**UE22CS341A: Database Management Systems**

**Case Study**

# STRATEGIC BUSINESS INSIGHT ENGINE

**-Empowering Entrepreneurs with Data-Driven Business Insights.**

**Arushi R Katta – PES1UG22CS109** **Amritaa Kalanee – PES1UG22CS077**

**GITHUB REPO LINK:** *https://github.com/aruk04/business-insight-engine*

*https://github.com/amritaak06/Business-Insight-Engine*

**Synopsis**

**Strategic Business Insight Engine** is an innovative platform designed to help aspiring entrepreneurs by providing comprehensive insights tailored to their business needs. This tool helps users refine their business ideas and navigate the complexities of starting and growing a business by offering detailed, strategic information in several key areas.

The platform identifies **potential competitors** within the user's target industry, providing detailed analyses of their market reach, strengths, weaknesses, and relevance. This allows users to understand the competitive landscape and position their business effectively.

**Strategic partnerships** are another critical focus, with the engine recommending partners that align with the user's business objectives. These partnerships are supported by relevant data on **vendors and suppliers**, detailing their feasibility and credibility within the context of the user's industry.

To further assist in market positioning, the platform connects users with expert **market analysts** who offer insights into current **market trends**, helping users to identify opportunities and threats in their target market.

For those seeking funding, the engine attracts **potential investors** interested in the user's business area, providing an opportunity to secure financial backing. In addition, the tool recommends **legal advisories** to ensure that all **contracts**, including MOUs (Memorandums of Understanding), and operations are compliant with regulatory requirements.

By consolidating all these resources into one user-friendly interface, Strategic Business Insight Engine enables entrepreneurs to make informed decisions, plan effectively, and set their businesses up for success.

Table of Contents

## 1. Introduction

### 1.1 Purpose

The purpose of this document is to outline the requirements for the **Strategic Business Insight Engine**. This system is designed to provide valuable business insights by leveraging SQL queries and CRUD operations, with a simplistic GUI for smooth user interaction. The system will focus on efficiently managing and analysing data through a central database, enabling users to retrieve helpful business insight. An iterative development model will be employed, allowing continuous refinement and improvements over multiple development cycles.

### 1.2 Product Scope

The Strategic Business Insight Engine is designed to assist entrepreneurs in gathering and analysing crucial business information. The system will utilize SQL queries and CRUD operations to manage data, offering insights into competitors, partnerships, market trends, and more through a centralized database.

## 2. Overall Description

### 2.1 Product Perspective

The **Strategic Business Insight Engine** is an independent application designed to support entrepreneurs by providing data-driven business insights. It functions through a centralized database, using SQL queries and CRUD operations to manage and analyze information on competitors, partnerships, market trends, and other related data. The iterative development approach ensures continuous improvement and adaptation, allowing the system to evolve based on user feedback and testing.

Key functions of the Strategic Business Insight Engine include:

- **Data Analysis**: Automated analysis of business data using SQL queries to extract insights on market intelligence.
- **Insight Management**: Creating, updating, and deleting data queries and rules to tailor the insights provided by the system.
- **Competitor Analysis**: Identification and comparison of potential competitors, highlighting their strengths, weaknesses, and market relevance.
- **User Management**: Role and permissions management for system users, ensuring secure access to business data and insights.
- **Reporting**: Generation of detailed reports on business insights, including competitor analysis, market trends, and potential partnerships.

### 2.2 User Classes and Characteristics

- **Entrepreneurs**: Review provided business insights, analyze competitor information, and make strategic decisions.
- **Administrators**: Configure system settings, manage data queries, oversee user roles, and maintain the integrity of the insights provided.
- **End-Users**: Input business ideas and fields of interest to receive tailored insights.
- **Developers**: Implement and maintain the system, ensuring continuous improvement based on iterative feedback and evolving business needs.

### 2.3 Operating Environment

- Server : Windows-based server
- Database : SQL-based (MySQL)
- Client : Web-based interface compatible with modern browsers (Chrome, Firefox, Edge)

### 2.4 Design and Implementation Constraints

- SQL queries and CRUD operations must be optimized for performance and scalability as the database grows.

### 2.5 Assumptions and Dependencies

- The system will rely on external data sources providing necessary access for retrieving relevant business information.
- SQL queries and CRUD operations will be sufficient for managing and analysing all business-related data within the centralized database.

### 2.6 Software Tools and Programming Languages

- *Frontend (Python) : Visual Studio Code*
- *Database : MySQL*
- *Collaboration and Version Control : Git and GitHub*

## 3. External Interface Requirements

### 3.1 User Interfaces

- Web Dashboard: A minimal and user-friendly interface for administrators to manage SQL queries, CRUD operations, and oversee user roles within the Strategic Business Insight Engine.

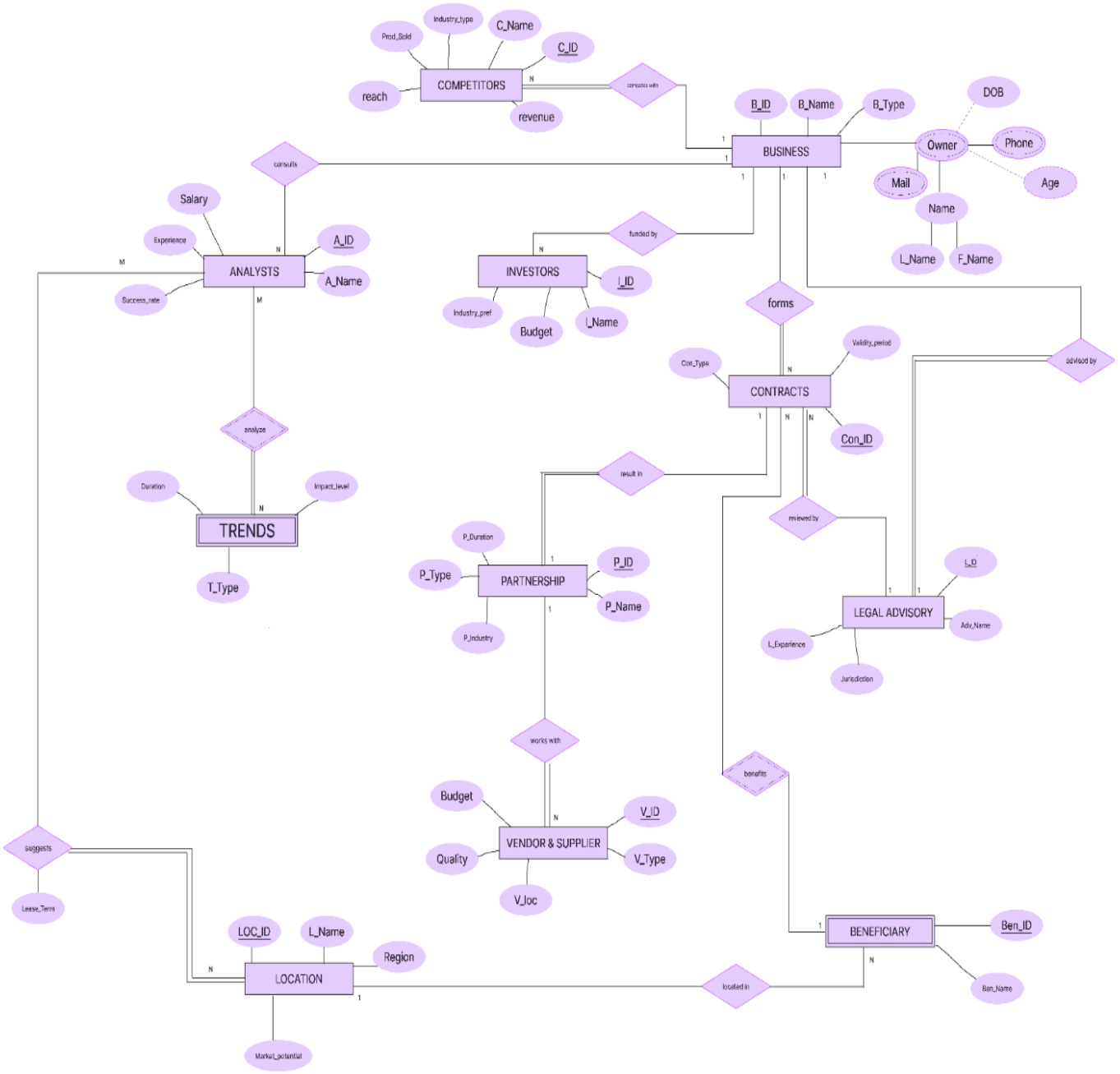
### 3.2 Software Interfaces

- Operating System: Windows Server
- Database: SQL-based (MySQL)
- Web Server: Microsoft Edge
- Programming Languages: Python

## 4. Analysis Models

- Entity Relationship Diagrams: Flowchart that represents the relations between entities. It also includes their attributes.
- Relational Schema: A logical model that represents the structure of a relational database, including the tables, attributes, and relationships between them.

## 4.1 ER DIAGRAM



## 4.2 RELATIONAL SCHEMA

## 5. System Features

### 5.1 System Feature 1: Data Analysis

- Description: Analyse business data, such as competitor information, market trends, and partnership details, using SQL queries to extract actionable insights.
- Priority: High
- Acceptance Criteria: Must accurately retrieve and present relevant business insights based on user queries and data inputs.

