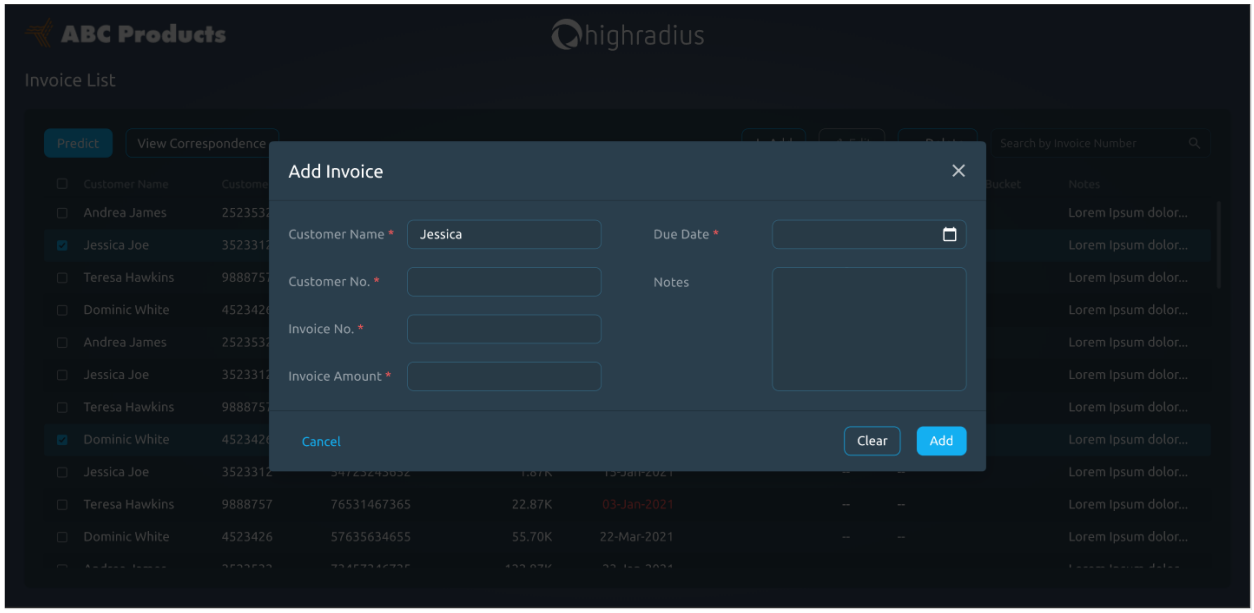
- Acct Doc ID
- Company ID
- Document Number
- Document Number Norm
- Business Code
- Create Year
- Document type
- Customer Number
- Customer Number Norm
- Customer Map ID
- Customer Name
- Document Create Date
- Document Create Date Norm
- Posting date
- Posting date norm

- Posting ID
- Due In Date
- Due In Date Norm
- Order Create Date
- Order Create Date Norm
- Invoice ID
- Invoice ID Norm
- Baseline Create Date
- Invoice Date norm
- Total Open Amount
- Customer Payment Terms
- Shipping date
- Shipping to
- Clear Date
- Clear Date Norm
- Is Open Invoice
- Doc Id
- Actual Outstanding amount
- Invoice Amount
- Dispute Valid Status
- Dispute Amount

