**Flowchart for Statistical Prediction Platform for Insurance Companies**

Project name: A statistical prediction platform for insurance Companies.

Subject focus : Risk Assessment and Predictive Analytics for Strategic Planning(Check the proposal)

**This all focus on the UI/UX design interface no functionality.**

**Insurance Analytics Platform Flowchart**

**====================================**

**1. User Authentication**

**├── Login/Signup Flow**

**│ ├── Email/Password Authentication**

**│ └── Role Assignment (Admin/User)**

**└── Session Management**

**└── JWT Token Handling**

**2. Dashboard (Main Interface)**

**├── Header Component**

**│ ├── Company Logo**

**│ ├── Search Functionality**

**│ └── User Profile & Notifications**

**│**

**├── Sidebar Navigation**

**│ ├── Dashboard Overview**

**│ ├── Upload Dataset**

**│ ├── Fraud Detection**

**│ ├── Claims Prediction**

**│ ├── Reports & Insights**

**│ └── Settings**

**│**

**└── Main Content Area**

**├── KPI Cards**

**│ ├── Total Claims**

**│ ├── Fraud Detected**

**│ ├── Predicted Claims**

**│ └── Risk Score**

**│**

**└── Interactive Charts**

**├── Claims Distribution (Doughnut)**

**├── Claims Trend (Line)**

**└── Actual vs Predicted (Bar)**

**3. Dataset Management**

**├── Upload Interface**

**│ ├── File Upload (CSV/XLSX/JSON)**

**│ ├── Drag & Drop Functionality**

**│ └── Progress Tracking**

**│**

**├── Data Validation**

**│ ├── Format Checking**

**│ ├── Required Fields Validation**

**│ └── Error Reporting**

**│**

**└── Data Transformation**

**├── Column Mapping**

**├── Data Cleaning**

**│ ├── Missing Values**

**│ ├── Duplicates**

**│ └── Format Normalization**

**└── Preview & Confirmation**

**4. Analytics Engine**

**├── Fraud Detection**

**│ ├── ML Model Processing**

**│ ├── Risk Scoring**

**│ └── Alert Generation**

**│**

**├── Claims Prediction**

**│ ├── Time Series Analysis**

**│ ├── Trend Forecasting**

**│ └── Confidence Intervals**

**│**

**└── Real-time Updates**

**├── WebSocket Connection**

**└── Live Data Streaming**

**5. Reporting System**

**├── Report Generation**

**│ ├── PDF Export**

**│ ├── Excel Export**

**│ └── Custom Formatting**

**│**

**├── Scheduling**

**│ ├── Automated Reports**

**│ └── Email Distribution**

**│**

**└── Data Visualization**

**├── Interactive Charts**

**├── Filtering Options**

**└── Custom Views**

**6. Settings & Configuration**

**├── Profile Management**

**│ ├── Personal Info**

**│ ├── Password Update**

**│ └── Profile Picture**

**│**

**├── System Preferences**

**│ ├── Theme Selection**

**│ ├── Chart Preferences**

**│ └── Real-time Updates**

**│**

**├── Notifications**

**│ ├── Alert Settings**

**│ ├── Email Preferences**

**│ └── System Notifications**

**│**

**└── Data Management**

**├── Retention Policies**

**├── Backup Settings**

**└── Privacy Controls**

**7. Footer Section**

**├── Copyright Information**

**├── Quick Links**

**└── Support Contact**

**Key Design Considerations:**

1. **Primary Color Scheme:**
   * **Deep Blue**: For headers, navigation bars, and chart elements.
   * **White**: For backgrounds and text areas to create contrast.
2. **UI/UX Elements:**
   * **Smooth Animations**: Button clicks, sidebar toggle, and chart loading.
   * **Responsive Design**: Ensures compatibility with all screen sizes.
   * **Date Tracking**: Prominent date picker in filters and dataset upload sections.
3. **Icons:**
   * Use intuitive icons for sidebar items and action buttons.
   * Incorporate animations for hover effects.
4. **Charts:**
   * Use **Chart.js** or **D3.js** for interactivity.
   * Colors: Different shades of blue and contrasting colors for clarity.
5. **Footer:**
   * Minimal and lightweight with essential links like "Privacy Policy" and "Contact Support."
6. **Animations:**
   * Animated KPI card number counters.
   * Smooth transitions between sections when navigating.
   * Charts should load dynamically with fade or grow effects.