### 5.2 System Feature 2: Insight Management

- Description: Provide functionalities for administrators to create, update, and manage SQL queries and rules for data retrieval and analysis.
- Priority: Medium
- Acceptance Criteria: Administrators can add, modify, and delete data queries and rules through the web interface, ensuring flexibility in managing business insights.

### 5.3 System Feature 3: Data Reporting

- Description: Automatically generate and provide access to detailed reports on business insights, including competitor analysis, market trends, and partnership effectiveness.
- Priority: High
- Acceptance Criteria: Reports must be accurate, comprehensive, and available on-demand, summarizing key data points and insights.

### 5.4 System Feature 4: Competitor and Market Trend Analysis

- Description: Provide detailed analysis of competitors and market trends based on input data, using SQL queries to evaluate competitors' strengths, weaknesses, and market positioning.
- Priority: High
- Acceptance Criteria: The system must provide accurate analyses of competitors and market trends, allowing for effective comparison and strategic decision-making.

### 5.5 System Feature 5: User Management

- Description: Manage user roles and permissions within the system.
- Priority: Medium
- Acceptance Criteria: Admins can create, modify, and delete user roles and permissions.

## 6. Other Nonfunctional Requirements

### 6.1 Performance Requirements

- Requirement: The system must process and categorize content within 5 seconds of submission.
- Verification: Performance testing to ensure response times meet the requirement.

### 6.2 Safety Requirements

- Requirement: The system must have mechanisms to prevent data corruption and ensure data recovery.
- Verification: Safety testing and validation procedures.

### 6.3 Security Requirements

- Requirement: Data must be encrypted in transit and at rest, and multi-factor authentication must be supported.
- Verification: Security audits and penetration testing.

### 6.4 Software Quality Attributes

- Reliability: System uptime must be 99.9%.
- Maintainability: Code should be modular and well-documented for easy updates.
- Portability: System must be deployable on both Linux and Windows environments.

### 6.5 Business Rules

- Requirement: All flagged content must be reviewed by a human moderator.
- Verification: Functional testing to ensure workflow compliance.

## 7. Other Requirements

- Legal and Compliance : The system must comply with local and international regulations relevant to content moderation.

## Appendices

### Appendix A: Field Layouts

- Database Tables: Detailed layouts of database tables including fields, types, and relationships.
- UI Forms: Layouts of web interface forms for rule management, content review, and reporting.

**Appendix B: Requirement Traceability Matrix**

| Requirement ID | Requirement Description | Design Specification ID | Implementation Module/Code | Test Case ID | Verification Method | Comments |
|---|---|---|---|---|---|---|
| FR-01 | The system must analyze potential competitors in the user's industry and provide detailed insights. | DS-01 | IM-001 | TC-01 | Functional Test | - |
| FR-02 | The system must suggest strategic partnerships relevant to the user’s business field. | DS-02 | IM-002 | TC-02 | Functional Test | - |
| FR-03 | The system must identify relevant vendors and suppliers, including their feasibility and credibility.. | DS-03 | IM-003 | TC-03 | Functional Test | - |
| FR-04 | Administrators must be able to add, edit, and delete content moderation rules via the web interface. | DS-04 | IM-004 | TC-04 | Functional Test | - |

| | | | | | | |
|---|---|---|---|---|---|---|
| FR-06 | The system must generate reports summarizing business insights and actions taken. | DS-06 | IM-006 | TC-06 | Functional Test | - |
| FR-07 | The system must suggest market analysers with expertise in the field of interest of the user | DS - 07 | IM-007 | TC-07 | Functional Test | |
| FR-08 | The system must provide a web-based dashboard for moderators and administrators to manage and review flagged content. | DS-08 | IM-008 | TC-08 | Usability Test | - |
| FR-09 | The system must support user role management, including creating and modifying roles and permissions. | DS-09 | IM-009 | TC-09 | Functional Test | - |
| FR-13 | The system must process and categorize content within 5 seconds of submission. | DS-13 | IM-013 | TC-13 | Performance Test | Requires timing verification |

## 8. SQL Queries and Functionality (With Snippets)

DDL COMMANDS

- CREATE: creation of database “SBIE” and tables.

```
mysql> use sbie;
Database changed
mysql> show tables;
+-----------------+
| Tables_in_sbie  |
+-----------------+
| admin           |
| agent           |
| analysts        |
| archivedtrends  |
| beneficiary     |
| business        |
| competitors     |
| contracts       |
| investors       |
| legal_advisory  |
| location        |
| partnership     |
| trends          |
| vendor_supplier |
+-----------------+
14 rows in set (0.07 sec)
```

```
def create_table():
    c.execute('''
        CREATE TABLE IF NOT EXISTS Business(
            B_ID VARCHAR(50) PRIMARY KEY,
            B_Name VARCHAR(100),
            L_Name VARCHAR(100),
            F_Name VARCHAR(100),
            B_Type VARCHAR(50),
            OO_Mail VARCHAR(100),
            Phone VARCHAR(15)
        )
    ''')
    c.execute('''
        CREATE TABLE IF NOT EXISTS Competitors (
            C_ID VARCHAR(50) PRIMARY KEY,
            C_Name VARCHAR(100),
            Industry_type VARCHAR(50),
            Prod_Sold INT,
            B_ID VARCHAR(50),
            FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)
        )
    ''')

    c.execute('''
        CREATE TABLE IF NOT EXISTS Analysts (
            A_ID VARCHAR(50) PRIMARY KEY,
            A_Name VARCHAR(100),
            Success_rate DECIMAL(5, 2),
            Experience INT,
            Salary DECIMAL(10, 2),
            B_ID VARCHAR(50),
            FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)
        )
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Investors (
                    I_ID VARCHAR(50) PRIMARY KEY,
                    I_Name VARCHAR(100),
                    Industry_pref VARCHAR(50),
                    Budget DECIMAL(15, 2),
                    B_ID VARCHAR(50),
                    FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)
        );
    ''')
```

```
    c.execute(''' CREATE TABLE IF NOT EXISTS Trends(
                    T_Type VARCHAR(50),
                    Duration INT,
                    Impact_level VARCHAR(50),
                    A_ID VARCHAR(50),
                    FOREIGN KEY (A_ID) REFERENCES ANALYSTS(A_ID)
        );
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Contracts (
                    Con_ID VARCHAR(50) PRIMARY KEY,
                    Con_Type VARCHAR(50),
                    Validity_period INT,
                    B_ID VARCHAR(50),
                    I_ID VARCHAR(50),
                    FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID),
                    FOREIGN KEY (I_ID) REFERENCES INVESTORS(I_ID)
        );
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Legal_Advisory (
                    L_ID VARCHAR(50) PRIMARY KEY,
                    Adv_Name VARCHAR(100),
                    L_Experience INT,
                    Jurisdiction VARCHAR(100),
                    Con_ID VARCHAR(50),
                    FOREIGN KEY (Con_ID) REFERENCES CONTRACTS(Con_ID)
        );
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Partnership (
                    P_ID VARCHAR(50) PRIMARY KEY,
                    P_Name VARCHAR(100),
                    P_Type VARCHAR(50),
                    P_Industry VARCHAR(50),
                    B_ID VARCHAR(50),
                    FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)
        );
    ''')
```

```
    c.execute('''CREATE TABLE IF NOT EXISTS Vendor_Supplier (
                    V_ID VARCHAR(50) PRIMARY KEY,
                    V_Name VARCHAR(100),
                    V_Type VARCHAR(50),
                    Budget DECIMAL(15, 2),
                    Quality VARCHAR(50),
                    V_loc VARCHAR(100),
                    B_ID VARCHAR(50),
                    FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)

        );
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Location (
                    LOC_ID VARCHAR(50) PRIMARY KEY,
                    L_Name VARCHAR(100),
                    Market_potential VARCHAR(50),
                    Region VARCHAR(100),
                    V_ID VARCHAR(50),
                    FOREIGN KEY (V_ID) REFERENCES VENDOR_SUPPLIER(V_ID)
        );
    ''')

    c.execute('''CREATE TABLE IF NOT EXISTS Beneficiary (
                    Ben_ID VARCHAR(50) PRIMARY KEY,
                    Ben_Name VARCHAR(100),
                    Age INT,
                    DOB DATE,
                    Lease_Term INT,
                    Mail VARCHAR(100),
                    Phone VARCHAR(15),
                    Owner VARCHAR(100),
                    B_ID VARCHAR(50),
                    FOREIGN KEY (B_ID) REFERENCES BUSINESS(B_ID)
        );
    ''')
```

