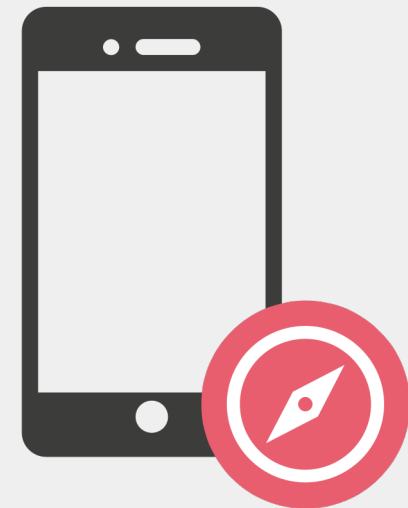


Backend Development

COMPASS NUM-APP

22.04.2021



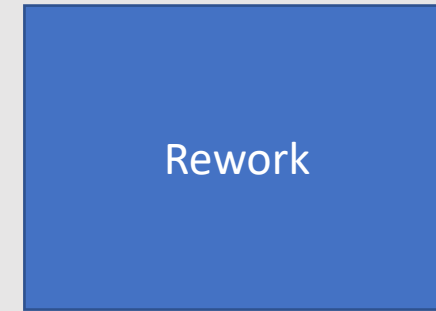
Objectives

After this session, you should be able to:

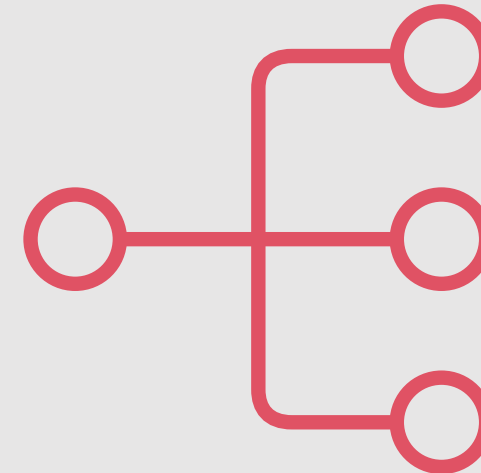
- Understand the technology stack of the NUM mobile backend
- Get a first overview of the functionality of the NUM mobile backend



Agenda

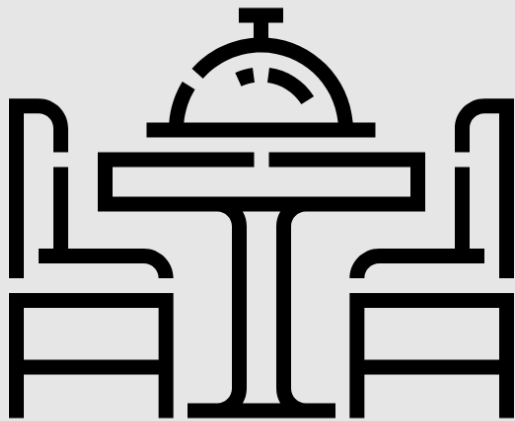


- Introduction Backend Development
- Technology Stack Overview
- Initial Introduction to NUM Mobile Backend
- Outlook on Backend Contributor Track
- Q&A

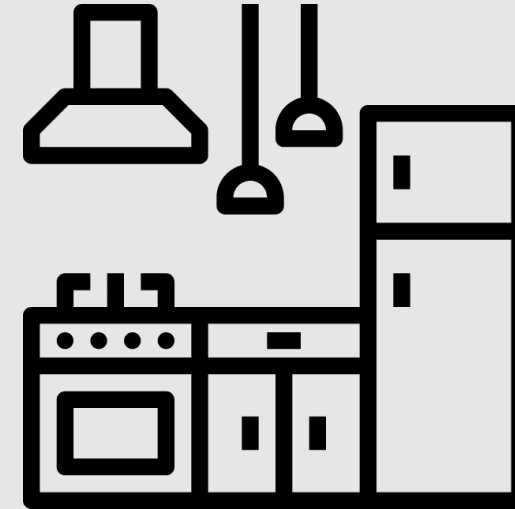


Introduction Backend Development

What is a Backend?