**1. Dashboard Overview**

**Purpose**: Provide a quick summary of key metrics and insights. **Content**:

* **Welcome Message**: "Welcome, [User Name]! Here's a snapshot of your system's performance."
* **KPI Cards**:
  + Total Claims (Count)
  + Fraud Detected (Count or Percentage)
  + Predicted Claims (Count or Trends)
  + Upcoming Risks (Alerts or Probability)
* **Quick Links**:
  + View Detailed Reports
  + Upload New Data
* **Charts Section**:
  + Doughnut Chart: Fraudulent vs Legitimate Claims
  + Line Chart: Claim Likelihood Trends
  + Bar Chart: Actual vs Predicted Claims

### ****2. Updated Content for the Upload Dataset Section****

#### **Purpose**

Enable admins to upload, preview, validate, and transform datasets before final submission for analysis.

### ****Content Details****

#### **1. Instructions Section**

* **Header:**  
  "Upload Your Dataset for Analysis"
* **Guidelines:**
  + Supported Formats: .csv, .xlsx, .json
  + "Drag and drop your file below or click to browse."
  + "Ensure your dataset contains the required fields: Policy Type, Claim Amount, Date, etc."
* **Tips Box:**
  + Expected date format: YYYY-MM-DD.
  + Numeric fields must not include special characters.
  + Avoid empty rows or columns.

#### **2. Upload Area**

* **Drag-and-Drop Zone**:
  + Display a large drop area for file uploads.
  + Include a progress bar to show upload status.
* **Browse Button**:
  + Option for manual file selection.
* **Error Handling**:
  + Show an error message if the file type or structure is incorrect.

#### **3. Preview Section**

After upload, the admin is redirected to the **Preview and Transform Section**.

* **Preview Table**:  
  Display the uploaded dataset in a tabular format with the following:
  + Rows and columns clearly labeled.
  + Sorting and filtering options for each column.

#### **4. Validation Results**

* **Automatic Checks**:
  + Highlight problematic rows (e.g., missing values, duplicates, invalid formats).
  + Use **color codes**:
    - Green: Valid data
    - Red: Invalid data
    - Yellow: Warnings
* **Summary Box**:
  + Number of rows flagged with issues.
  + Total missing or duplicate values.

#### **5. Data Transformation**

Allow the admin to modify the dataset directly.

* **Editable Table**:
  + Inline editing of cells.
  + Dropdown menus for predefined categories like Policy Type.
* **Transform Options**:
  + **Column Mapping**:  
    Admin maps uploaded column names to expected fields (e.g., Claim\_Amount → Amount).
  + **Fill Missing Values**:  
    Provide options like replacing blanks with 0, average, or default values.
  + **Remove Duplicates**:  
    Highlight duplicates with the option to delete or resolve them.
  + **Normalize Data**:  
    Reformat fields (e.g., convert MM/DD/YYYY → YYYY-MM-DD).

#### **6. Final Validation**

* Recheck the data after transformation.
* Display a "Validation Passed" message if the dataset is ready.
* Provide a **Download Corrected Dataset** option.

#### **7. Action Buttons**

* **Upload Dataset**:  
  Save the cleaned data to the database for analysis.
* **Reset**:  
  Clear the uploaded data and start over.
* **Download Corrected File**:  
  Allow the admin to download the validated dataset.