Given below are snippets of the table descriptions:

```
mysql> desc admin;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| admin_username | varchar(50) | NO   | PRI | NULL    |       |
| admin_password | varchar(255)| NO   |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
2 rows in set (0.03 sec)

mysql> desc agent;
+----------------+-------------+------+-----+---------+-------+
| Field          | Type        | Null | Key | Default | Extra |
+----------------+-------------+------+-----+---------+-------+
| agent_username | varchar(50) | NO   | PRI | NULL    |       |
| agent_password | varchar(255)| NO   |     | NULL    |       |
+----------------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql> desc analysts;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| A_ID         | varchar(50)  | NO   | PRI | NULL    |       |
| A_Name       | varchar(100) | YES  |     | NULL    |       |
| Success_rate | decimal(5,2) | YES  |     | NULL    |       |
| Experience   | int          | YES  |     | NULL    |       |
| Salary       | decimal(10,2)| YES  |     | NULL    |       |
| B_ID         | varchar(50)  | YES  | MUL | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)

mysql> desc archivedtrends;
+---------------+------------+------+-----+---------+-------+
| Field         | Type       | Null | Key | Default | Extra |
+---------------+------------+------+-----+---------+-------+
| T_Type        | varchar(50)| YES  |     | NULL    |       |
| Duration      | int        | YES  |     | NULL    |       |
| Impact_level  | varchar(50)| YES  |     | NULL    |       |
| A_ID          | varchar(50)| YES  | MUL | NULL    |       |
| archived_date | datetime   | YES  |     | NULL    |       |
+---------------+------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

```
mysql> desc beneficiary;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| Ben_ID     | varchar(50) | NO   | PRI | NULL    |       |
| Ben_Name   | varchar(100)| YES  |     | NULL    |       |
| Age        | int         | YES  |     | NULL    |       |
| DOB        | date        | YES  |     | NULL    |       |
| Lease_Term | int         | YES  |     | NULL    |       |
| Mail       | varchar(100)| YES  |     | NULL    |       |
| Phone      | varchar(15) | YES  |     | NULL    |       |
| Owner      | varchar(100)| YES  |     | NULL    |       |
| B_ID       | varchar(50) | YES  | MUL | NULL    |       |
+------------+-------------+------+-----+---------+-------+
9 rows in set (0.01 sec)

mysql> desc business;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| B_ID    | varchar(50) | NO   | PRI | NULL    |       |
| B_Name  | varchar(100)| YES  |     | NULL    |       |
| L_Name  | varchar(100)| YES  |     | NULL    |       |
| F_Name  | varchar(100)| YES  |     | NULL    |       |
| B_Type  | varchar(50) | YES  |     | NULL    |       |
| OO_Mail | varchar(100)| YES  |     | NULL    |       |
| Phone   | varchar(15) | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
7 rows in set (0.00 sec)

mysql> desc competitors;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| C_ID          | varchar(50) | NO   | PRI | NULL    |       |
| C_Name        | varchar(100)| YES  |     | NULL    |       |
| Industry_type | varchar(50) | YES  |     | NULL    |       |
| Prod_Sold     | int         | YES  |     | NULL    |       |
| B_ID          | varchar(50) | YES  | MUL | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

```
mysql> desc investors;
+---------------+--------------+------+-----+---------+-------+
| Field         | Type         | Null | Key | Default | Extra |
+---------------+--------------+------+-----+---------+-------+
| I_ID          | varchar(50)  | NO   | PRI | NULL    |       |
| I_Name        | varchar(100) | YES  |     | NULL    |       |
| Industry_pref | varchar(50)  | YES  |     | NULL    |       |
| Budget        | decimal(15,2)| YES  |     | NULL    |       |
| B_ID          | varchar(50)  | YES  | MUL | NULL    |       |
+---------------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> desc legal_advisory;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| L_ID         | varchar(50)  | NO   | PRI | NULL    |       |
| Adv_Name     | varchar(100) | YES  |     | NULL    |       |
| L_Experience | int          | YES  |     | NULL    |       |
| Jurisdiction | varchar(100) | YES  |     | NULL    |       |
| Con_ID       | varchar(50)  | YES  | MUL | NULL    |       |
| B_ID         | varchar(50)  | YES  | MUL | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
6 rows in set (0.00 sec)

mysql> desc location;
+------------------+--------------+------+-----+---------+-------+
| Field            | Type         | Null | Key | Default | Extra |
+------------------+--------------+------+-----+---------+-------+
| LOC_ID           | varchar(50)  | NO   | PRI | NULL    |       |
| L_Name           | varchar(100) | YES  |     | NULL    |       |
| Market_potential | varchar(50)  | YES  |     | NULL    |       |
| Region           | varchar(100) | YES  |     | NULL    |       |
| V_ID             | varchar(50)  | YES  | MUL | NULL    |       |
+------------------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

```
mysql> desc partnership;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| P_ID       | varchar(50) | NO   | PRI | NULL    |       |
| P_Name     | varchar(100)| YES  |     | NULL    |       |
| P_Type     | varchar(50) | YES  |     | NULL    |       |
| P_Industry | varchar(50) | YES  |     | NULL    |       |
| B_ID       | varchar(50) | YES  | MUL | NULL    |       |
+------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> desc trends;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| T_Type       | varchar(50) | YES  |     | NULL    |       |
| Duration     | int         | YES  |     | NULL    |       |
| Impact_level | varchar(50) | YES  |     | NULL    |       |
| A_ID         | varchar(50) | YES  | MUL | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc vendor_supplier;
+---------+--------------+------+-----+---------+-------+
| Field   | Type         | Null | Key | Default | Extra |
+---------+--------------+------+-----+---------+-------+
| V_ID    | varchar(50)  | NO   | PRI | NULL    |       |
| V_Name  | varchar(100) | YES  |     | NULL    |       |
| V_Type  | varchar(50)  | YES  |     | NULL    |       |
| Budget  | decimal(15,2)| YES  |     | NULL    |       |
| Quality | varchar(50)  | YES  |     | NULL    |       |
| V_loc   | varchar(100) | YES  |     | NULL    |       |
| B_ID    | varchar(50)  | YES  | MUL | NULL    |       |
+---------+--------------+------+-----+---------+-------+
7 rows in set (0.00 sec)
```

CRUD OPERATIONS

- INSERT (insertion of entries into all tables)

```
-- Sample Data Insertion
INSERT INTO BUSINESS (B_ID, B_Name, L_Name, F_Name, B_Type, OO_Mail, Phone) VALUES
    ('FC_1', 'Clothes Co.', 'Smith', 'John', 'Fashion', 'clothesco@mail.com', '1234567890'),
    ('FS_1', 'Shoes Inc.', 'Doe', 'Jane', 'Fashion', 'shoesinc@mail.com', '0987654321'),
    ('FA_1', 'Accessories Hub', 'Brown', 'Alice', 'Fashion', 'accessories@mail.com', '1122334455'),
    ('TR_1', 'Travel Co 1', 'Taylor', 'Bob', 'Travel', 'travel1@mail.com', '1223334444'),
    ('TR_2', 'Travel Co 2', 'Clark', 'Emma', 'Travel', 'travel2@mail.com', '1333444555'),
    ('TR_3', 'Travel Co 3', 'Jones', 'Harry', 'Travel', 'travel3@mail.com', '1444555666'),
    ('SP_1', 'Sports Goods 1', 'White', 'James', 'Sports', 'sports1@mail.com', '1555666777'),
    ('SP_2', 'Sports Goods 2', 'Green', 'Eve', 'Sports', 'sports2@mail.com', '1666777888'),
    ('SP_3', 'Sports Goods 3', 'Black', 'Oscar', 'Sports', 'sports3@mail.com', '1777888999'),
    ('FB_1', 'Beverage Co.', 'Hall', 'Kate', 'Food', 'beverages@mail.com', '1888999000'),
    ('FE_1', 'Baked Goods', 'Scott', 'Lily', 'Food', 'bakedgoods@mail.com', '1999000111'),
    ('FE_2', 'Chocolate World', 'Young', 'Zoe', 'Food', 'chocolates@mail.com', '2111222333'),
    ('FA_2', 'Jewelry House', 'Moore', 'Ella', 'Fashion', 'jewelry@mail.com', '2223334444'),
    ('FS_2', 'Sneaker City', 'Harris', 'Noah', 'Fashion', 'sneakercity@mail.com', '2333444555'),
    ('SP_4', 'Outdoor Gear', 'Martin', 'Leo', 'Sports', 'outdoorgear@mail.com', '2444555666'),
    ('TR_4', 'Adventure Tours', 'Lee', 'Lucas', 'Travel', 'adventure@mail.com', '2555666777'),
    ('FC_2', 'High Fashion', 'Walker', 'Sophie', 'Fashion', 'highfashion@mail.com', '2666777888'),
    ('FB_2', 'Tea Time', 'Allen', 'Ryan', 'Food', 'teatime@mail.com', '2777888999'),
    ('FE_3', 'Cake Boutique', 'Wright', 'Mia', 'Food', 'cake@mail.com', '2888999000'),
    ('SP_5', 'Fitness Plus', 'Hill', 'Jack', 'Sports', 'fitness@mail.com', '2999000101');
```

