

Narrative Query:

Give five functional and non-functional requirements for a web app that helps sports enthusiasts play outdoor games with each other by finding a sports field. Users of this app can be anyone who is 18 or older with a user account. Users can see the leaderboard with every user's ranking on the platform. The web app also has a notifications feature (notifying users of game creation and message requests) and a chat feature for chatting with each other and chatting in a group chat. Use technology stack as ReactJS for frontend, Spring Boot, and PostgreSQL for the backend.

Response by ChatGPT:

Functional requirements:

- Location-based search: The web app should allow users to search for sports fields near their current location or any other specified location using ReactJS.
- Game scheduling: The web app should allow users to schedule games on the platform and invite other users to join the game using Spring Boot.
- User profile management: The web app should allow users to manage their profiles, including updating their personal information, profile picture, and other relevant details using PostgreSQL.
- Game leaderboard: The web app should display a leaderboard that shows the ranking of every user on the platform based on their performance in the games using ReactJS.
- Chat feature: The web app should have a chat feature that allows users to communicate with each other individually or in a group chat using ReactJS.

Non-functional requirements:

- Performance: The web app should be fast and responsive, ensuring that users can quickly find sports fields, schedule games, and chat with each other using ReactJS and Spring Boot.

- Security: The web app should ensure that users' personal information is secure and protected from unauthorized access or data breaches using Spring Boot and PostgreSQL.
- Scalability: The web app should be designed to handle a large number of users and a high volume of traffic, ensuring that the platform can grow over time using Spring Boot and PostgreSQL.
- Usability: The web app should be easy to use and intuitive, allowing users to navigate the platform without any difficulty using ReactJS.
- Compatibility: The web app should be compatible with a range of devices, including desktops, laptops, tablets, and mobile phones, ensuring that users can access the platform from any device they choose using ReactJS.