

Timesheet Application - Technical Specification

1. Project Overview

The goal of this project is to develop a simple internal web application for daily time tracking. The application will allow users to enter their name, date, hours worked, project name, and task description. All data will be stored in a PostgreSQL database and later exported to Excel when needed.

2. Frontend Specification (ReactJS)

- Single-page application built with ReactJS.
- One main form containing:
 - Name selection (radio buttons).
 - Date input field.
 - Hours input field (text or number).
 - Project input field (text).
 - Description input field (text or textarea).
- Submit button to send data to backend API.
- Display success message after successful submission.
- No authentication required (internal use only).

3. Backend Specification (Python)

- Backend developed using Python (FastAPI or Flask recommended).
- Expose REST API endpoint: POST /timesheet
- API receives: name, date, hours, project, description.
- Insert submitted data into PostgreSQL database.
- Return JSON response confirming successful insert.
- Optional: GET /export endpoint to export data as CSV file.

4. Database Specification (PostgreSQL)

- Database: PostgreSQL

- Table name: timesheets
- Columns:
 - - id (SERIAL PRIMARY KEY)
 - - name (TEXT NOT NULL)
 - - date (DATE NOT NULL)
 - - hours (TEXT NOT NULL)
 - - project (TEXT NOT NULL)
 - - description (TEXT NOT NULL)
 - - created_at (TIMESTAMP DEFAULT NOW())

5. Deployment

- Frontend deployed on Vercel.
- Backend deployed on a Python-compatible hosting service.
- PostgreSQL hosted via managed database provider.
- Environment variables used for database connection security.