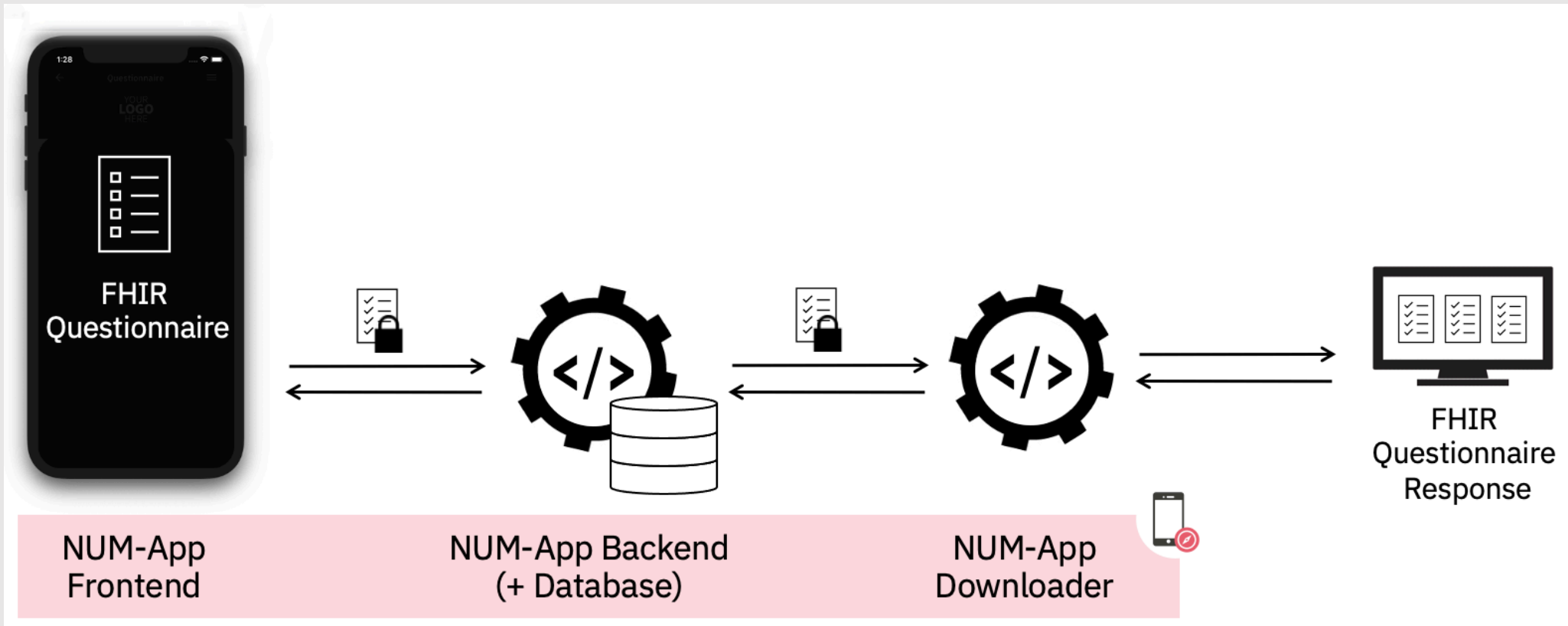


Frontend



Backend

Num Components



Technology Stack Overview

Node.js



JavaScript runtime designed for developing server-side and networking applications

Node.js = Runtime Environment + JavaScript Library

Features:

- Asynchronous and Event Driven
- Very Fast
- Single Threaded but Highly Scalable
- No Buffering
- Open source

Minimum Version: 12

TypeScript



Typed JavaScript

Features:

- Open source
- Code validation
- Type-checking
- Supports JavaScript

```
const user = {  
  firstName: "Angela",  
  lastName: "Davis",  
  role: "Professor"  
}
```

```
console.log(user.name)
```

```
Property 'name' does not exist on type '{ firstName: string;  
lastName: string; role: string; }'.
```

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}
```

```
function updateUser(id: number, update: Partial<User>) {  
  const user = getUser(id)  
  const newUser = {...user, ...update}  
  saveUser(id, newUser)  
}
```

Express



Web framework for Node.js

Features:

- Open source
- Lightweight
- Flexibility
- Enhances Node.js
- Support for JavaScript

