**Test Question Bank Management System**

## **Architecture Overview**

### **1.2 Backend**

* **Framework:** Django Rest framework
* **Database:** PostgreSQL for storing data securely.
* **Authentication:** JWT-based authentication for secure user access.

### **1.1 Frontend**

* **Technology:** React.js for the user interface.

### **1.3 Database**

* Use PostgreSQL as the relational database management system (RDBMS) for data storage.
* Create tables to store User profiles, Topic details, and Question banks.

1. **Actors and Roles**

**2.1 User**

* **Login and Profile:** User can log in (all users as normal user) and maintain their profiles.
* **View Topics and Question bank**: User can view Topics and Question bank.

**2.2 Admin (Super User)**

* **See and Search all users details:** Admin can see and search users details.
* **Change users type:** Admin can change normal users type ( Owner, Editor, viewer).
* **Block user:** Admin can block user.
* **Topic:** Admin can add topics and also Soft delete (Disable) Topics.

**2.3 Owner**

* **Topic and Question bank details:** Owner can add, update or delete topics and question bank.

**2.4 Editor**

* **Topic and Question bank details**: Editor can view , search and update topics and question bank.