**3. Fraud Detection**

**Purpose**: Highlight flagged claims and potential frauds.

**Content**:

* **Summary Section**:
  + Total Claims Reviewed
  + Fraud Detection Accuracy (%)
  + Savings from Detected Fraud (Financial Amount)
* **Charts Section**:
  + Fraudulent Claims over Time (Line Chart)
  + Fraud Types by Frequency (Bar Chart)
* **Data Table**:
  + Detailed flagged claims:
    - Claim ID, Policy Type, Amount, Status (Fraudulent/Legitimate), Date
* **Filters**:
  + Date Range
  + Claim Amount
  + Policy Type

**4. Claims Prediction**

**Purpose**: Analyze trends in claim forecasts and future risks.

**Content**:

* **Charts Section**:
  + Forecasted Claims Volume (Line Chart)
  + Comparison of Actual vs Predicted Claims (Bar Chart)
  + Risk Distribution by Region/Policy Type (Pie or Heat Map)
* **Insights Section**:
  + Predicted claims for the next 5 years.
  + Future risk areas flagged.
* **Quick Filters**:
  + Policy Type
  + Location
  + Date Range

**5. Reports & Insights**

**Purpose**: Provide detailed analysis and downloadable reports.

**Content**:

* **Predefined Reports**:
  + Fraud Summary Report
  + Claims Volume Trends
  + Financial Implications of Predictions
* **Custom Report Builder**:
  + Select Metrics: Fraud Savings, Total Claims, Predicted Risks.
  + Filters: Policy Type, Claim Type, Date Range.
* **Download Section**:
  + Buttons to export as PDF/Excel.
  + Shareable Links.
* **Visualization**:
  + Key charts embedded for visual insights.

**6. Settings**

**Purpose**: Manage system preferences and configurations.

**Content**:

* **Profile Management**:
  + Update Profile Information
  + Change Password
* **System Preferences**:
  + Toggle Real-Time Updates
  + Adjust Chart Themes (Light/Dark Mode)
* **Notifications**:
  + Enable/Disable Alerts for fraud or prediction updates.
* **Data Settings**:
  + Data Retention Policies
  + Manage Uploaded Datasets
* **Help & Support**:
  + FAQ
  + Contact Support (Email, Phone)

**Additional Notes:**

* Each menu item should have **tooltips** for clarity.
* Use **icons** alongside menu names for visual appeal:
  + Dashboard Overview: 🏠 (Home Icon)
  + Upload Dataset: 📂 (Folder Icon)
  + Fraud Detection: 🚨 (Alert Icon)
  + Claims Prediction: 📈 (Chart Icon)
  + Reports & Insights: 📊 (Graph Icon)
  + Settings: ⚙️ (Gear Icon)

now looking a the complete project, give me the whole db for this project

**Flowchart Updates**

The flowchart for the entire process should include the following stages to ensure that the platform effectively serves its purpose:

1. **User/Admin Login**
   * User is authenticated and granted access to different functionalities (Admin/ User roles).
2. **Upload Dataset**
   * Admin selects and uploads the dataset (CSV, Excel, JSON).
   * **Validation**: System checks for missing values, duplicates, or formatting issues in the uploaded dataset.
3. **Dataset Preprocessing**
   * Data is preprocessed (missing values handled, duplicates removed, features extracted).
   * **Transformation & Preview**: Admin can preview the dataset and make necessary corrections before finalizing the upload.
4. **Dataset Storage & Access**
   * The cleaned data is stored in the database for analysis.
   * The dataset is linked to future predictions, claims, and forecasting tasks.
5. **Model Training & Forecasting**
   * Based on historical data, the system uses ML models (e.g., ARIMA, Prophet) for time-series forecasting of future claims and fraud patterns.
   * Admin is notified once the model is trained and ready for use.
6. **Dashboard Visualization**
   * Forecasts and analytics results are displayed in the dashboard with interactive charts:
     + **Doughnut chart** for fraudulent vs legitimate claims.
     + **Bar chart** for year-on-year claims comparison.
     + **Line chart** for claims trends and fraud projections.
   * Dynamic filters allow the admin to drill down into specific years, claim types, or policy types.
7. **Real-time Prediction Updates**
   * When new datasets are uploaded, the system automatically retrains models and updates predictions and charts on the dashboard.
   * Predictions for fraud savings, claims volume, and financial implications are dynamically shown in the dashboard.

