Deployment plan

Version	Date	Author(s)	Changes
0.1	22/04/21	David La Gordt Dillie, Florin Deleanu	First draft
0.2	10/05/21	David La Gordt Dillie, Florin Deleanu	Added steps to run local instance

Contents

1. Overview

1.1 Purpose

The purpose of the deployment plan are as follows:
- Lay out the steps for running the application.
- Cover the capabilities of the application.
- Keep a record of the dependencies used.
- Describe the deployment steps

1.2 Assumptions

The website should have a place for users to:

- Login with their outlook account
- Provide a list of upcoming appointments
- Provide a search feature to find a specific appointment
- Have sorting capabilities to order appointments by date or alphabetically
- Provide an indication if the appointment has started and whether it started on-time

1.3 Dependencies

The back end of the application which takes images and transforms them into usable string variables relies on a license plate recognition system as an external dependency which is added as a module into our project.
We are also using a JavaCV dependency which allows us to input a video and split it into images to be further used by the license plate recognition module.

1.4 Constraints

These factors can limit the ability to deploy the application:

- Change of client's needs for the project
- License plate recognition or API failure

2. Deployment

2.1 Schedule

Once all tests pass and the needs of the client have been met, the application will be ready for deployment. The deployment process will happen every 3 weeks/ every iteration.

3. Local

3.1 Front-end

Required:
NodeJS (https://nodejs.org/en/)
Node packages (run "npm i" inside of front-end folder)

Steps:
1. Open a command prompt inside the seanfrontend folder
2. Run the "npm start" command and wait until the development server is up and running

3.2 Back-end

Required:
IntellijIDEA
MySQL instance (credentials in application.properties file)
Maven

Steps:
1. Open "license plate recognition" folder in Intellij
2. Click File->New->Module from existing sources
3. Select Presentables/CertioremSean/SeansInterface and continue
4. Select Import module from external model-> Maven
5. Finish

After the steps are completed the relative file paths will work correctly because they begin in the "license plate recognition" folder.

In order to input a video to the system a mp4 file named anprVideo.mp4 should be placed in the "vids". When running the server normally from the "main" method in "SeansInterfaceApplication" class, the spring boot app as well as the automatic recognition will start and send messages to the users based on when they arrive in the video.