OBWOB Container Diagram

Participant
[Attendee]

Workshop participant

Facilitator
[Attendee]

Workshop facilitator

Coordinator(s)
[User]

Manages organization's info, runs reports

Interacts with →

Organization systems
[Software systems, databases]

Internal systems e.g. user databases, event management

Scans QR code to access

Registers (if required), receives question prompts, submits responses

Selects or creates questions & pushes them to participants' devices in real-time

Visits obwob.com using [HTTPS]

Manages org's info, runs searches and reports using

Mobile App (Progressive Web App) user interface

Technology: React, WebSockets for real-time data synchronization

Responsibilities:

- Allow participants to register (if required)
- Allow facilitators to select/create/push questions
- Present question prompts to participants
- Submit responses

Single-Page Application

Technology: React

Responsibilities:
All functionality for users' interactions with app, including

- Make API requests
- Display dynamic content from the API

Delivers PWA to user's mobile web browser/home screen (offline access)

Makes requests (e.g. fetching questions, submitting responses)

Sends requested data to be dynamically displayed in the user interface

Sends requested data to be dynamically displayed in SPA

Makes requests (e.g. fetching data, submitting forms)

Delivers to the user's web browser

API layer

Technology: Axios with React Query

Responsibilities:

- Process API requests from the user interface
- Interact with backend/database as required based on API requests

API layer

Technology: Django REST framework

Responsibilities:

- Process API requests from the SPA
- Interact with database as required based on API requests

Interacts based on API requests from PWA user interface
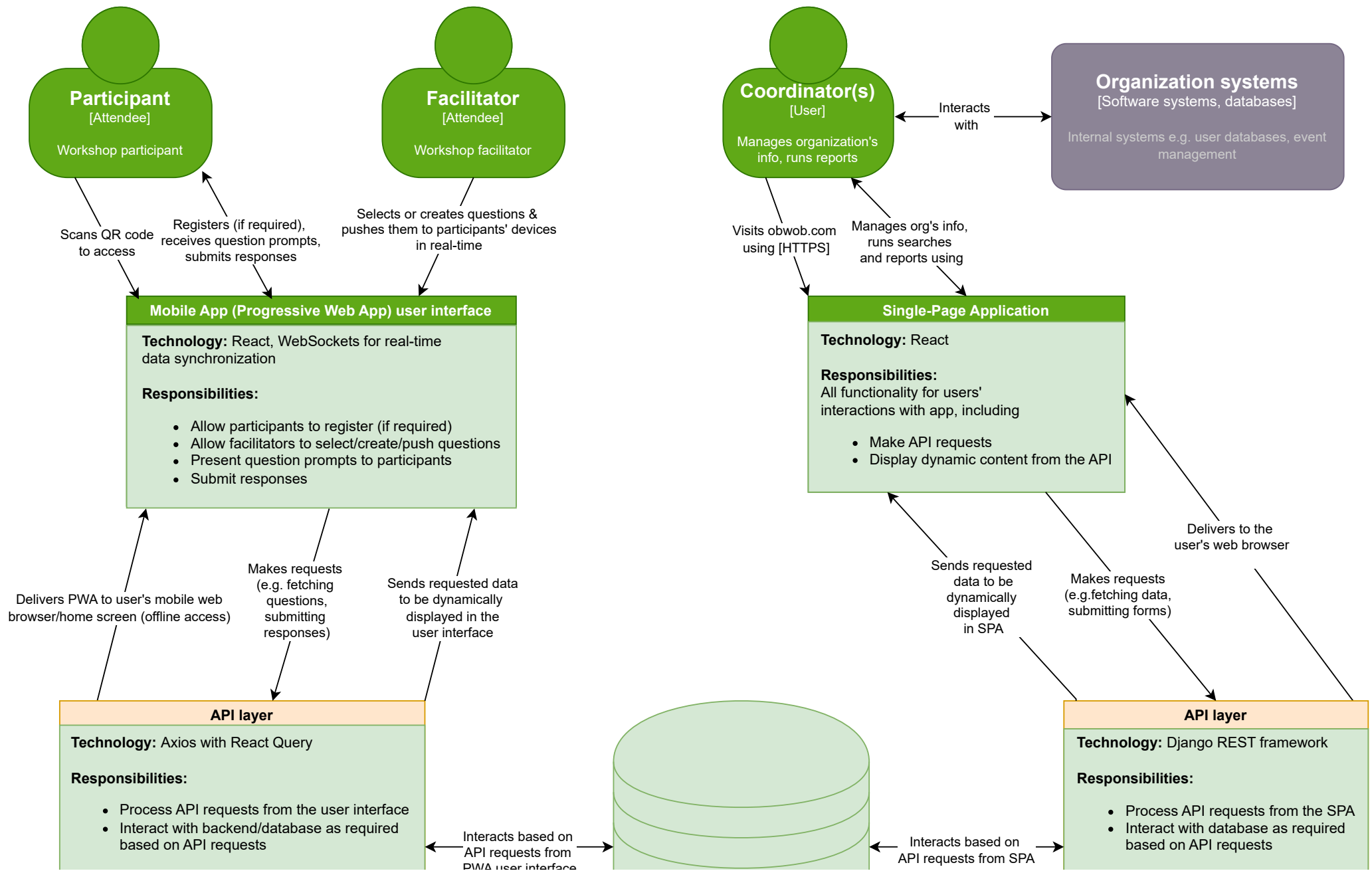
Interacts based on API requests from SPA

PWA user interface

Mobile App (Progressive Web App) backend

Technology: Ruby on Rails

Responsibilities:

- Serve the user interface to participants
- Interact with database vie API layer

Models:

- Attendees (Facilitators, Participants, EventParticipants)
- Responses

Database

Technology: MariaDB

Apps:

- Attendees
- Authentication
- Coordinators
- Demographics
- Events
- Organizations
- Questions
- Reports
- Responses

Web App backend

Technology: Django

Responsibilities:

- Serve the SPA to coordinators
- Interact with database via API layer

Models:

- Authentication
- Coordinators
- Demographics (tied closely to event & reports)
- Organizations, Events, Questions, Reports (for backend management)