**Final Flowchart Updates**

**Step 1:** **Admin Login**

* Authenticate the admin user.
* **Input:** Username, Password.
* **Output:** Access granted to dataset management, forecasting, and dashboard.

**Step 2:** **Upload Dataset**

* **Input:** Choose a file (CSV, Excel, JSON).
* **Action:** Upload file, validate format, check for missing values or duplicates.
* **Output:** File preview and validation results. Admin makes necessary corrections.

**Step 3:** **Preprocess Dataset**

* **Input:** Dataset (after corrections).
* **Action:** Data transformation (remove missing values, handle outliers, extract features).
* **Output:** Cleaned dataset ready for analysis.

**Step 4:** **Store and Process Data**

* **Input:** Cleaned dataset.
* **Action:** Store data in the database for analysis.
* **Output:** Data is ready for predictions.

**Step 5:** **Forecast & Predictive Modeling**

* **Input:** Historical claims data.
* **Action:** Use time-series forecasting models (e.g., ARIMA, Prophet) to generate predictions.
* **Output:** Future claims predictions, fraud likelihood projections.

**Step 6:** **Dashboard Visualization**

* **Input:** Forecasting results and historical data.
* **Action:** Display interactive charts (doughnut, line, bar) on the dashboard.
* **Output:** Insights into claims predictions, fraud detection, and financial savings.

**Step 7:** **Real-time Updates**

* **Input:** New dataset uploaded.
* **Action:** System triggers re-training and updates all predictions in real-time.
* **Output:** Updated predictions on the dashboard.

**Dataset Format:**

* **CSV:** Ideal for tabular, straightforward data with rows and columns.
* **Excel:** Useful for structured data with multiple sheets and features (advanced formatting and formulas).
* **JSON:** Suitable for more complex, hierarchical data structures with nested information.

**How It Works:**

* Admins upload datasets (CSV, Excel, JSON).
* Claims and fraud detection data are processed for predictive modeling.
* Predictive models generate forecasts (claims, fraud, savings).
* The forecasting and fraud detection results are stored and available for analysis.
* Logs track the actions performed by users in the system for transparency and auditing.

**1. CSV Format Example**

customer\_id,customer\_name,age,gender,location,policy\_number,email,claim\_date,claim\_amount,policy\_type,fraud\_flag

1,John Doe,34,Male,New York,ABC123,john.doe@email.com,2023-01-15,1500.00,Health,TRUE

2,Jane Smith,29,Female,California,DEF456,jane.smith@email.com,2023-03-12,2300.00,Auto,FALSE

3,Mary Johnson,45,Female,Texas,GHI789,mary.johnson@email.com,2023-05-20,4000.00,Health,FALSE

4,James Brown,55,Male,Florida,JKL012,james.brown@email.com,2023-07-22,3200.00,Home,TRUE

5,Emily White,38,Female,Georgia,MNO345,emily.white@email.com,2023-08-03,2500.00,Auto,FALSE

6,David Green,50,Male,Ohio,PQR678,david.green@email.com,2023-09-10,3800.00,Health,FALSE

**DATABASE**

**Database Structure Summary**

1. **Users Table** – For storing admin login credentials and roles.
2. **Datasets Table** – Manages metadata related to uploaded datasets (e.g., CSV, Excel, JSON).
3. **Claims Table** – Stores claims data for analysis and predictions.
4. **Customers Table** – Stores customer demographic data.
5. **Fraud Detection Table** – Stores fraud detection results and scores for claims.
6. **Predictions Table** – Stores forecast results for claims, fraud, savings, etc.
7. **Model Training Table** – Tracks the model training process and its status.
8. **Forecasting Results Table** – Stores the year-by-year forecast data.
9. **Logs Table** – Stores logs for tracking actions and updates to the system.