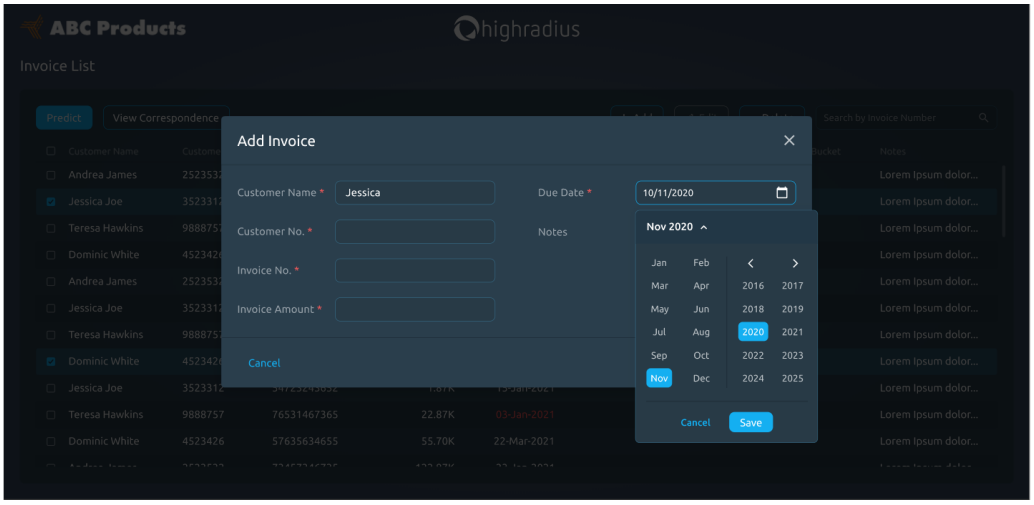
Milestone 6: Functionalities in Detail:

Add Button Functionality :

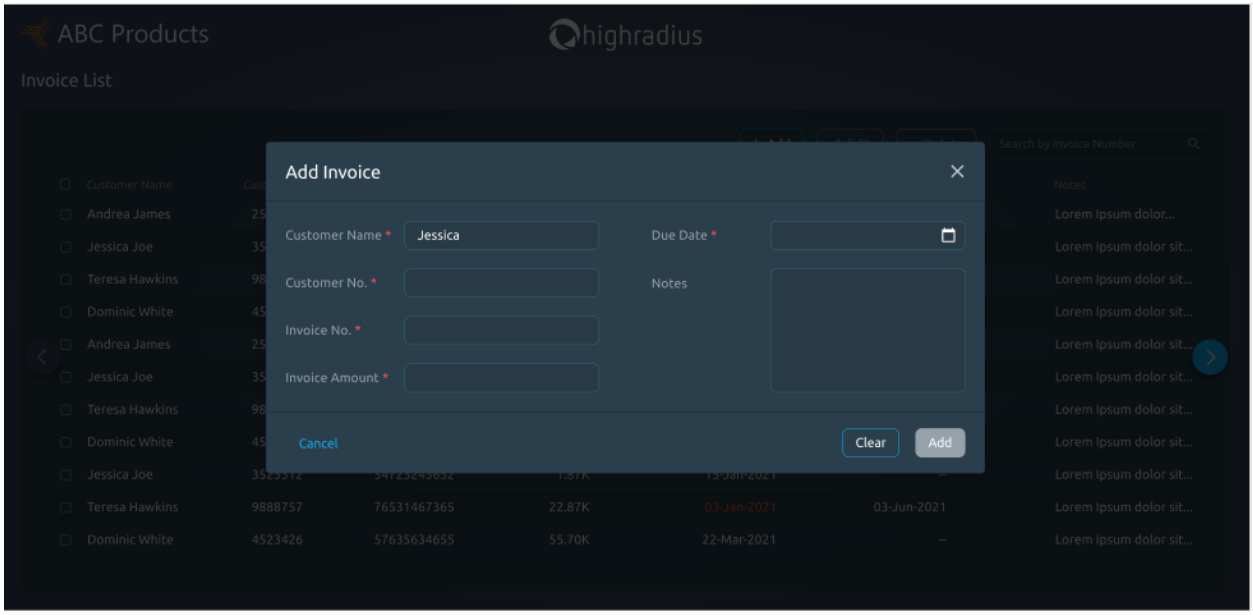
- The Add button will remain in the enabled **state** if no rows are selected.
- Whenever more than one row is selected, the Add button will be disabled.
- After clicking on the Add button, a pop up window will appear with all the fields for which values need to be added along with a Clear and an Add button.
- The user should be able to **type in the values,** except for the due date of the invoice for which there should be a calendar view from where the user is able to select the required date, month and year.
- The user should fill all the required fields before adding.



- Once the user clicks on the add button after all mandatory fields are filled, the new values should be displayed in the UI and should remain persistent.

