Scenario Description

SAP Work Zone Product Management comes up with the idea for a new feature that allows a company administrator to assign:

- users to roles and
- roles to applications.

Based on the role, users can access and launch applications.
Users and applications can belong to multiple roles.

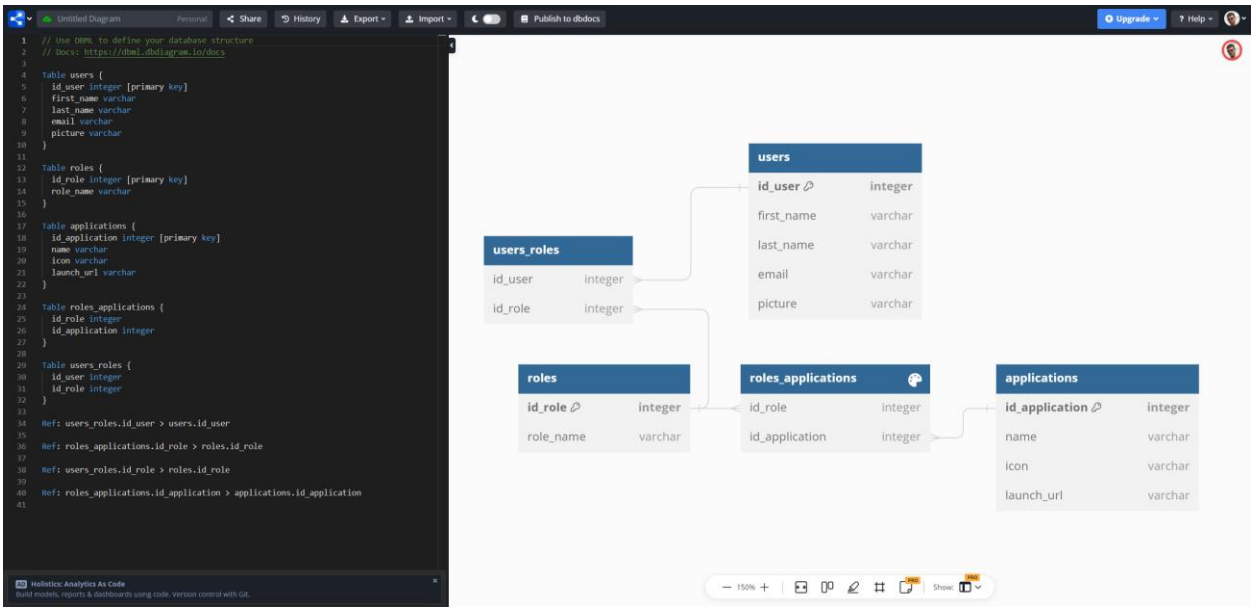
Scenario Based Questions

1) You are given the task of creating a data model for this relation. Please provide an entity relation diagram.

Assume that all entities have a GUID key field. Additionally

- Each application has a name, icon and URL to launch in a new browser tab.
- Users should have a first, last name and picture.
- Roles have only a name.

https://dbdiagram.io/d/667471175a764b3c72031d02



2) You are given the task to create a service APIs for an administrator Web UI to

- List and maintain (add, remove, change) application, user and user role entities.

Please give an example API for one of the entities. What are the necessary database actions that you need to perform.

- **API for User Entity:**
  - **List users:** GET /api/users
  - **Add user:** POST /api/users
  - **Update user:** PUT /api/users/{userId}
  - **Delete user:** DELETE /api/users/{userId}

- **Necessary Database Actions:**
  - **List users:** SELECT query
  - **Add user:** INSERT query
  - **Update user:** UPDATE query
  - **Delete user:** DELETE query