```
const express = require('express')
const app = express()
const port = 3000

app.get('/', (req, res) => {
  res.send('Hello World!')
})

app.listen(port, () => {
  console.log(`Example app listening at http://localhost:${port}`)
})
```

Flyway

Database migration tool

Features:

- Track database schema changes
- Prevents manual changes in the database

```
-- Table: apiuser
CREATE TABLE apiuser
(
  api_id character varying(355) NOT NULL,
  api_key character varying(355) NOT NULL,
  api_key_salt character varying(355) NOT NULL,
  CONSTRAINT apiuser_pkey PRIMARY KEY (api_id)
);
```



Introduction to NUM Mobile Backend

Functionality of Mobile Backend



Login



Provisioning of
FHIR questionnaire



Signing of FHIR
questionnaire response



Storage of FHIR
questionnaire response

☰ README.adoc ✎

NUM-App Mobile Back End

[Main Repository](#) | [Back End Documentation](#)

Welcome

This repository provides the source code for the mobile back end of the [Compass NUM-App Project](#). This project provides a set of open source components meant for the digital conduct of questionnaire based studies. The mobile back end itself is a part of [COMPASS](#) (Coordination On Mobile Pandemic Apps best practice and Solution Sharing).

The mobile back end provides study data for the NUM-App in form of [FHIR Questionnaires](#). It also stores the study data that is uploaded from the mobile app. Additionally it makes the collected data accessible for other parties.

Development

Local Setup

- Make sure you have a recent version (LTS recommended) of [Node.js](#) installed and run the following commands to download and prepare this repository:

