project

This document explains the specifications of the "Eventure" product, which is an event organization software. It is an integral part of your curriculum, designed to provide a practical experience in full-stack software development. This project is structured to reinforce and apply the concepts learned in your coursework to a real-world-like scenario.

# The Eventure product

## Background

In the dynamic business landscape, professionals face challenges in efficiently organizing and managing various events. Traditional tools are often fragmented, leading to inefficiencies and communication gaps.

## The Solution

"Eventure" a unified event management platform. Designed to address the complexities of organizing professional events, Eventure offers a comprehensive suite of tools that integrates task management, participant engagement, scheduling, and real-time updates, all within a user-friendly interface.

## Key Features

Here are the main features of the product:

* Interactive Dashboard: For centralizing event management tasks.
* Efficient Task Management: Simplifying coordination among team members.
* Integrated Scheduling: Streamlining date selection and event planning.
* Real-Time Communication: Enhancing engagement between organizers and attendees.

## Technical Strengths

The platform is built on a robust full-stack framework, combining a powerful Java and NoSQL backend with a dynamic front-end. Security is a top priority, with OpenID Connect ensuring secure user authentication.

# Specifications

## Global architecture

### Front End

Use of React to develop a rich and interactive user interface.

Integration of JavaScript or TypeScript for dynamic functionalities like real-time data updates, animations, etc.

### Back End

Development of a RESTful API in Java to handle business logic, data access, and user authentication.

Use of a NoSQL database (like MongoDB) for flexible and scalable data storage.

### API Management, Authentication and Security

Implementation of user authentication using OpenID Connect for secure login.

Integration of API management strategies to secure communication between the front end and the back end.

### Architecture diagram

Une image contenant texte, diagramme, Plan, capture d’écran

Description générée automatiquement

***Fig****. architecture diagram*

Then the ER diagram for the entities:

Une image contenant texte, diagramme, Plan, ligne

Description générée automatiquement

## Features and constraints

### Home Page

Landing page showcasing the platform's purpose and key features.

**Features**

* Login/Signup buttons.
* Brief intro to the platform.
* Navigation bar to access other pages.

### User Dashboard

Personalized dashboard for each user after login.

**Features**

* Overview of upcoming events.
* Quick access to create a new event.
* Notifications and recent activity summary.

### Event Creation Page

Page for creating a new event.

**Features:**

* Form to input event details (name, date, location, description).
* Options to add tasks, assign roles, and invite participants.
* Integration with calendar for date selection.

### Event Detail Page

Detailed view of an individual event.

**Features:**

* Overview of event information.
* Editable sections for agenda, participants, and tasks.
* Real-time updates and collaborative editing capabilities.
* Chat or messaging feature for participant communication.

### Profile Management

User profile settings and personalization.

**Features:**

* Editable user information (name, contact, preferences).
* Option to change password and manage account settings.
* View past events and activities.

### Search and Filter Functionality

Facilitates searching and filtering events and participants.

**Features**:

* Search bar for quick access to events or users.
* Filter options based on date, type, and role in the event.

### Administrative Panel (for Admin Users)

Specialized dashboard for administrative tasks.

**Features**:

* User management (add/remove users, reset passwords).
* Event oversight (view all events, edit or delete as necessary).