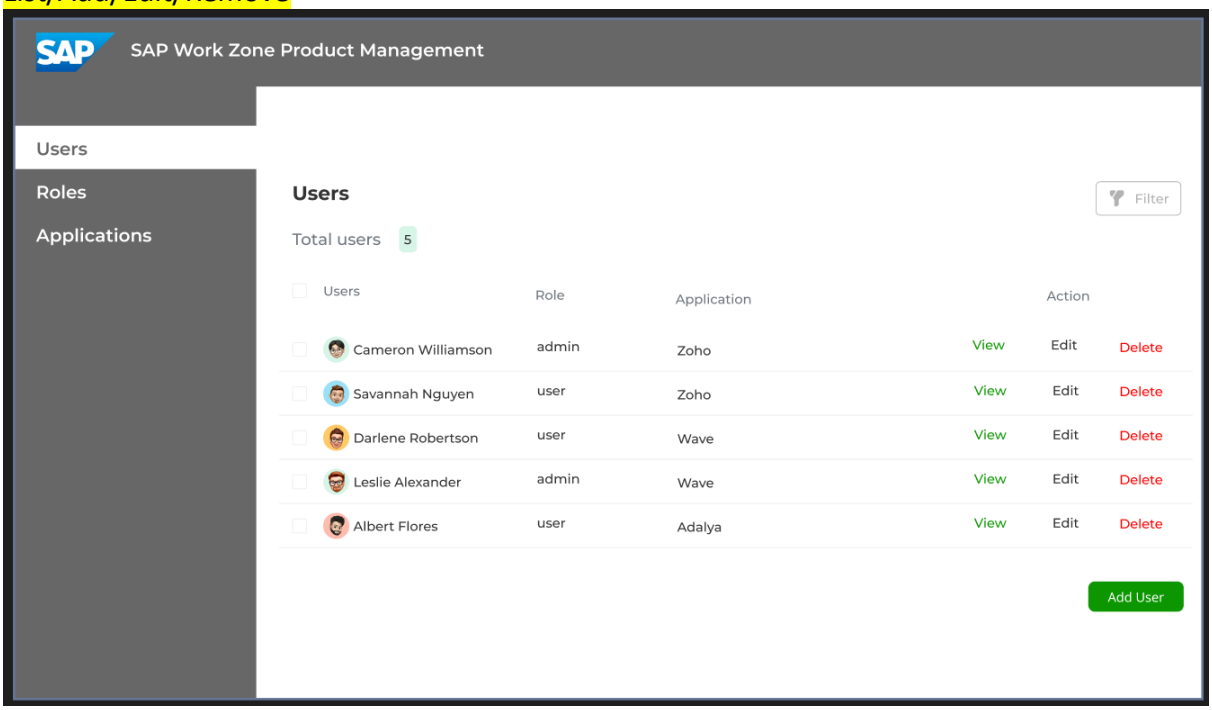
3) You are given the task of creating an administrator Web UI to maintain the above entities. Please create a sketch of the UI and describe the flow to