```
git clone https://github.com/NUMde/compass-numapp-backend.git
npm install
```

- In case you use VSCode as your editor, install the recommended extensions

Run the back end locally

Create .env file

Some configuration values need to be present as environment variables during runtime. The application loads a file with the name `.env` during startup, if it is present.

To get started copy the file `.env.sample` to `.env` and add your values.

Generating RSA key pair

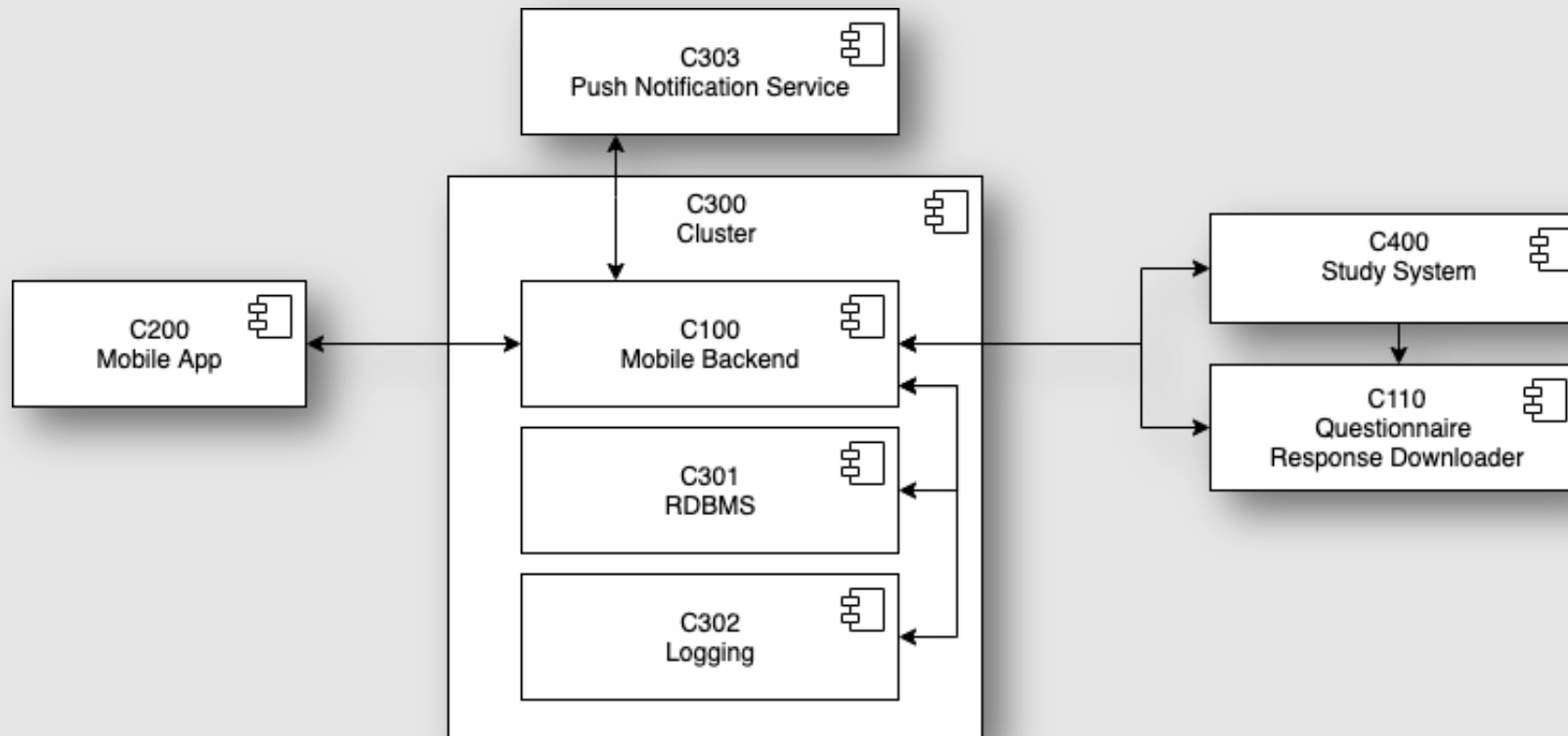
Component Overview

C1xx - Backend Components

C2xx - Frontend Components

C3xx - Technical Components (Middleware)

C4xx - University Hospital Specific Components

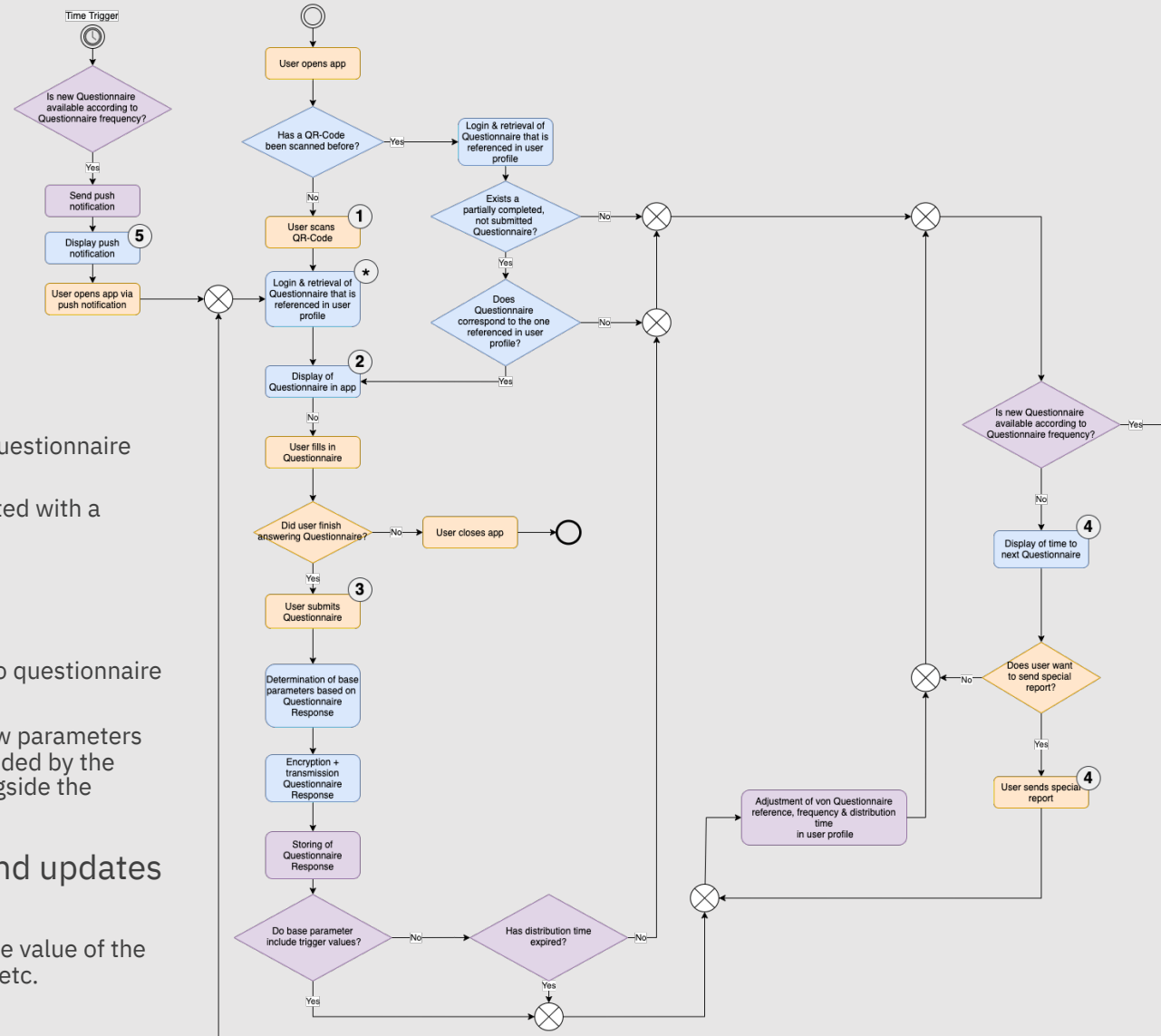


The App Flow

App Flow An Overview

- The user triggers an update, by:
 - Logging in / Opening the app
 - Manual User Refresh