```
INSERT INTO COMPETITORS (C_ID, C_Name, Industry_type, Prod_Sold, B_ID) VALUES
    ('1', 'ZARA', 'Fashion', 500, 'FC_1'),
    ('2', 'MOCHI', 'Fashion', 300, 'FS_1'),
    ('3', 'CLAIRES', 'Fashion', 450, 'FA_1'),
    ('4', 'TRIVAGO', 'Travel', 200, 'TR_1'),
    ('5', 'MAKE MY TRIP', 'Travel', 350, 'TR_2'),
    ('6', 'RED BUS', 'Travel', 400, 'TR_3'),
    ('7', 'DECATHLON', 'Sports', 250, 'SP_1'),
    ('8', 'ASICS', 'Sports', 600, 'SP_2'),
    ('9', 'SPEEDO', 'Sports', 1000, 'SP_3'),
    ('10', 'FROZEN BOTTLE', 'Food', 700, 'FB_1'),
    ('11', 'DOSA MANE', 'Food', 550, 'FE_1'),
    ('12', 'BURGER KING', 'Food', 800, 'FE_2'),
    ('13', 'LENSKART', 'Fashion', 350, 'FA_2'),
    ('14', 'BATA', 'Fashion', 500, 'FS_2'),
    ('15', 'WILSON', 'Sports', 300, 'SP_4'),
    ('16', 'EASE MY TRIP', 'Travel', 150, 'TR_4'),
    ('17', 'ZUDIO', 'Fashion', 450, 'FC_2'),
    ('18', 'PRIDE OF COWS', 'Food', 600, 'FB_2'),
    ('19', 'TACO BELL', 'Food', 650, 'FE_3'),
    ('20', 'WROGN', 'Sports', 500, 'SP_5');
```

```sql
INSERT INTO ANALYSTS (A_ID, A_Name, Success_rate, Experience, Salary, B_ID) VALUE
    ('A1', 'Alice', 90.00, 5, 60000, 'FC_1'),
    ('A2', 'Bob', 85.00, 7, 70000, 'FS_1'),
    ('A3', 'Charlie', 75.00, 4, 55000, 'FA_1'),
    ('A4', 'David', 92.00, 6, 72000, 'TR_1'),
    ('A5', 'Emma', 88.00, 3, 48000, 'TR_2'),
    ('A6', 'Frank', 80.00, 5, 59000, 'TR_3'),
    ('A7', 'Grace', 78.00, 4, 47000, 'SP_1'),
    ('A8', 'Hannah', 84.00, 3, 53000, 'SP_2'),
    ('A9', 'Ian', 76.00, 2, 42000, 'FE_1'),
    ('A10', 'Jack', 90.00, 5, 61000, 'FB_1');


INSERT INTO TRENDS (T_Type, Duration, Impact_level, A_ID) VALUES
    ('Sustainable Fashion', 12, 'High', 'A1'),
    ('AI in Fashion', 24, 'Medium', 'A2'),
    ('Eco-Friendly Materials', 36, 'High', 'A3'),
    ('Travel Personalization', 12, 'Medium', 'A4'),
    ('Budget Travel', 18, 'Medium', 'A5'),
    ('Luxury Travel', 24, 'High', 'A6'),
    ('Sports Health', 12, 'Medium', 'A7'),
    ('Wearable Technology', 18, 'High', 'A8'),
    ('Eco-Friendly Sports', 12, 'Medium', 'A9'),
    ('Healthy Food Trends', 6, 'High', 'A10');
```

```sql
INSERT INTO INVESTORS (I_ID, I_Name, Industry_pref, Budget, B_ID) VALUES
    ('I1', 'ARUSHI KATTA', 'Fashion', 1000000, 'FC_1'),
    ('I2', 'AMISHA JAIN', 'Fashion', 1500000, 'FS_1'),
    ('I3', 'ARCHISHA JANAWADE', 'Fashion', 2000000, 'FA_1'),
    ('I4', 'ARITRO MAITI', 'Travel', 2500000, 'TR_1'),
    ('I5', 'ADITYA TIWARI', 'Travel', 3000000, 'TR_2'),
    ('I6', 'BHASWATI CHOWDHARY', 'Travel', 3500000, 'TR_3'),
    ('I7', 'AMRITAA KALANEE', 'Sports', 500000, 'SP_1'),
    ('I8', 'ANNAPOORNESHWARI', 'Food', 600000, 'FE_1'),
    ('I9', 'ANJANA GANESH', 'Sports', 700000, 'SP_2'),
    ('I10', 'MS NAGASUNDARI', 'Food', 800000, 'FB_1');

INSERT INTO CONTRACTS (Con_ID, Con_Type, Validity_period, B_ID, I_ID) VALUES
    ('C1', 'Supply', 12, 'FC_1', 'I1'),
    ('C2', 'Partnership', 24, 'FS_1', 'I2'),
    ('C3', 'Collaboration', 36, 'FA_1', 'I3'),
    ('C4', 'Investment', 12, 'TR_1', 'I4'),
    ('C5', 'Joint Venture', 18, 'TR_2', 'I5'),
    ('C6', 'Merger', 24, 'TR_3', 'I6'),
    ('C7', 'Endorsement', 12, 'FE_1', 'I7'),
    ('C8', 'Sponsorship', 18, 'SP_1', 'I8'),
    ('C9', 'Licensing', 12, 'SP_2', 'I9'),
    ('C10', 'Franchise', 6, 'FB_1', 'I10'),
    ('C11', 'Endorsement', 12, 'SP_3', 'I7');
```

```sql
INSERT INTO LEGAL_ADVISORY (L_ID, Adv_Name, L_Experience, Jurisdiction, Con_ID, B_ID) VALUES
    ('L001', 'Brown & Associates', 15, 'International', 'C3', 'FA_1'),
    ('L002', 'Elite Legal Services', 10, 'Domestic', 'C5', 'FA_2'),
    ('L003', 'Global Law Partners', 20, 'International', 'C10', 'FB_1'),
    ('L004', 'Tea Leaf Legal', 8, 'Domestic', 'C7', 'FB_2'),
    ('L005', 'Fashion Law Hub', 12, 'International', 'C1', 'FC_1'),
    ('L006', 'The High Legal Group', 18, 'Domestic', 'C9', 'FC_2'),
    ('L007', 'Bakehouse Legal', 9, 'Domestic', 'C7', 'FE_1'),
    ('L008', 'ChocoLegal Services', 14, 'Domestic', 'C8', 'FE_2'),
    ('L009', 'Cake Counsel', 7, 'Domestic', 'C8', 'FE_3'),
    ('L010', 'Footwear Legal Advisors', 10, 'International', 'C2', 'FS_1'),
    ('L011', 'Sneaker Legal Network', 13, 'Domestic', 'C9', 'FS_2'),
    ('L012', 'Sports Legal Partners', 16, 'International', 'C8', 'SP_1'),
    ('L013', 'Green Legal Consultancy', 11, 'Domestic', 'C9', 'SP_2'),
    ('L014', 'Black & Co. Sports Law', 20, 'International', 'C11', 'SP_3'),
    ('L015', 'Outdoor Legal Support', 9, 'Domestic', 'C8', 'SP_4'),
    ('L016', 'Fitness Legal Advisors', 12, 'Domestic', 'C8', 'SP_5'),
    ('L017', 'Travel Counsel Partners', 15, 'International', 'C4', 'TR_1'),
    ('L018', 'Clark Legal Services', 10, 'Domestic', 'C5', 'TR_2'),
    ('L019', 'Jones & Partners', 17, 'International', 'C6', 'TR_3'),
    ('L020', 'Adventure Law Group', 18, 'Domestic', 'C4', 'TR_4');
```

```sql
INSERT INTO PARTNERSHIP (P_ID, P_Name, P_Type, P_Industry, B_ID)
VALUES
    ('P1', 'Partner1', 'Collaboration', 'Fashion', 'FC_1'),
    ('P2', 'Partner2', 'Sponsorship', 'Fashion', 'FS_1'),
    ('P3', 'Partner3', 'Joint Venture', 'Fashion', 'FA_1'),
    ('P4', 'Partner4', 'Collaboration', 'Travel', 'TR_1'),
    ('P5', 'Partner5', 'Sponsorship', 'Travel', 'TR_2'),
    ('P6', 'Partner6', 'Joint Venture', 'Travel', 'TR_3'),
    ('P7', 'Partner7', 'Collaboration', 'Sports', 'SP_1'),
    ('P8', 'Partner8', 'Sponsorship', 'Sports', 'SP_2'),
    ('P9', 'Partner9', 'Joint Venture', 'Sports', 'SP_3'),
    ('P10', 'Partner10', 'Collaboration', 'Food', 'FB_1'),
    ('P11', 'Partner11', 'Sponsorship', 'Food', 'FE_1'),
    ('P12', 'Partner12', 'Joint Venture', 'Food', 'FE_2'),
    ('P13', 'Partner13', 'Collaboration', 'Fashion', 'FA_2'),
    ('P14', 'Partner14', 'Sponsorship', 'Fashion', 'FS_2'),
    ('P15', 'Partner15', 'Joint Venture', 'Sports', 'SP_4'),
    ('P16', 'Partner16', 'Collaboration', 'Travel', 'TR_4'),
    ('P17', 'Partner17', 'Sponsorship', 'Fashion', 'FC_2'),
    ('P18', 'Partner18', 'Joint Venture', 'Food', 'FB_2'),
    ('P19', 'Partner19', 'Collaboration', 'Food', 'FE_3'),
    ('P20', 'Partner20', 'Sponsorship', 'Sports', 'SP_5');
```