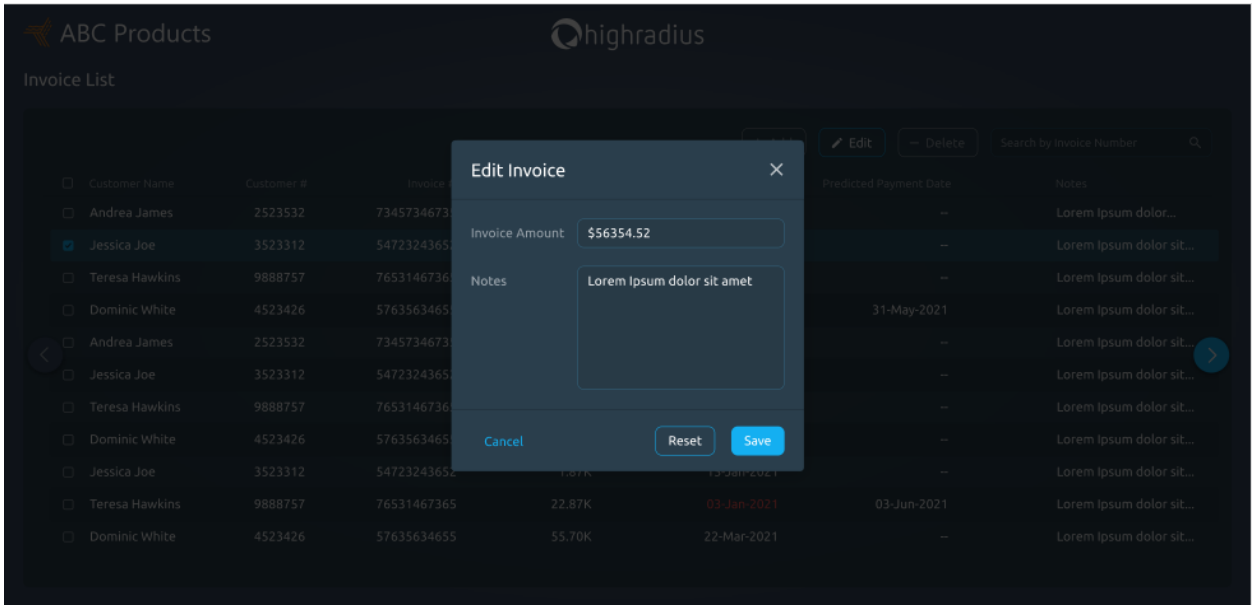


- The add button will remain in disabled state, if the user tries to click on add before all mandatory fields are filled.



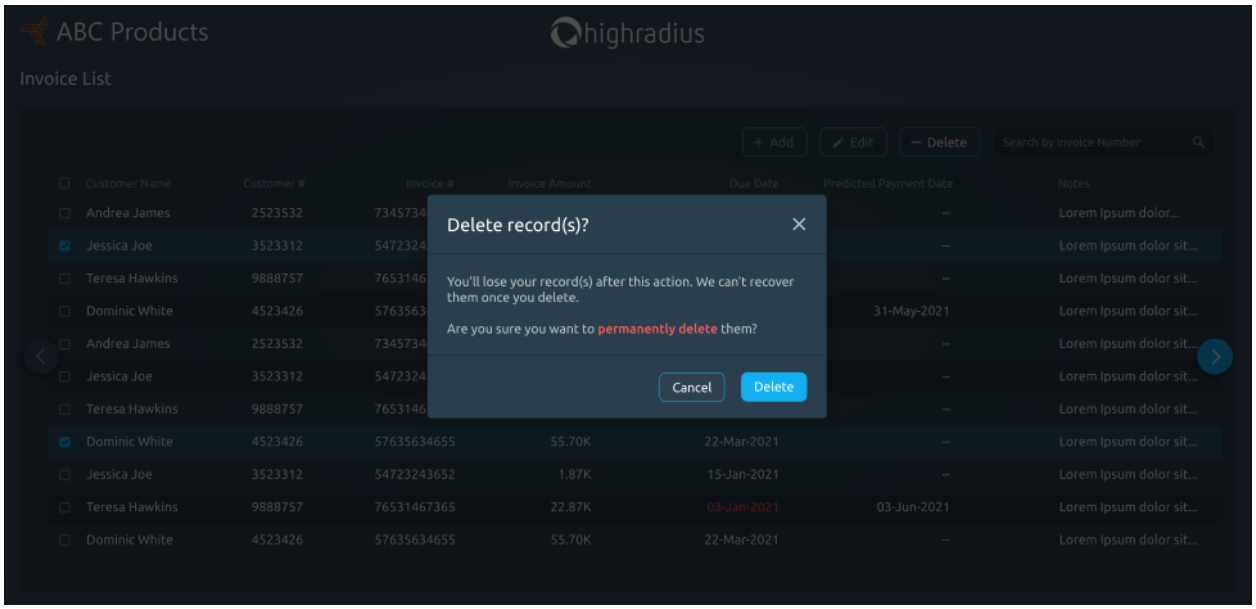
Edit Button Functionality :