- Add,
- Remove,
- Change

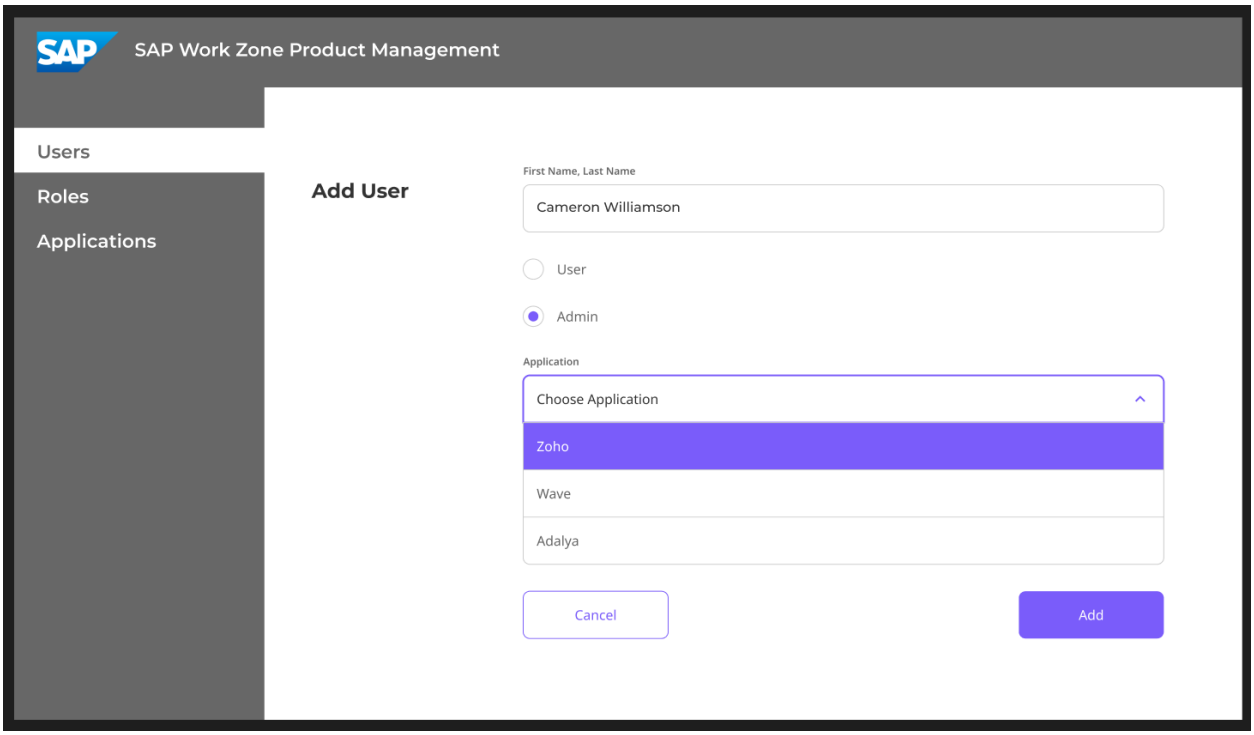
entities.
Consider allowing the administrator to easily switch between the different entities.

https://www.figma.com/design/ZnEEjQuC8c4ecT7j2ohT0W/SAP-Work-Zone-product-management-app?node-id=1-380&t=5q3hRyDzT75XEOpa-1

