

# Fleet management system

---

Date: 29.06.2021

Course: Internet Applications Programming

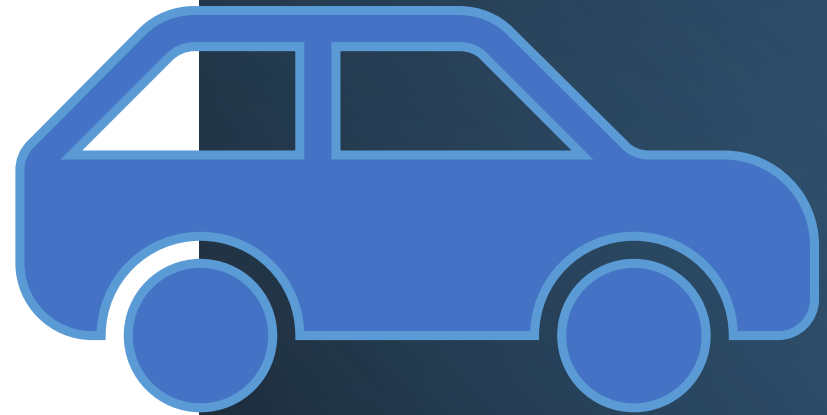
Group #8

# Team members

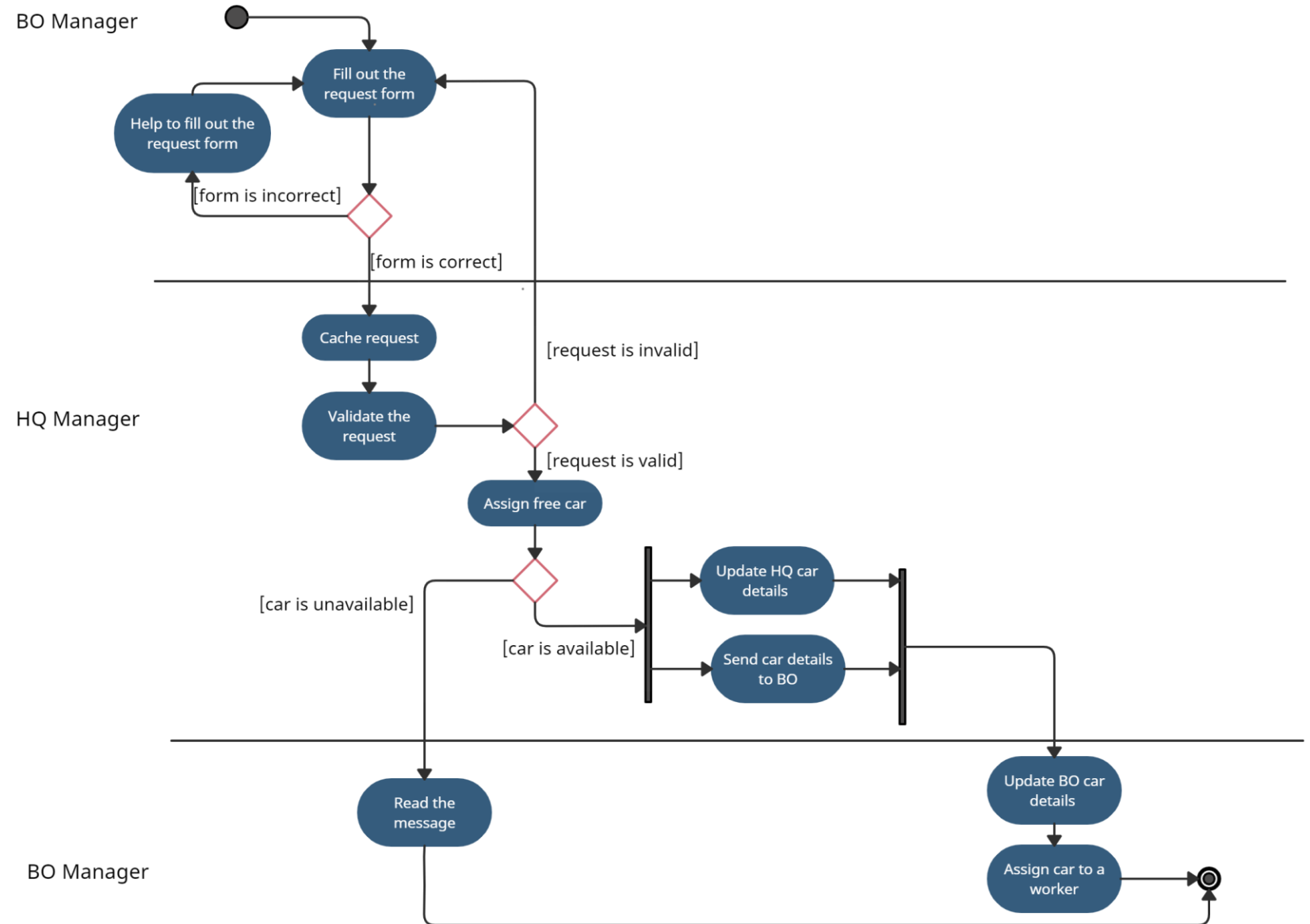
- Monika Rosa 239113
- Godfrey Mghase 239195
- Stanisław Puławski 239111
- Wiktor Muraszko 239109