 - Sending out a Questionnaire Response / Sending out a report
- The app loads the questionnaire (if available), either from:
 - The backend (should there be no locally persisted questionnaire whose ID matched the currently active one)
 - The local storage (in case there is one locally persisted with a matching ID)
- The user completes the questionnaire and sends it to backend
 - Alternatively: The user sends out a report (in case no questionnaire is available)
 - If the user sends out a questionnaire response, a few parameters are determined (based on a predefined ruleset provided by the backend or through a local definition) and sent alongside the encrypted questionnaire response.
- The backend persists the received data and updates the user.
 - The user updated is based on various factors, like the value of the additional parameters, the currently active interval, etc.

Flow Chart

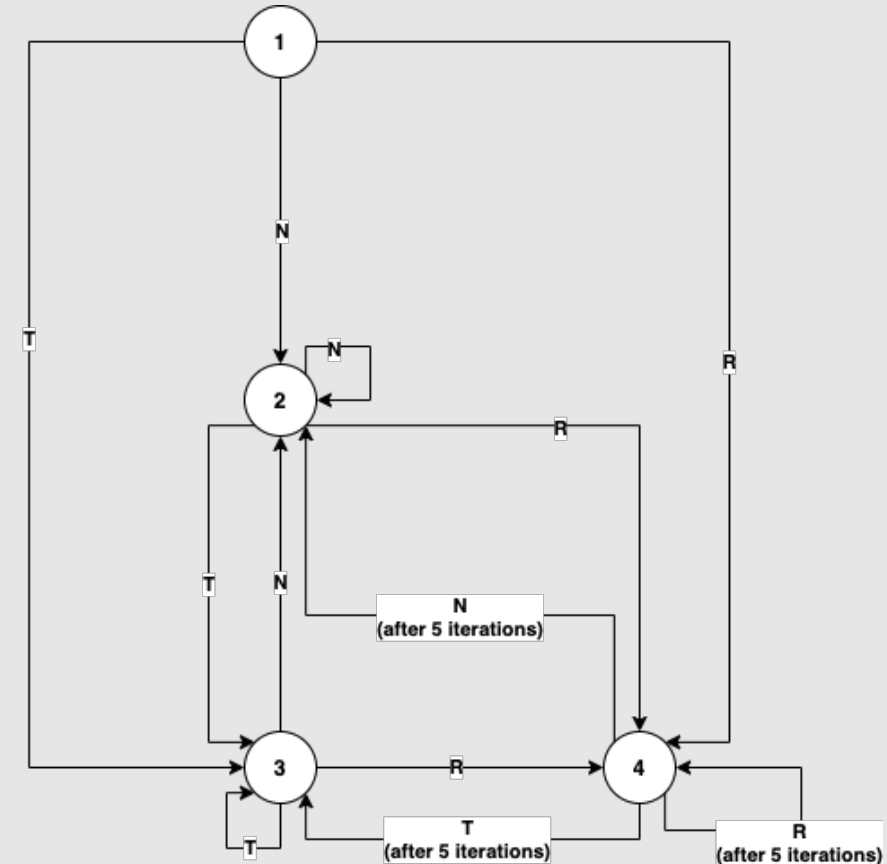


App Flow Updating the User (server side)

This graph displays how the backend determines **what questionnaire the user will receive next**.

- 1 Questionnaire 1, Starting Point, will be sent out once per user
- 2 Questionnaire 2, default, 7 days interval
- 3 Questionnaire 3, after hitting a Trigger, 2 days interval
- 4 Questionnaire 4, after sending out a report, 2 days interval at 5 iterations

| | |
|---|---|
| N | Neutral (no Trigger activated, next questionnaire: 2) |
| T | Trigger (trigger was activated, next questionnaire: 3) |
| R | Report (report was sent out, next questionnaire: 4; for 5 iterations, after that: reassessment) |



Back End Configuration files & options

`.env`

(defines config for local development)

`COMPASSConfig.ts`

(defines all logic related parameters)