List/Add/Edit/Remove



Add user

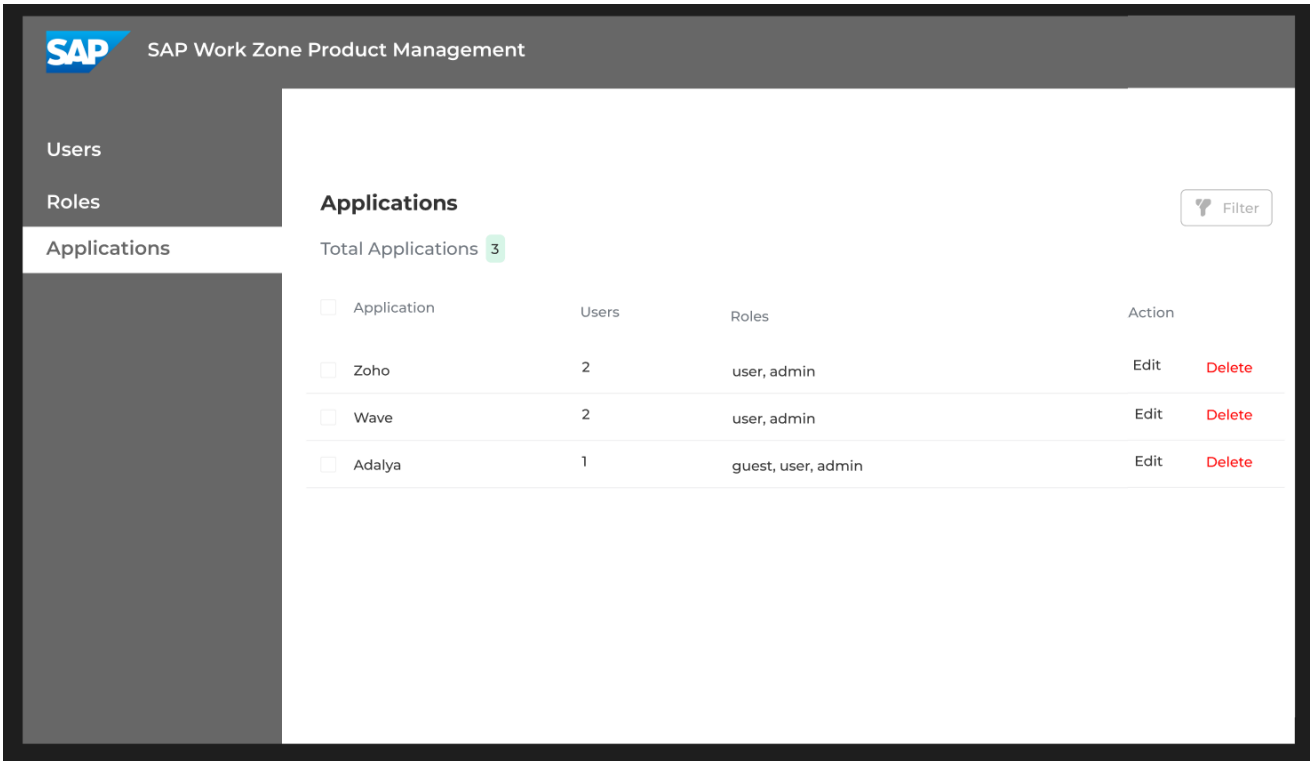


4) You are given the task of creating an end user Web UI to
- list and
- launch

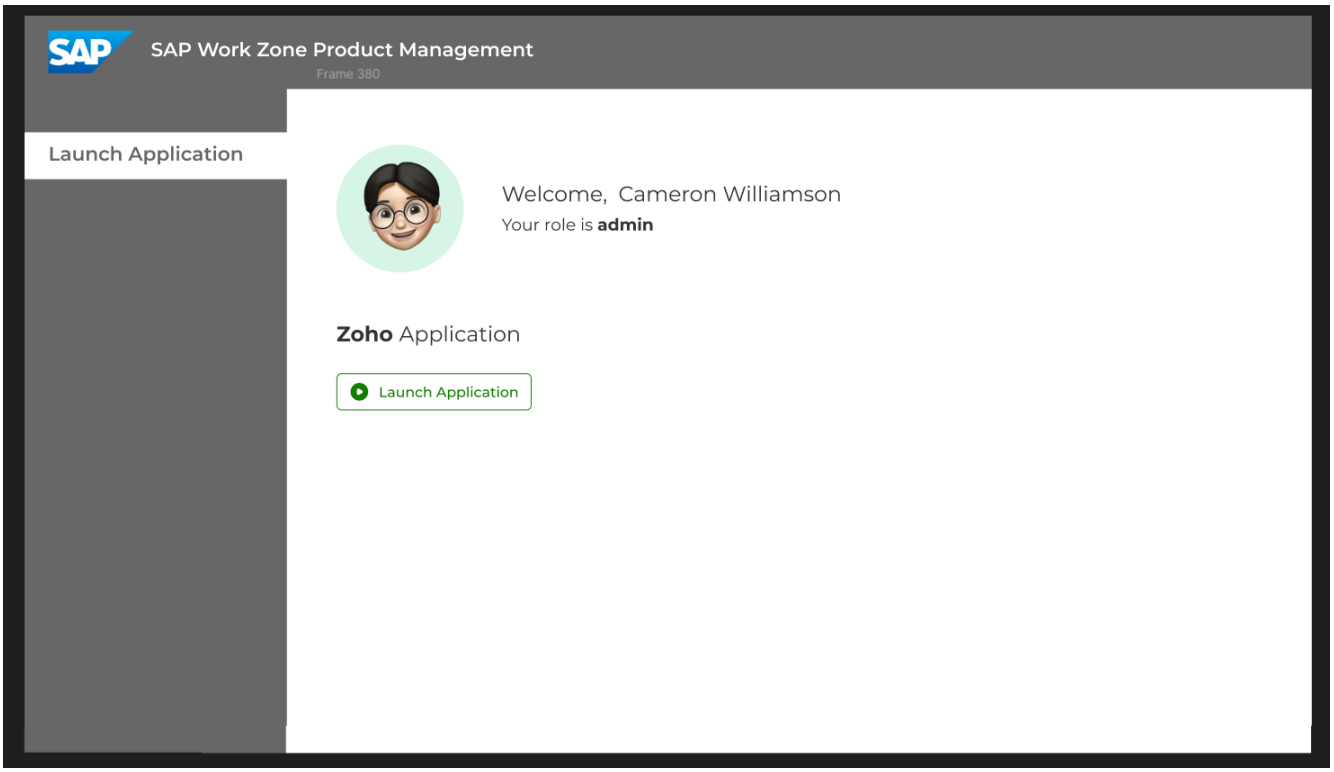
applications. Which service API is needed and how would a UI look like.