# Project Objective(s)

- Create an IT System for managing car fleet for the company, which consists of a headquarter and one branch office.
- Implement a business process to enable the branch manager to request a car from the headquarter office.

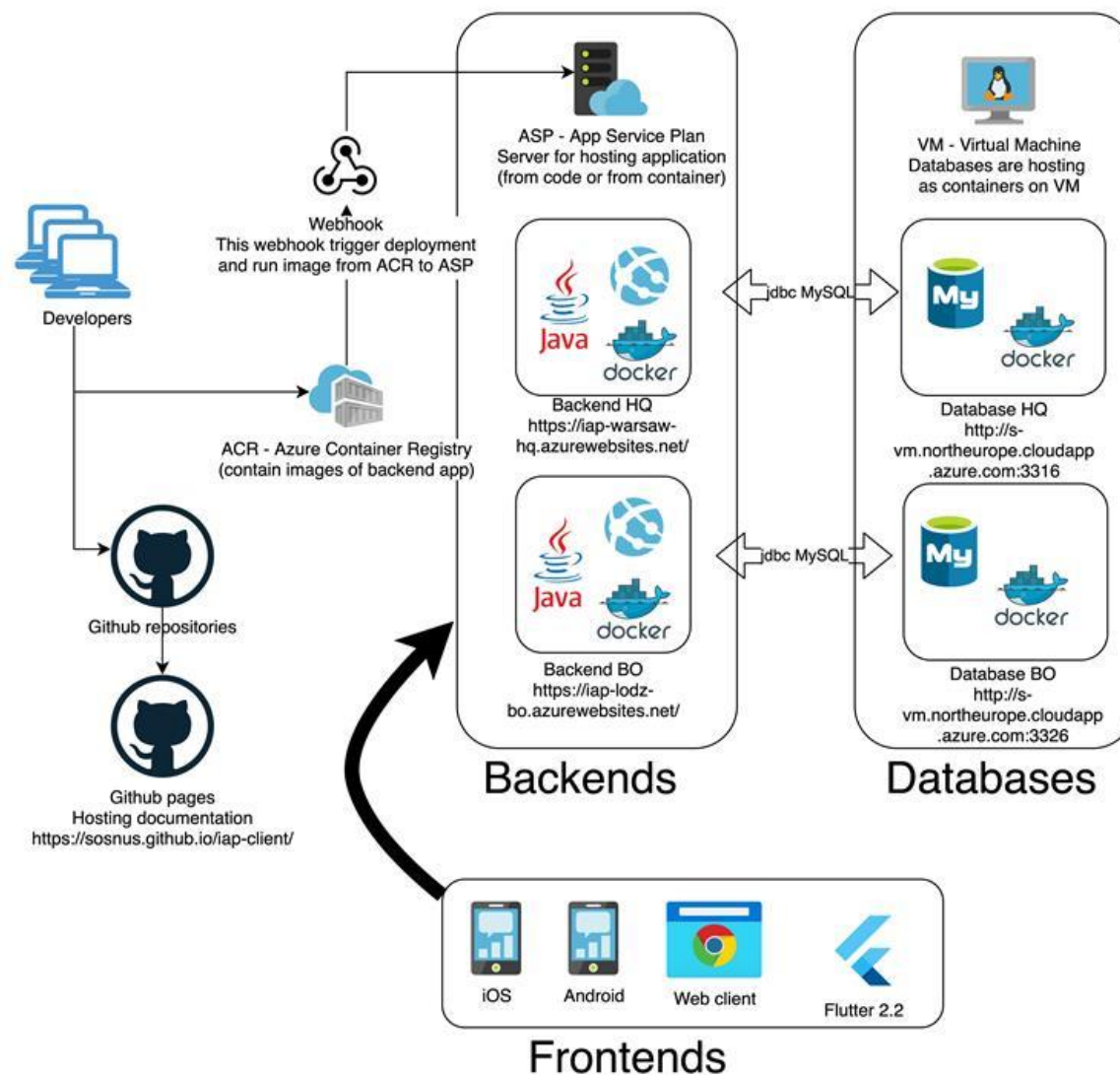


# Car request process



# Solution Architecture

## Fleet Management System Internet Application Programming



# Technology Stack

## Database

- MySQL MySQL Server 8.0.23-1debian10

## Backend

- Java Spring Spring Boot (v2.4.3)

## Frontend

- Flutter Flutter 2.2.2
  - Android
  - iOS
  - Web

# Java Spring Boot

AutoConfiguration

Starter POMs

Spring Boot CLI

Actuator

Spring Boot Initializer

# Data Exchange

Prepare an object and invoke the post method for transfer

```
// Prepare the data to be sent to HQ for this request

requestData.setRequestorId(r.getRequestorId().getId());
requestData.setBranchId(r.getBranchId()); Kliknij, aby dodać tekst
requestData.setCarModel(r.getCarModel());