```sql
INSERT INTO VENDOR_SUPPLIER (V_ID, V_Name, V_Type, Budget, Quality, V_loc, B_ID)
VALUES
    ('VS1_', 'Vendor1', 'Supplier', 50000.00, 'High', 'Mumbai', 'FC_1'),
    ('VS2_', 'Vendor2', 'Distributor', 60000.00, 'Medium', 'Delhi', 'FS_1'),
    ('VS3_', 'Vendor3', 'Retailer', 75000.00, 'High', 'Bangalore', 'FA_1'),
    ('VS4_', 'Vendor4', 'Supplier', 40000.00, 'Low', 'Mumbai', 'TR_1'),
    ('VS5_', 'Vendor5', 'Distributor', 55000.00, 'Medium', 'Delhi', 'TR_2'),
    ('VS6_', 'Vendor6', 'Retailer', 63000.00, 'High', 'Bangalore', 'TR_3'),
    ('VS7_', 'Vendor7', 'Supplier', 80000.00, 'High', 'Mumbai', 'FE_1'),
    ('VS8_', 'Vendor8', 'Distributor', 45000.00, 'Low', 'Delhi', 'SP_1'),
    ('VS9_', 'Vendor9', 'Retailer', 67000.00, 'Medium', 'Bangalore', 'SP_1'),
    ('VS10_', 'Vendor10', 'Supplier', 49000.00, 'Low', 'Mumbai', 'FB_1');

INSERT INTO LOCATION (LOC_ID, L_Name, Market_potential, Region, V_ID)
VALUES
    ('LOC1', 'Mumbai Central', 'High', 'West', 'VS1_'),
    ('LOC2', 'Delhi North', 'Medium', 'North', 'VS2_'),
    ('LOC3', 'Bangalore East', 'High', 'South', 'VS3_'),
    ('LOC4', 'Hyderabad South', 'Low', 'South', 'VS4_'),
    ('LOC5', 'Chennai West', 'Medium', 'South', 'VS5_'),
    ('LOC6', 'Kolkata East', 'High', 'East', 'VS6_'),
    ('LOC7', 'Pune West', 'High', 'West', 'VS7_'),
    ('LOC8', 'Ahmedabad Central', 'Low', 'West', 'VS8_'),
    ('LOC9', 'Jaipur North', 'Medium', 'North', 'VS9_'),
    ('LOC10', 'Lucknow Central', 'Low', 'North', 'VS10_');
```

```sql
INSERT INTO BENEFICIARY (Ben_ID, Ben_Name, Age, DOB, Lease_Term, Mail, Phone, Owner, B_ID)
VALUES
    ('BEN1', 'Rajesh Kumar', 35, '1989-05-12', 24, 'rajesh.kumar@example.com', '9123456780', 'Sunita Kumar', 'FC_1'),
    ('BEN2', 'Priya Sharma', 29, '1995-08-23', 36, 'priya.sharma@example.com', '9123456781', 'Amit Sharma', 'FS_1'),
    ('BEN3', 'Vikram Singh', 42, '1982-02-10', 18, 'vikram.singh@example.com', '9123456782', 'Meena Singh', 'FA_1'),
    ('BEN4', 'Neha Verma', 31, '1993-11-15', 48, 'neha.verma@example.com', '9123456783', 'Rohit Verma', 'TR_1'),
    ('BEN5', 'Arjun Patel', 27, '1997-04-20', 12, 'arjun.patel@example.com', '9123456784', 'Lakshmi Patel', 'SP_1'),
    ('BEN6', 'Aditi Nair', 30, '1994-07-13', 36, 'aditi.nair@example.com', '9123456785', 'Suresh Nair', 'FB_1'),
    ('BEN7', 'Ananya Gupta', 28, '1996-03-17', 24, 'ananya.gupta@example.com', '9123456786', 'Vivek Gupta', 'FE_1'),
    ('BEN8', 'Karan Desai', 33, '1991-09-22', 48, 'karan.desai@example.com', '9123456787', 'Rekha Desai', 'SP_2'),
    ('BEN9', 'Sneha Joshi', 26, '1998-11-30', 12, 'sneha.joshi@example.com', '9123456788', 'Anil Joshi', 'FB_2'),
    ('BEN10', 'Rohan Mehta', 40, '1984-01-05', 24, 'rohan.mehta@example.com', '9123456789', 'Pooja Mehta', 'TR_2');
```

- READ

```
def read(cursor):
    """
    Display business details and related information from the database.

    Args:
        cursor: MySQL database cursor object
    """
    # Fetch the data from the database
    result = view_all_data()

    # Dictionary mapping table names to their column names
    tables = {
        'Business': ['Business ID', 'Business Name', 'Last Name', 'First Name', 'Business Type', 'Official Email', 'Phone Number']
    }

    # Create a selectbox for table selection
    selected_table = st.selectbox("Select table to view:", list(tables.keys()))

    try:
        # Construct and execute query
        query = f"SELECT * FROM `{selected_table}`"
        cursor.execute(query)
        results = cursor.fetchall()

```

- UPDATE

```
# Function to edit a business record
def edit_business_data(new_b_id, new_b_name, new_l_name, new_f_name, new_b_type, new_oo_mail, new_phone,
                       b_id, b_name, l_name, f_name, b_type, oo_mail, phone):
    c.execute('''
        UPDATE BUSINESS
        SET B_ID = %s, B_Name = %s, F_Name = %s, L_Name = %s, B_Type = %s, OO_Mail = %s, Phone = %s
        WHERE B_ID = %s AND B_Name = %s AND L_Name = %s AND F_Name = %s AND B_Type = %s AND OO_Mail = %s AND Phone = %s
    ''', (new_b_id, new_b_name, new_l_name, new_f_name, new_b_type, new_oo_mail, new_phone,
          b_id, b_name, l_name, f_name, b_type, oo_mail, phone))
    mydb.commit()
```

- DELETE

```
# Function to delete a business record by name
def delete_data(b_name):
    c.execute('DELETE FROM BUSINESS WHERE B_Name = %s', (b_name,))
    mydb.commit()
```

PROCEDURES AND TRIGGERS:

```
--TRIGGER for inserting registered business name into competitors table for future users
DELIMITER //

CREATE TRIGGER after_business_insert
AFTER INSERT ON BUSINESS
FOR EACH ROW
BEGIN
    -- Insert into COMPETITORS with default values, linking B_ID from BUSINESS
    INSERT INTO COMPETITORS (C_ID, C_Name, Industry_type, Prod_Sold, B_ID)
    VALUES (CONCAT('C_', NEW.B_ID), NEW.B_Name, NEW.B_Type, 0, NEW.B_ID);
END //

DELIMITER ;
```

```
--procedure to archive trends shorter / equal to an input number
DELIMITER $$

CREATE PROCEDURE ArchiveShortDurationTrends(IN input_duration INT)
BEGIN
    -- Archive trends with a duration less than the input value
    INSERT INTO archivedtrends (T_Type, Duration, Impact_level, A_ID, archived_date)
    SELECT T_Type, Duration, Impact_level, A_ID, NOW()
    FROM trends
    WHERE Duration < input_duration;

    -- Delete the archived trends from the original table
    DELETE FROM trends
    WHERE Duration < input_duration;

END $$

DELIMITER ;
```

```
--PROCEDURE: undo the archive
DELIMITER $$

CREATE PROCEDURE UndoArchiveTrends()
BEGIN
    -- Insert all records back to trends table from archivedtrends
    INSERT INTO trends (T_Type, Duration, Impact_level, A_ID)
    SELECT T_Type, Duration, Impact_level, A_ID
    FROM archivedtrends;

    -- Delete the restored records from archivedtrends
    DELETE FROM archivedtrends;
END $$

DELIMITER ;
```

```
-- Insert stored procedure with trigger integration
DELIMITER //

CREATE PROCEDURE InsertBusiness (
    IN p_B_ID VARCHAR(10),
    IN p_B_Name VARCHAR(50),
    IN p_B_Type VARCHAR(50),
    IN p_OO_Mail VARCHAR(100),
    IN p_Phone VARCHAR(50)
)
BEGIN
    -- Insert into BUSINESS table
    INSERT INTO BUSINESS (B_ID, B_Name, B_Type, OO_Mail, Phone)
    VALUES (p_B_ID, p_B_Name, p_B_Type, p_OO_Mail, p_Phone);

    -- No direct insertion into COMPETITORS needed; trigger will handle it
END //

DELIMITER ;


DELIMITER //
```

```
mysql> show triggers;
+--------------------+--------+-------+-----------+---------+----------------------+--------------------+-----------------------+--------+-----------+---------+----------+
| Trigger            | Event  | Table | Statement | Definer | character_set_client | collation_connection | Database Collation | Timing | Created | sql_mode |
+--------------------+--------+-------+-----------+---------+----------------------+--------------------+-----------------------+--------+-----------+---------+----------+
| after_business_insert | INSERT | business | BEGIN

