

# Vishwa Soni: **Gold Investment Management System**

---

## Problem Statement:

Develop a web application for managing gold investment portfolios. The system should allow users to perform CRUD operations on their gold investment records. Each record should include details such as:

- **Gold Type:** (e.g., bullion, coins, jewelry)
- **Weight:** (e.g., in grams)
- **Purity:** (e.g., 24K, 22K)
- **Purchase Price:** (e.g., in USD)
- **Current Value:** (e.g., in USD)
- **Date of Purchase**

## Features:

- **Create:** Users can add new gold investment records.
- **Read:** Users can view a list of all their gold investments, as well as details for each record.
- **Update:** Users can update the details of existing records.
- **Delete:** Users can remove records from their portfolio.

## Additional Requirements:

- Implement data validation to ensure that all input values are appropriate (e.g., weight is positive, purity is valid).
- Use JSP to display data in a tabular format and forms for input.
- Implement a search feature to filter records based on gold type or purchase date.

# Mohak Patel: Food Recipe Management System

---

## Problem Statement:

Create a web application for managing food recipes. The system should allow users to perform CRUD operations on recipes, which include the following information:

- **Recipe Name:** (e.g., Spaghetti Carbonara)
- **Ingredients:** (e.g., spaghetti, eggs, pancetta, parmesan cheese)
- **Instructions:** (detailed steps for preparing the dish)
- **Preparation Time:** (e.g., in minutes)
- **Category:** (e.g., appetizer, main course, dessert)

## Features:

- **Create:** Users can add new recipes.
- **Read:** Users can view a list of all recipes, search for recipes by name or category, and see detailed instructions for each recipe.
- **Update:** Users can edit existing recipes to modify ingredients, instructions, or other details.
- **Delete:** Users can remove recipes they no longer want.

## Additional Requirements:

- Implement file upload functionality to allow users to upload images of the dishes.
- Use JSP to display recipes with images and provide forms for recipe input.
- Implement user authentication to ensure only registered users can modify or delete recipes.

# Dhyey Patel: Travel Destination Management System

---

## Problem Statement:

Build a web application for managing travel destinations. The system should enable users to perform CRUD operations on travel destinations with the following attributes:

- **Destination Name:** (e.g., Paris, Tokyo, New York)
- **Country:** (e.g., France, Japan, USA)
- **Description:** (brief information about the destination)
- **Best Travel Time:** (e.g., summer, winter)
- **Average Cost:** (e.g., in USD)

## Features:

- **Create:** Users can add new travel destinations.
- **Read:** Users can view a list of all destinations, search by country or best travel time, and see detailed descriptions.
- **Update:** Users can modify details of existing destinations.
- **Delete:** Users can delete destinations from their list.

## Additional Requirements:

- Include a map view to display destinations using an API (e.g., Google Maps).
- Use JSP to display destinations with interactive features (e.g., search, sorting).
- Implement sorting options to view destinations based on cost or best travel time.