**Cordova-project 2APPAI – 2WT - 3IoT**

**Brief description of the app**

**A picture containing person, outdoor

Description automatically generated**This application is an agenda/deadline planner.

When the user opens this application, he is prompted to login with a username (if he wasn't logged in yet).

Then the user arrives on his main page where he sees all his appointments or deadlines ordered by date and by group. The day before each deadline, the user gets a speech notification that says the name of the deadline and the description. The hour the notification will trigger can be set when you make a new deadline.

It is possible to make groups in the groups tab. A group is a combination of a colour and name of the group. The deadlines on the home page can be filtered, if a user deselects a group, this group won't be shown on the home page. This filter is stored so if the user logs in next time his deadlines will be filtered in the same way.

When the user creates a new deadline, he has the possibility to link a location with it. This is done by pinning a point on an easy-to-use map. If the user wants to see the location of a deadline on the home page, he can simply tap on this deadline and the map will slide open. This map is made possible by the leaflet.js API.

All this data is collected on a rest API written in Spring Boot and it runs on my server at home.

Users can create multiple agenda on my application. For example, the user can use one agenda as a real agenda to write down some deadlines, and the other to create a planning for the upcoming exams.

The main goal for my application is to be useful in the real world. Last year, I was struggling to find an application to localize al my deadlines for school and spare time but concluded there was no application that fitted my needs. Thats why I thought an application as this would be very useful for myself, and maybe for others too.

**Plugins**

* Cordova-plugin-whitelist:

Used for implementing a whitelist policy for navigation the application webview on Cordova 4.0 and above. It Provides security and configurability.

* Cordova-plugin-mapbox:

Used together with leaflet js. The mapbox Cordova plugin provides OpenGL powered vector-based native mapping. These maps are used to specify the location of the appointment with a pinpoint.

* Cordova-plugin-fullscreen:

Used to open my application on fullscreen on my Android phone. This makes the app more clear since there are 2 navbars on the screen (one on the bottom and one on the top). Especially the bottom navigation is packed full of information if the native Android navbar is displayed together with the navbar of my application. I've also looked up example apps that have a bottom navbar like Instagram and Twitter, and here I don't like the fact that they still show the Android navigation. That's why I've chosen to hide the native Android bottom navbar.

* Cordova-plugin-dialogs: Used to notify the user when he/she is doing something in the app with irreversible concequences like removing a group or deadline. The user gets an alert when he tries to delete a group in which there are still some deadlines/appointments.
* Cordova-plugin-background-mode:

This plugin makes the app run in the background when focus is lost. All code such as timeouts keep running now. I use this to make a notification for the user. There is a timeout set and while the app is running in the background, the timeout is counted down. When it reaches 0, the notification is sent. This even works when the phone is in sleep mode.

* Cordova-plugin-compat:

To make the background mode plugin backwards compatible with previous version of cordova.

* Cordova-plugin-device:

Defines a global device object, which describes the device's hardware and software.

* Cordova-plugin-speechrecognition:

A plugin that uses your microphone and Google Speech to convert microphone input to an array of text strings. This is used to navigate the user through the app and to input a title and description for a new deadline.

* Cordova-plugin-tts:

This plugin converts a string to a spoken message. This is the notification the user gets when the deadline is near. It converts the title and description of a deadline to a text message.

**APIs**

* Fullrest API Spring Boot:

A rest api that manages the CRUD of deadlines, groups and users. It runs on my server at home and is accessible that the url: <http://lennertsoffers.com/plennert> .

* Leaflet.js:

An open source javascript library for mobile-firendly interactive apps. I use leaflet to create an interactive map to show the location of deadlines. A marker is added on the location of the clicked coordinates.

**Local storage**

This is where I store the user that is logged in. In this way, the app is automatically started with the correct user in. The group filter is stored here as well, it is stored across sessions so that the user doesn't need to add the filter every time he starts up the app.

**Screenshots**

|  |  |  |
| --- | --- | --- |
| Login page: | Home page: | Groups page: |
| About page: |
| New deadline page: | New group page: | Speech recognition: |