EDTECH ASSIGNMENT TRACKER

Part A: System Design (Written)

1. Core Entities & Relationships

Entity Descriptions:

Entity	Fields
User:	id, username, email, password, role (Student/Teacher)
Assignment:	id, title, description, due_date, created_at, teacher (FK to User)
Submission:	id, assignment (FK), student (FK to User), file/url, submitted_at

- One teacher can create many assignments
- One assignment can have many submissions (from students)
- One student can submit only one submission per assignment

ER Diagram (Text-Based)

User
- ├── id (PK)
- ├── username
- ├── email
- ├── password
- └── role [Teacher | Student]

Assignment
- ├── id (PK)
- ├── title
- ├── description
- ├── due_date
- ├── created_at
- └── teacher_id (FK -> User)

Submission

├── id (PK)

├── assignment_id (FK -> Assignment)

├── student_id (FK -> User)

├── file_url / uploaded_file

└── submitted_at

2. API Endpoints

Authentication (JWT-based)

Endpoint	Method	Description
/api/register/	POST	Register student/teacher
/api/login/	POST	Login and get JWT token

Assignment APIs (Teacher Only)

Endpoint	Method	Description
/api/assignments/	POST	Create new assignment
/api/assignments/	GET	List all assignments
/api/assignments/<id>/	GET	View assignment detail
/api/assignments/<id>/submissions/	GET	View submissions for an assignment

Submission APIs (Student Only)

Endpoint	Method	Description
/api/assignments/<id>/submit/	POST	Submit assignment file/URL
/api/my-submissions/	GET	View student's own submissions

Suggestions for Scaling the System

Area	Suggestions
Database	Use PostgreSQL for production
Caching	Cache frequent queries with Redis
Pagination	Add pagination for assignments/submissions
Asynchronous Task	Use Celery for email notifications, deadline reminders