- Clicking on the **edit button** will allow the user to **modify the editable fields** in the data.
- The editable columns are **Invoice Amount** and **Notes.**
- The edit button will **remain in disabled state** by default.
- When the **user selects one row**, the Edit button gets enabled.
- If a user selects multiple rows, the edit button will remain disabled.
- Clicking on the Edit button displays a popup window on the screen. The window should contain the Invoice Amount and Notes headers along with the existing data values for both, a Reset and a Save button.
- The user should be able to **edit the values.**
- Once the user clicks on the save button, the new values should be displayed in the UI and should remain persistent.



Delete Button Functionality :

- Clicking on the **delete button** will allow the user to **delete records** from the grid.
- The delete button will **remain in disabled state** by default.
- When the **user selects one or more rows**, the delete button gets enabled.
- A pop up should be displayed on clicking delete to confirm that the user wants to delete the selected records permanently.
- Once the user clicks on the delete button, the row(s) should be removed from the grid in the UI and should remain persistent.



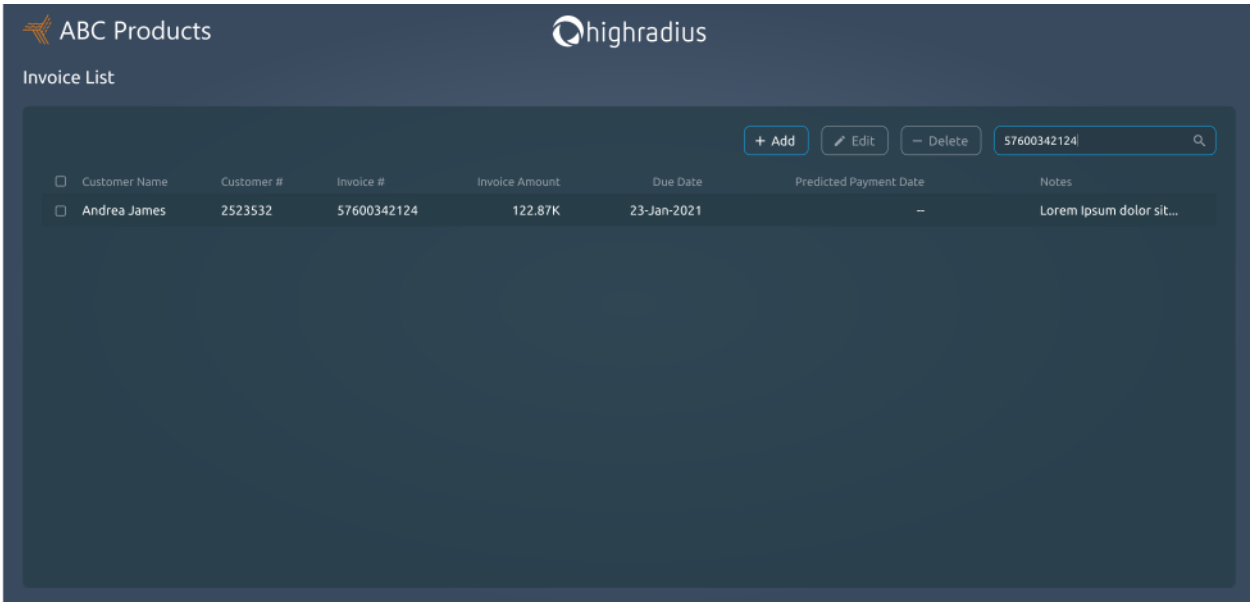
Navigation Functionality:

- The user should be able to navigate to move across different pages in grid.
- Two nav buttons will be present to go to the Next and Previous page.