SQL STRUCTURE

TABLES

**1. Tables Created**

**-users(admin)**

**- claims**

**- audit\_trail**

**- risk\_factors**

**- datasets**

**- customers**

**- model\_training**

**- forecasting\_results**

**- logs**

-- Database creation

CREATE DATABASE insurance\_prediction;

USE insurance\_prediction;

-- Users Table

CREATE TABLE users (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

username VARCHAR(255) NOT NULL UNIQUE,

password\_hash VARCHAR(255) NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Datasets Table

CREATE TABLE datasets (

dataset\_id INT AUTO\_INCREMENT PRIMARY KEY,

dataset\_name VARCHAR(255) NOT NULL,

uploaded\_by INT,

upload\_date TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

file\_format ENUM('csv', 'xlsx', 'json'),

status ENUM('pending', 'processed', 'error') DEFAULT 'pending',

FOREIGN KEY (uploaded\_by) REFERENCES users(user\_id)

);

-- Customers Table

CREATE TABLE customers (

customer\_id INT AUTO\_INCREMENT PRIMARY KEY,

customer\_name VARCHAR(255),

age INT,

gender ENUM('Male', 'Female', 'Other'),

location VARCHAR(255),

policy\_number VARCHAR(255),

email VARCHAR(255)

);

-- Claims Table

CREATE TABLE claims (

claim\_id INT AUTO\_INCREMENT PRIMARY KEY,

claim\_date DATE,

claim\_amount DECIMAL(10, 2),

policy\_type VARCHAR(255),

fraud\_flag BOOLEAN DEFAULT FALSE,

customer\_id INT,

claim\_status ENUM('open', 'closed', 'pending'),

dataset\_id INT,

FOREIGN KEY (dataset\_id) REFERENCES datasets(dataset\_id),

FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)

);

-- Fraud Detection Table

CREATE TABLE fraud\_detection (

fraud\_id INT AUTO\_INCREMENT PRIMARY KEY,

claim\_id INT,

fraud\_score DECIMAL(5, 2),

investigation\_status ENUM('pending', 'completed', 'under investigation'),

FOREIGN KEY (claim\_id) REFERENCES claims(claim\_id)

);

-- Predictions Table

CREATE TABLE predictions (

prediction\_id INT AUTO\_INCREMENT PRIMARY KEY,

prediction\_type ENUM('fraud', 'claims', 'savings'),

forecast\_date DATE,

forecasted\_value DECIMAL(10, 2),

dataset\_id INT,

model\_used VARCHAR(255),

FOREIGN KEY (dataset\_id) REFERENCES datasets(dataset\_id)

);

-- Model Training Table

CREATE TABLE model\_training (

model\_id INT AUTO\_INCREMENT PRIMARY KEY,

model\_name VARCHAR(255),

last\_trained TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

status ENUM('training', 'trained', 'error'),

trained\_by INT,

FOREIGN KEY (trained\_by) REFERENCES users(user\_id)

);

-- Forecasting Results Table

CREATE TABLE forecasting\_results (

result\_id INT AUTO\_INCREMENT PRIMARY KEY,

forecast\_year INT,

forecasted\_claims DECIMAL(10, 2),

actual\_claims DECIMAL(10, 2),

forecasted\_fraud DECIMAL(10, 2),

actual\_fraud DECIMAL(10, 2),

dataset\_id INT,

FOREIGN KEY (dataset\_id) REFERENCES datasets(dataset\_id)

);

-- Logs Table

CREATE TABLE logs (

log\_id INT AUTO\_INCREMENT PRIMARY KEY,

action VARCHAR(255),

user\_id INT,

timestamp TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

details TEXT,

FOREIGN KEY (user\_id) REFERENCES users(user\_id)

);

FINAL STEP

Now give me the full db structure for this project.

## Major Technologies

- HTML5

- CSS

- JAVASCRIPT

- BOOTSTRAP

- PHP