Full-Stack Todo Application (Spring Boot + React)

📌 Overview

This project is a simple Full-Stack Todo Application developed as a screening assignment. The application allows users to create, view, update, complete, and delete tasks using a RESTful backend and a React frontend.

The system follows clean layered architecture: - Controller → Handles HTTP requests - Service → Business logic - Repository → Database operations - React UI → Consumes REST APIs and manages UI state

---

🧰 Tech Stack

Backend

- Java 17+
- Spring Boot
- Spring Data JPA
- H2 In-Memory Database
- REST API
- Bean Validation

Frontend

- React (Functional Components + Hooks)
- Axios for API communication

🗜️ Features

- Create Todo (title mandatory)
- View all todos
- Mark todo as completed / incomplete
- Update todo details
- Delete todo
- Filter todos by completion status (API support)

🗂️ Project Structure

```
todo-fullstack/
 ├── backend/
 │    ├── controller
 │    ├── service
 │    ├── repository
```

```
│   ├── model
│   └── exception
│
└── frontend/
    └── src/App.js
```

⚙️ Backend Setup

1. Navigate to backend folder

```
cd backend
```

2. Run application

```
mvn spring-boot:run
```

Backend will run at:

```
http://localhost:8080
```

H2 Database Console

```
http://localhost:8080/h2-console
```

JDBC URL:

```
jdbc:h2:mem:tododb
```

💻 Frontend Setup

1. Navigate to frontend folder

```
cd frontend
```

2. Install dependencies

```
npm install
```

3. Start application

```
npm start
```

Frontend will run at:

```
http://localhost:3000
```

🔗 API Endpoints

Create Todo

```
POST /api/todos
```

Get All Todos

```
GET /api/todos
GET /api/todos?completed=true
```

Update Todo

```
PUT /api/todos/{id}
```

Delete Todo

```
DELETE /api/todos/{id}
```

🧪 Validation Rules

- Title is mandatory
- Returns proper HTTP status codes
- 404 returned if todo not found

📈 Future Improvements

- Authentication (JWT Login)
- Pagination & Sorting
- Docker Deployment
- Persistent database (PostgreSQL/MySQL)

- Responsive UI design

💻 Author

Developed as part of a Full-Stack Screening Assignment.