```

.env.sample
  You, a month ago | 1 author (You)
  1 # database connection values
  2 DB_HOST='localhost'
  3 DB_PORT='5432'
  4 database-name='compass'
  5 database-password='123456'
  6 database-user='compass_user'
  7
  8 # a secret used for JWT generation
  9 JWT_SECRET='mein_schatz'
 10
 11 # optional Push Service configuration
 12 PUSH_CLIENT_SECRET='xyz'
 13 PUSH_API_KEY='xyz'
 14 PUSH_APP_ID='xyz'
 15
 16 # have node use a common time zone
 17 TZ = 'Europe/Amsterdam'
 18
 19 # encryption keys
 20 COMPASS_PUBLIC_KEY='-----BEGIN PUBLIC KEY
 21 COMPASS_PRIVATE_KEY='-----BEGIN PRIVATE K
 22

src > config > TS COMPASSConfig.ts > COMPASSConfig
  You, a month ago | 1 author (You)
  1 /*
  2  * Copyright (c) 2021, IBM Deutschland GmbH
  3  */
  4 import * as env from 'env-var';
  5
  6 export class COMPASSConfig {
  7     /**
  8     * Gets the value the type-attribute of a request must hold so that the backend knows that
  9     * its body holds an encrypted questionnaire response. This is necessary because of said encryption,
 10     * as there is no other way to determine what the body holds without decrypting it.
 11     *
 12     * @static
 13     * @return {*} {string}
 14     * @memberof COMPASSConfig
 15     */
 16     public static getQuestionnaireResponseType(): string {
 17         return env.get('COMPASS_QR_REQUEST_TYPE').default('questionnaire_response').asString();
 18     }
 19
 20     /**
 21     * Gets the default interval for a regular questionnaire. This determines after what amount of days a new iteration starts.
 22     *
 23     * @static
 24     * @return {*} {number}
 25     * @memberof COMPASSConfig
  
```

Database

ER Model

| studyuser | |
|--------------------------------|----------|
| ABC study_id | NOT NULL |
| 🕒 last_action | |
| ABC current_questionnaire_id | |
| 🕒 start_date | |
| 🕒 due_date | |
| ABC current_instance_id | |
| 123 current_interval | |
| 123 additional_iterations_left | |

| questionnaires | |
|----------------|----------|
| ABC id | NOT NULL |
| JSON body | NOT NULL |

| apiuser | |
|------------------|----------|
| ABC api_id | NOT NULL |
| ABC api_key | NOT NULL |
| ABC api_key_salt | NOT NULL |

| questionnairehistory | |
|----------------------|----------|
| ABC id | NOT NULL |
| ABC study_id | NOT NULL |
| ABC questionnaire_id | NOT NULL |
| 🕒 date_received | NOT NULL |
| 🕒 date_sent | |
| ABC instance_id | |

| queue | |
|--------------------|----------|
| ABC id | NOT NULL |
| ABC study_id | NOT NULL |
| ABC encrypted_resp | NOT NULL |
| 🕒 date_sent | NOT NULL |
| 🕒 date_received | NOT NULL |

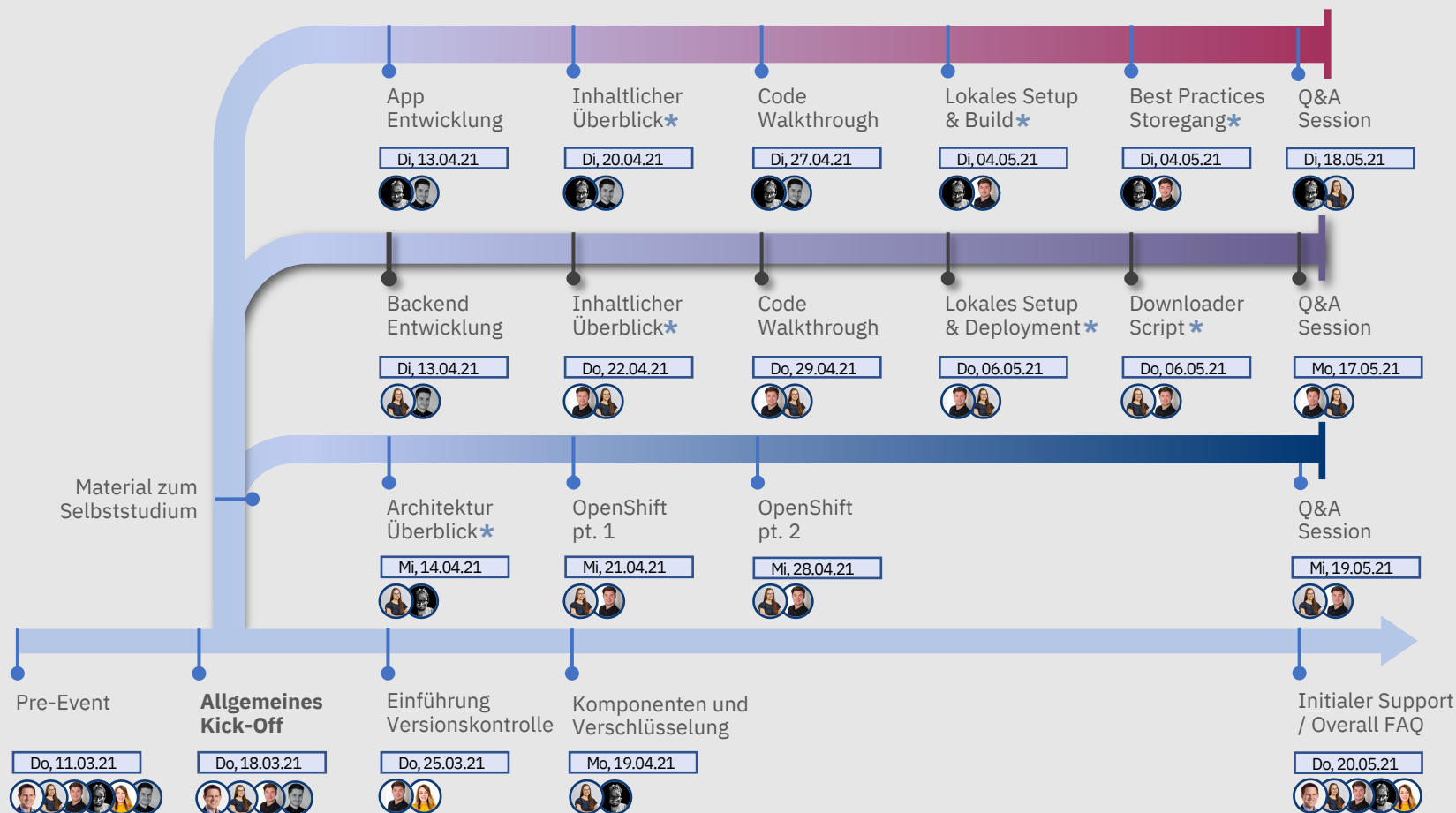
API Documentation



Demo

Outlook on Backend Contributor Track

Backend Contributor Track





Links

Links

Node.js

- <https://nodejs.org/en/about/>

Express

- <https://expressjs.com/de/>

JavaScript / TypeScript

- <https://www.typescriptlang.org/>
- https://www.tutorialspoint.com/typescript/typescript_quick_guide.htm
- <https://blog.logrocket.com/typescript-with-node-js-and-express/>
- <https://www.w3schools.com/js/DEFAULT.asp>



Q&A

Q&A

*What questions
do you have?*