https://www.figma.com/design/ZnEEjQuC8c4ecT7j2ohT0W/SAP-Work-Zone-product-management-app?node-id=1-380&t=5q3hRyDzT75XEOpa-1

List applications



Launch Application



5) Which qualities need to be covered for this feature?

Choose the five most important ones from your perspective

- Average API response times < 50ms for read access.
- Average API average response times < 50ms for read and write access.
- Mass Data/Load tests on database and API
- Client response times < 1sec for administrators
- Accessibility for end users
- Accessibility for administrators
- 80%+ code coverage in unit tests
- Theming of the UI
- CI/CD Test Automation
- Traceability
- Internationalization
- Security scans
- Client response times < 1sec for end users
- Others? - **Backup and Disaster Recovery**. Implementation of robust backup and disaster recovery plans to protect data and ensure business continuity.

General Questions

1) Assuming a company has up to 50000 users and applications and 5000 roles and up to 200 concurrent user.

What additional mechanisms would you consider in the UI, Service API or database to create a fast customer experience?

- Caching – Implement some kind of caching, especially for data that is frequently accessed but infrequently updated. This can be achieved by leveraging in-memory caching solutions, reducing database load and improving response times.

- Load balancing - to ensure no single server becomes a bottleneck, also improves the system's performance and reliability. By spreading the workload, load balancing can help to achieve faster response times and a more consistent user experience. We can use cloud-based solutions such as AWS Elastic Load Balancer.

- Ensure proper indexes are added to database columns that are frequently used in queries. Indexing these columns speeds up data retrieval and improves query performance.

- UI – The UI needs to be Accesibble and easy to use for all of the users. The most frequently used features in the applications should be easy to access and navigate. The UI must use contrast colors for easy reading. The UI must have support for the most

frequently used languages, users should be able to choose a preferred language and region they like to use in the Web UI.

2) Assuming the database storing the above entities is shared for many companies, which additional information/considerations are needed on the database and the service level to achieve a company/tenant separation.

- Include company/tenant ID in all tables
- Ensure that data is kept separate and access is controlled at the database level.

3) Assuming the names of applications are translatable, how would you approach the internationalization of the application name from a database and API level.

- Store translatable names in a separate table with language codes. This table can be cached in the memory as it will not be changed that often. This will take some of the load of the database.
- Fetch the names based on user's preferred language

4) In addition, users should be able to choose a preferred language and region they like to use in the Web UI.
What information would you suggest storing in the user table? How you find out a good fallback if the users did not specifically choose a preferred language and region?

- **users** table can be updated with the following fields: language, region
- Fallback mechanism – we can use browser geolocation data or default to a common language like English

5) Your Team Architect has not yet decided on the database, middleware (for the service) and UI technology to use for the implementation. Which technologies would you recommend to her/him?

- Database: MySQL and MongoDB if we have statistics for the application.
- Middleware/Backend: Node.js with Express
- UI Technology: React, Angular if we want the application to be secure.

6) For statistical reasons, SAP likes to track the usage (launch) of applications, where and how you would track this information?

- Log each application launch with user ID, timestamp and IP address
- Store logs in a separate analytics database, service or log files