    INSERT INTO COMPETITORS (C_ID, C_Name, Industry_type, Prod_Sold, B_ID)
    VALUES (CONCAT('C_', NEW.B_ID), NEW.B_Name, NEW.B_Type, 0, NEW.B_ID);
END | AFTER | 2024-11-08 09:08:40.60 | ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION | root@localhost | cp850 | cp850_general_ci | utf8mb4_0900_ai_ci |
+--------------------+--------+-------+-----------+---------+----------------------+--------------------+-----------------------+--------+-----------+---------+----------+
1 row in set (0.04 sec)
```

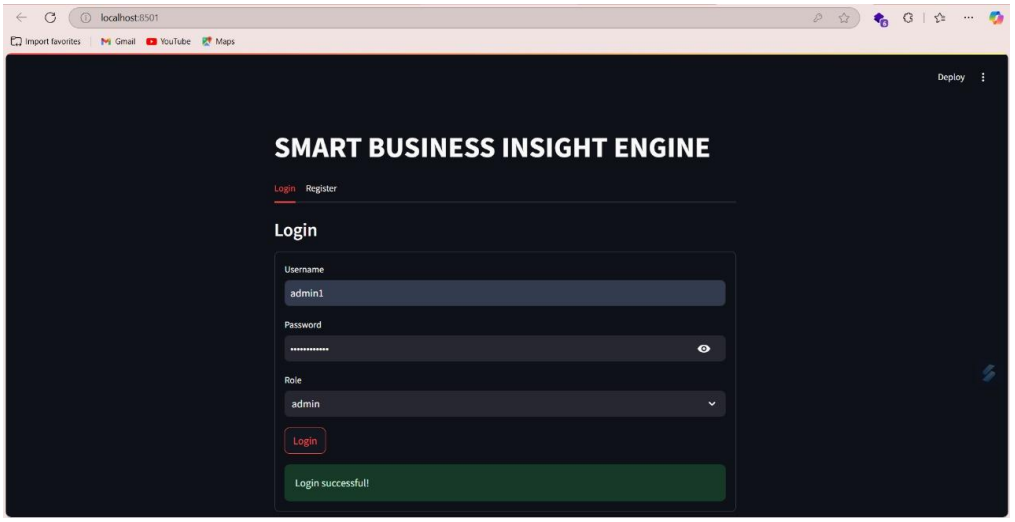
```
mysql> show procedure status where db="sbie";
+------+-----------------------------+-----------+-------------+---------------------+---------------------+---------------+---------+----------------------+----------------------+-----------------------+
| Db   | Name                        | Type      | Definer     | Modified            | Created             | Security_type | Comment | character_set_client | collation_connection | Database Collation    |
+------+-----------------------------+-----------+-------------+---------------------+---------------------+---------------+---------+----------------------+----------------------+-----------------------+
| sbie | ArchiveShortDurationTrends  | PROCEDURE | root@localhost | 2024-11-15 18:56:11 | 2024-11-15 18:56:11 | DEFINER       |         | cp850                | cp850_general_ci     | utf8mb4_0900_ai_ci    |
| sbie | GetBusinessDetails          | PROCEDURE | root@localhost | 2024-11-08 09:08:40 | 2024-11-08 09:08:40 | DEFINER       |         | cp850                | cp850_general_ci     | utf8mb4_0900_ai_ci    |
| sbie | InsertBusiness              | PROCEDURE | root@localhost | 2024-11-08 09:08:40 | 2024-11-08 09:08:40 | DEFINER       |         | cp850                | cp850_general_ci     | utf8mb4_0900_ai_ci    |
| sbie | UndoArchiveTrends           | PROCEDURE | root@localhost | 2024-11-15 19:18:19 | 2024-11-15 19:18:19 | DEFINER       |         | cp850                | cp850_general_ci     | utf8mb4_0900_ai_ci    |
+------+-----------------------------+-----------+-------------+---------------------+---------------------+---------------+---------+----------------------+----------------------+-----------------------+
4 rows in set (0.01 sec)
```

```python

    def perform_archive(duration):
        try:
            mydb = init_connection()
            cursor = mydb.cursor(dictionary=True)

            # Check for eligible trends
            cursor.execute("SELECT COUNT(*) as count FROM trends WHERE Duration <= %s", (duration,))
            count = cursor.fetchone()['count']

            if count == 0:
                st.session_state.archive_status = "no_trends"
                cursor.close()
                mydb.close()
                return

            # Execute archive procedure
            cursor.callproc("ArchiveShortDurationTrends", [duration])
            mydb.commit()

            # Fetch archived trends to display
            cursor.execute("SELECT * FROM archivedtrends ORDER BY Duration ASC")
            st.session_state.archived_trends = cursor.fetchall()

            st.session_state.archive_status = "success"
            cursor.close()
            mydb.close()
        except Exception as e:
            st.session_state.archive_status = f"error: {str(e)}"

```

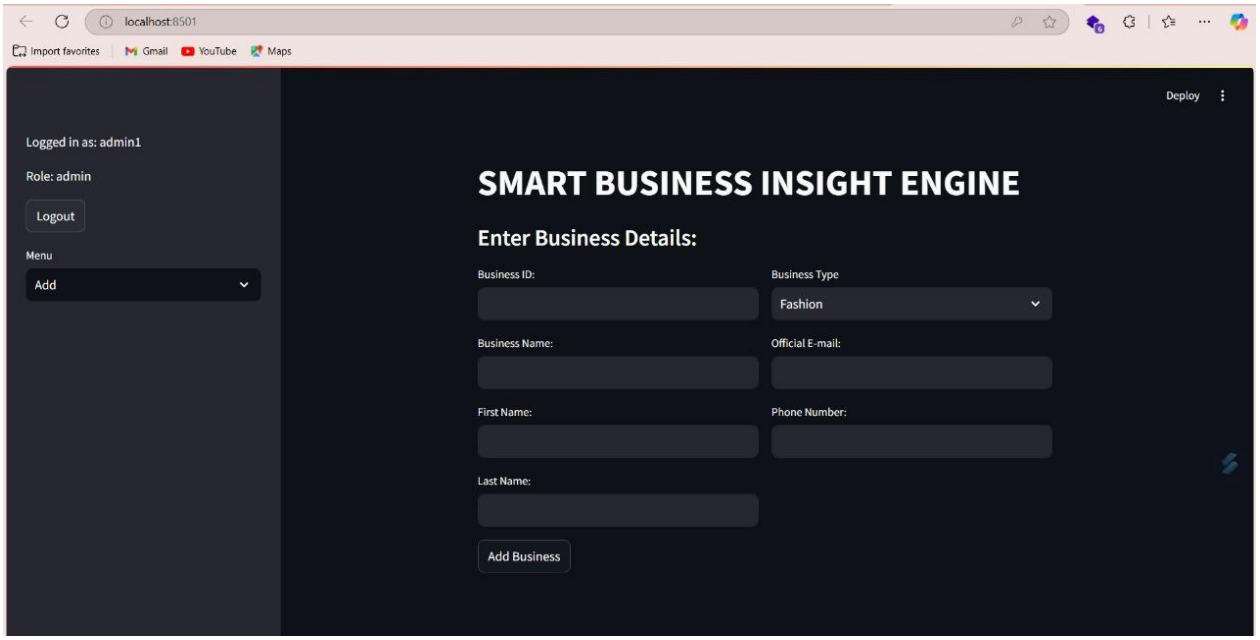
```python
    # Handle undo operation
    if st.session_state.undo_clicked:
        try:
            mydb = init_connection()
            cursor = mydb.cursor()
            cursor.callproc("UndoArchiveTrends")
            mydb.commit()
            st.session_state.archive_status = "undo_success"
            st.session_state.archived_trends = None  # Reset archived trends on undo
            cursor.close()
            mydb.close()
        except Exception as e:
            st.session_state.archive_status = f"undo_error: {str(e)}"
        st.session_state.undo_clicked = False

    # Display archive status
    if st.session_state.archive_status:
        if st.session_state.archive_status == "success":
            st.success("Successfully archived trends!")
        elif st.session_state.archive_status == "no_trends":
            st.warning("No trends found with the specified duration.")
        elif st.session_state.archive_status == "undo_success":
            st.success("Successfully undid archive!")
        elif "error" in st.session_state.archive_status:
            st.error(f"Error: {st.session_state.archive_status}")