requestData.setVehiclePreffered(r.getVehiclePreffered());
requestData.setRequestDate(r.getRequestDate());
requestData.setRequestId(r.getRequestId());

// Now send it
//String uri = "http://localhost:8080/request";
RestTemplate restTemplate = new RestTemplate();
try {
    RequestData result = restTemplate.postForObject(SynchronizationConfiguration.uriToHq,requestData, RequestData.class);
} catch (Exception e) {
```



# Data Sync

- Enable Scheduling in spring boot main application using @annotation.
- Call @Scheduled to any method.
- Enable Swagger by using @EnableSwagger2 annotation at the main application file in spring boot.
- SYNC is scheduled after every 10 seconds.

```
package com.IAP.car_exchange;
```

```
@RestController
```

```
@SpringBootApplication
```

```
@EnableScheduling
```

```
@EnableSwagger2
```

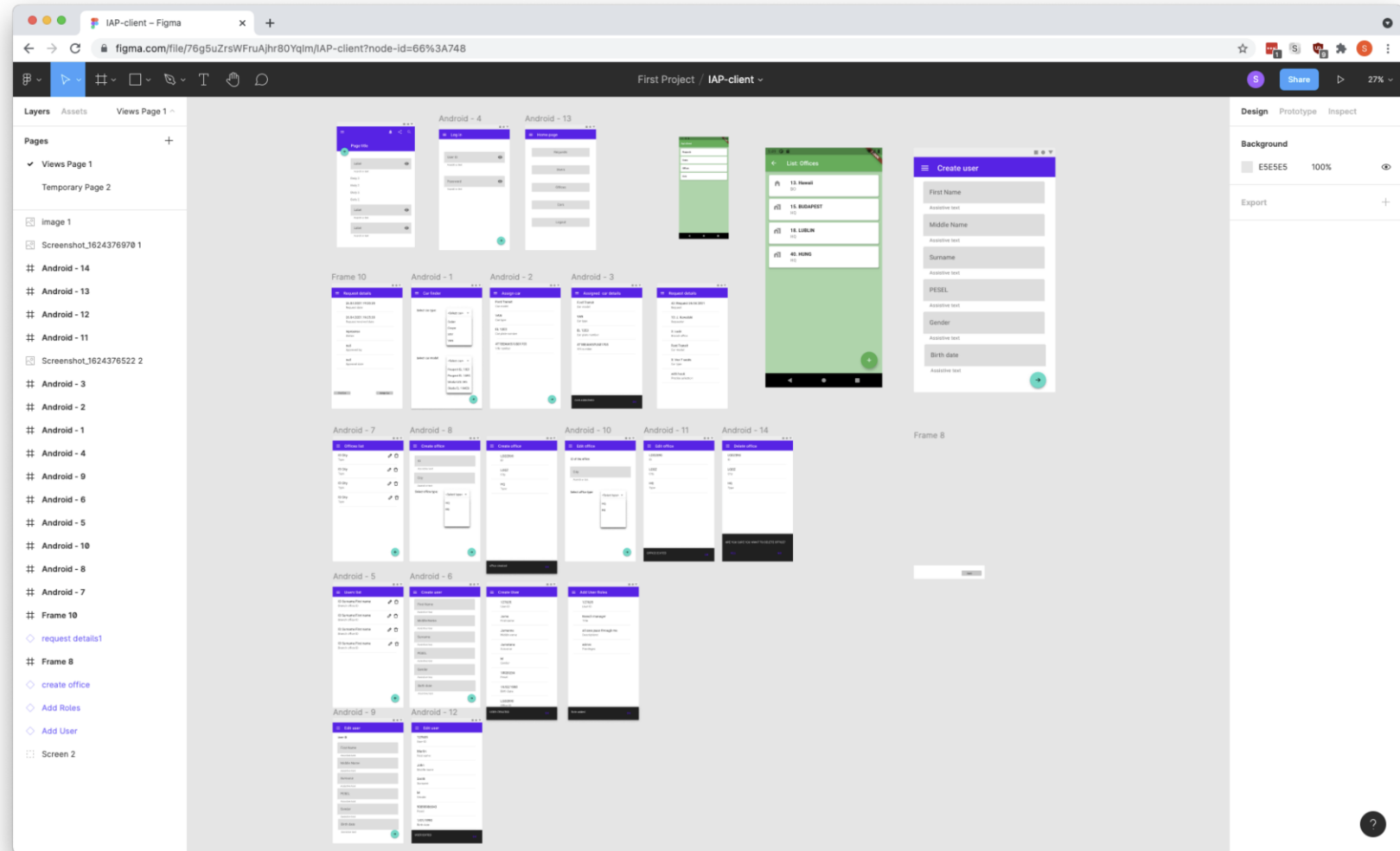
```
public class CarExchangeApplication {
```

```
    public static void main(String[] args) {  
        SpringApplication.run(CarExchangeApplication.class, args);  
    }
```