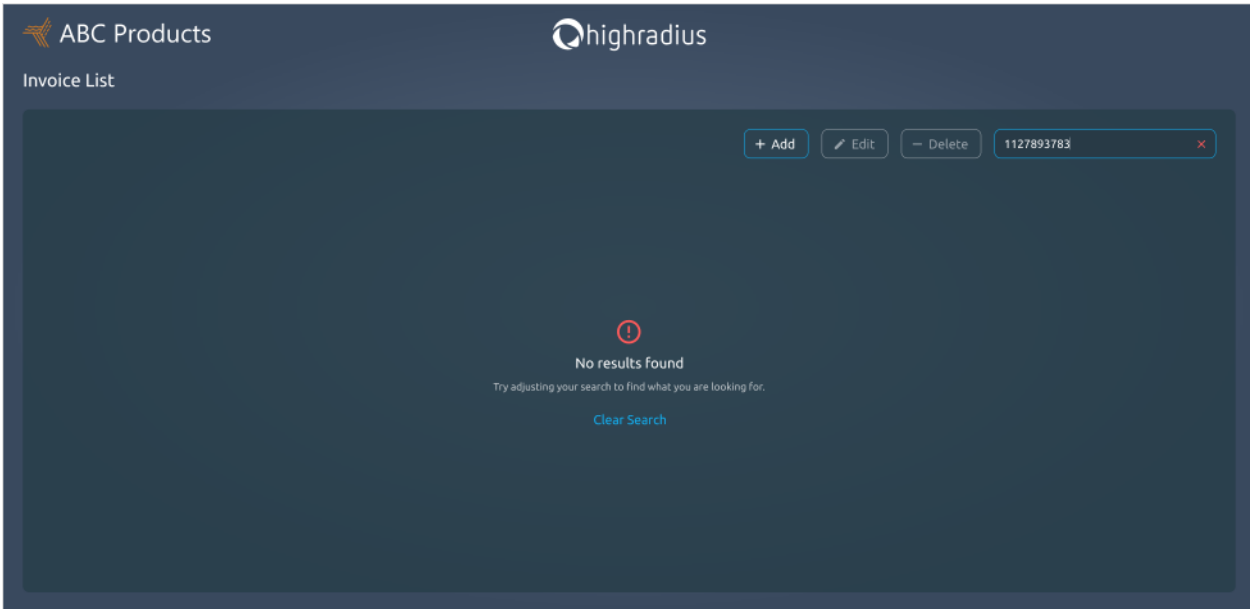
- If user is on first page, previous nav button will be disabled, if user is present on last page, next nav button will be disabled

Optional Milestone: Search Bar Functionality:

- The user should be able to type the complete invoice number in the search bar.
- The invoice with matching invoice number should be filtered in the grid.



- If no such invoice is present, a message should be displayed stating “No Results Found”.



Glossary

1. Invoice - A document which is issued by a seller to a buyer when some goods are purchased. The fields which can be part of the invoice are defined below
2. Open Invoice - Invoices which are not cleared (payment not done) are called Open Invoices.
3. Closed Invoice - Invoices which are cleared (payment done) are called Closed Invoices.
4. B2B - Business to Business
5. B2C- Business to Consumer
6. C2C - Consumer to Consumer
7. Payment Terms - These indicate the period within which payments should be made and how. These terms are usually included in the invoices generated by companies and sent to customers.Eg Net 30, Net 60

UX/XD

https://xd.adobe.com/view/59f0e940-f492-4218-97fd-8a50f0b1bd3a-0ca3/