        # Reset status for better experience
        st.session_state.archive_status = None

    # Display archived trends
    if st.session_state.archived_trends:
        st.subheader("Archived Trends")
        df_archived = pd.DataFrame(st.session_state.archived_trends)
        st.dataframe(df_archived)

```

FUNCTIONALITY AND FEATURES + FRONTEND

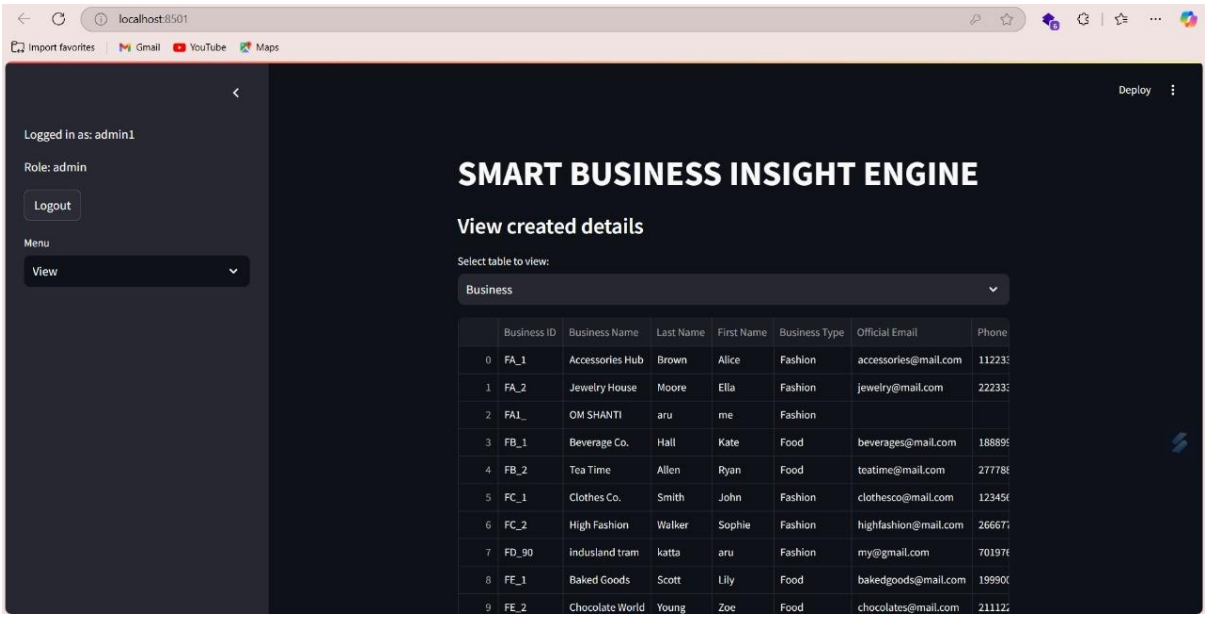
LOGIN PAGE FOR ADMIN:



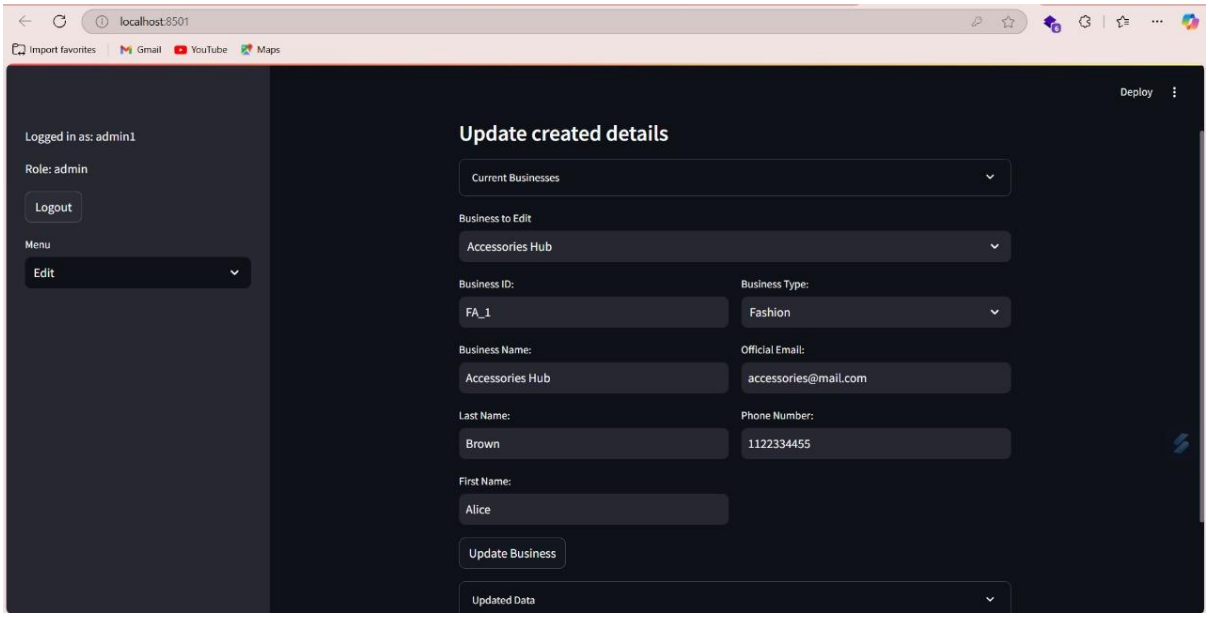

ADD BUSINESS (MAIN DASHBOARD):



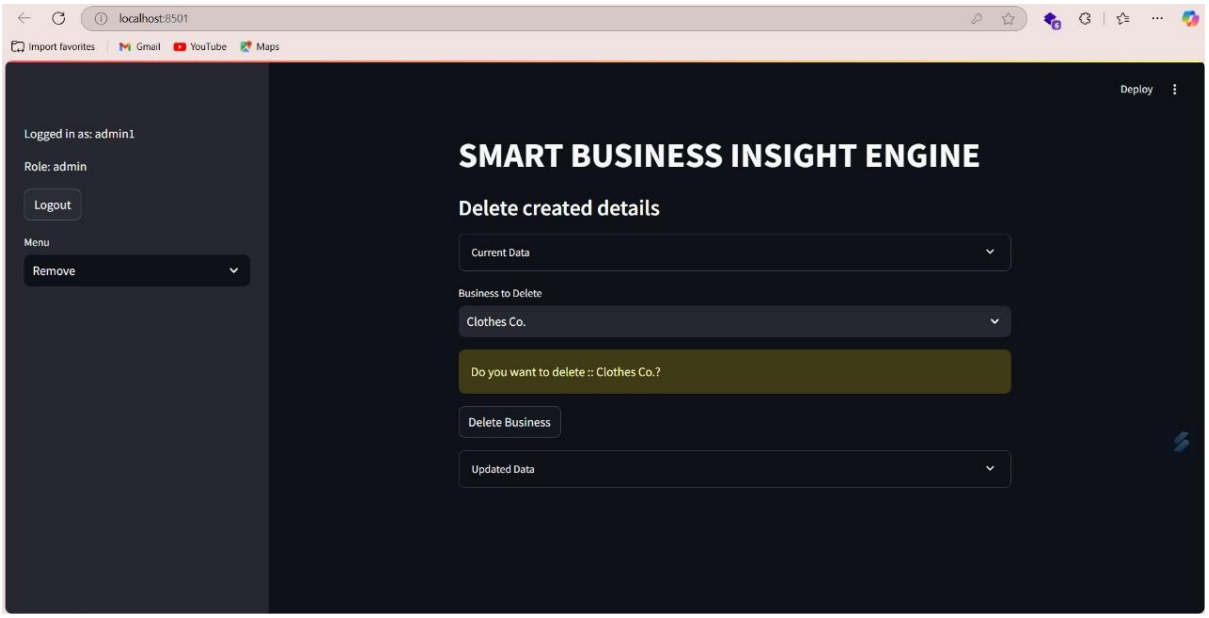
VIEW BUSINESS:



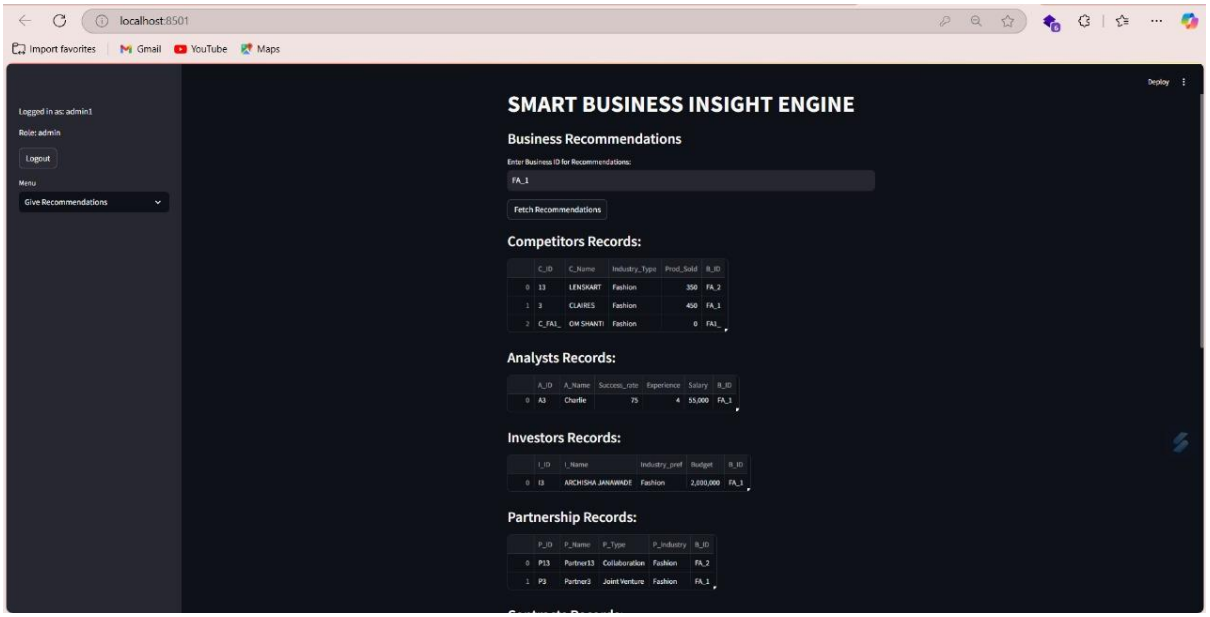

UPDATE BUSINESS:

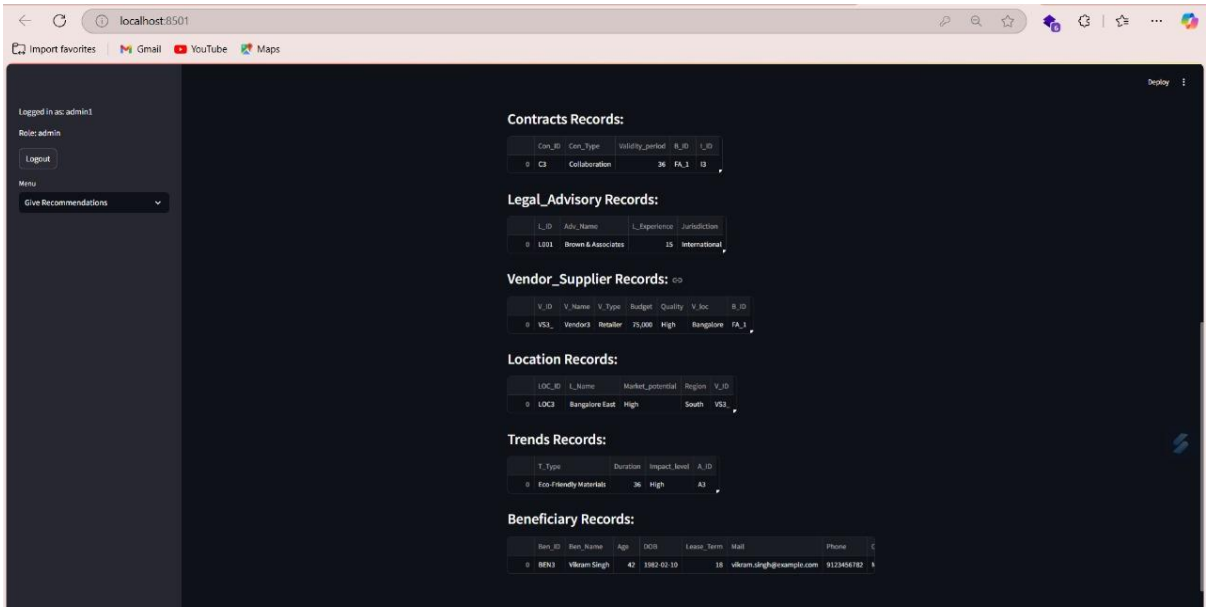


DELETE BUSINESS:

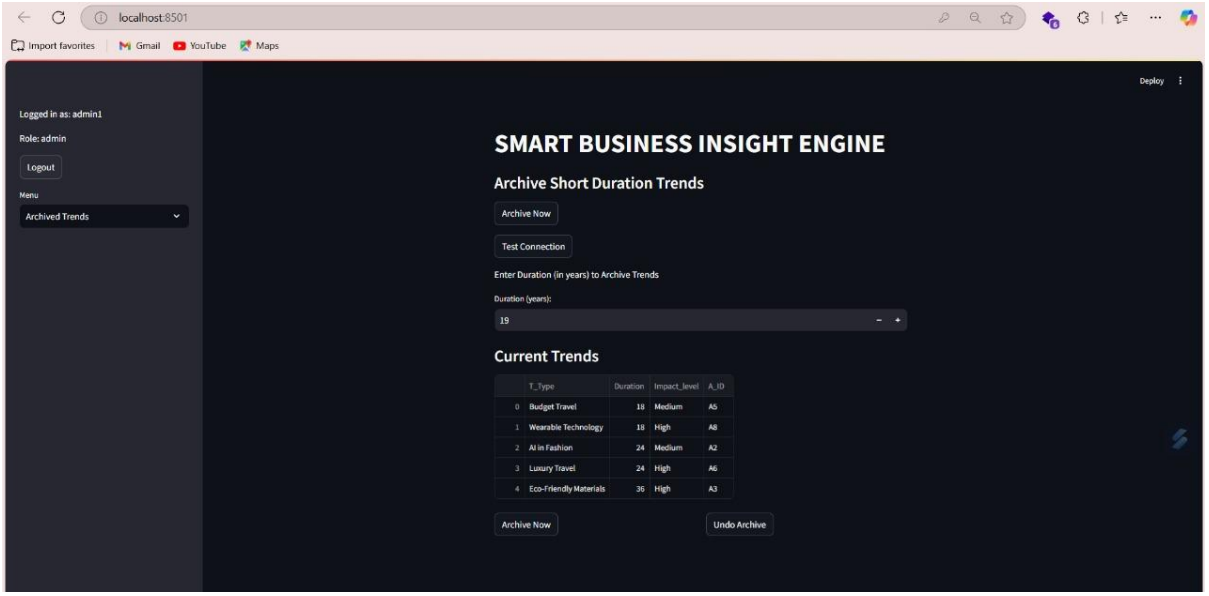


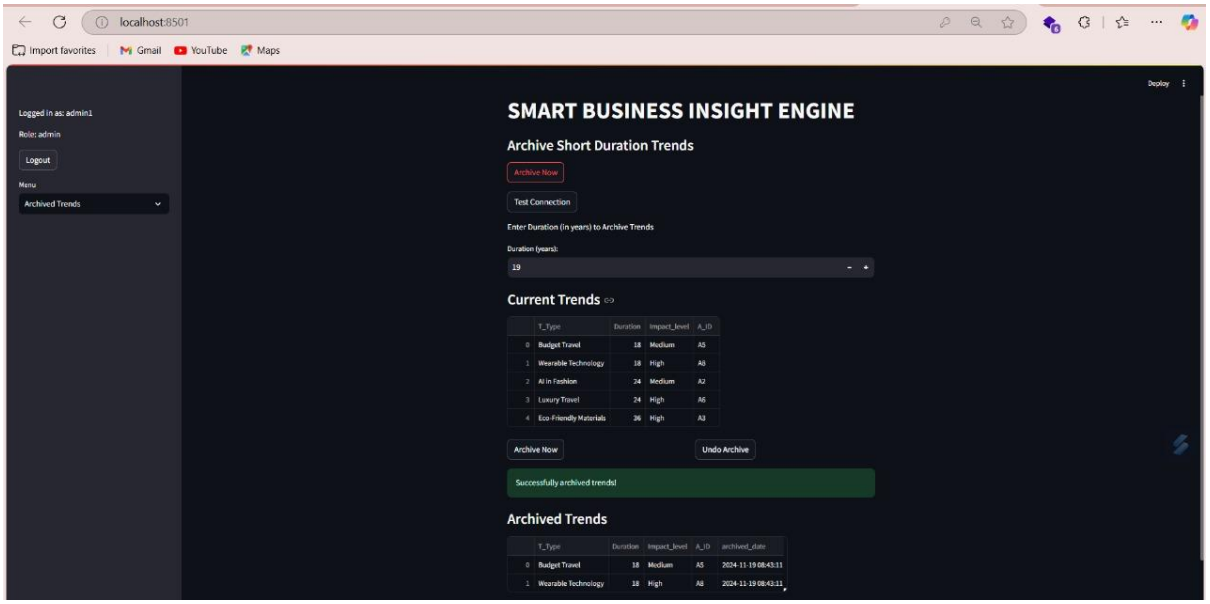

BUSINESS RELATED RECOMMENDATIONS:



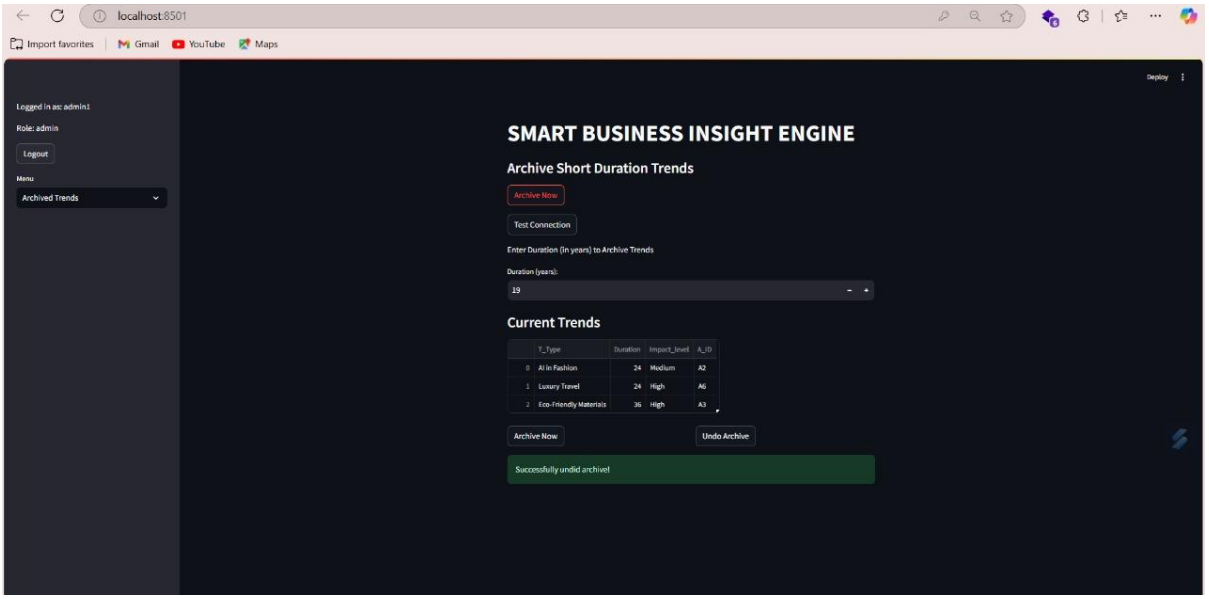



ARCHIVE TRENDS:



UNARCHIVE TRENDS:




AGENT ACCESS DASHBOARD (LIMITED FEATURES):