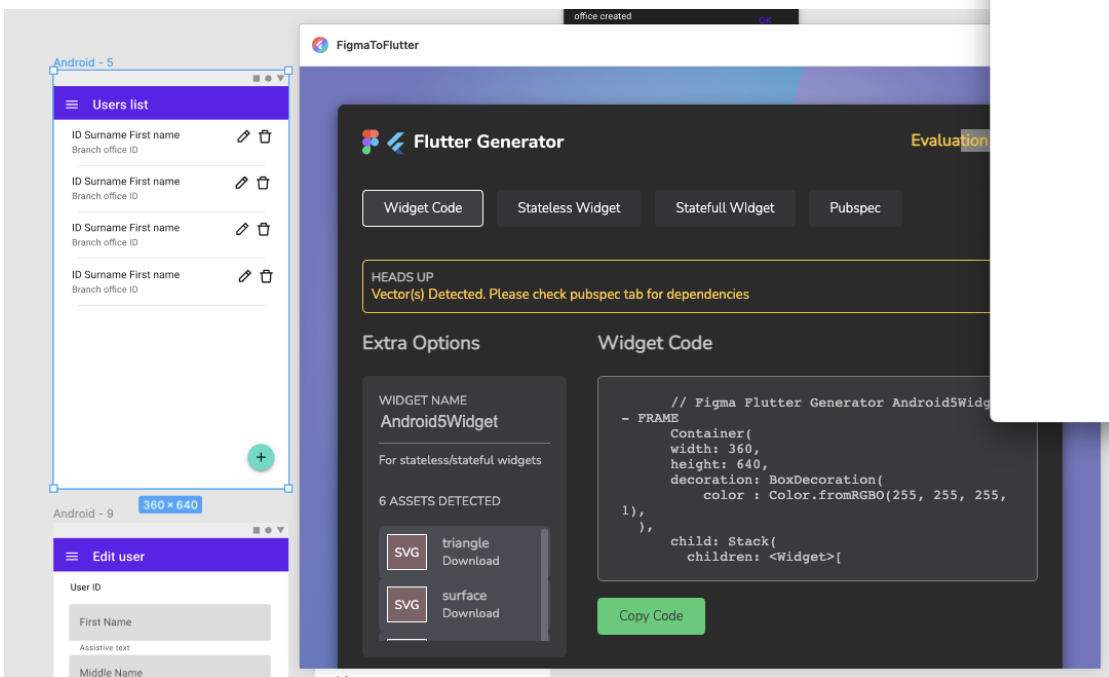
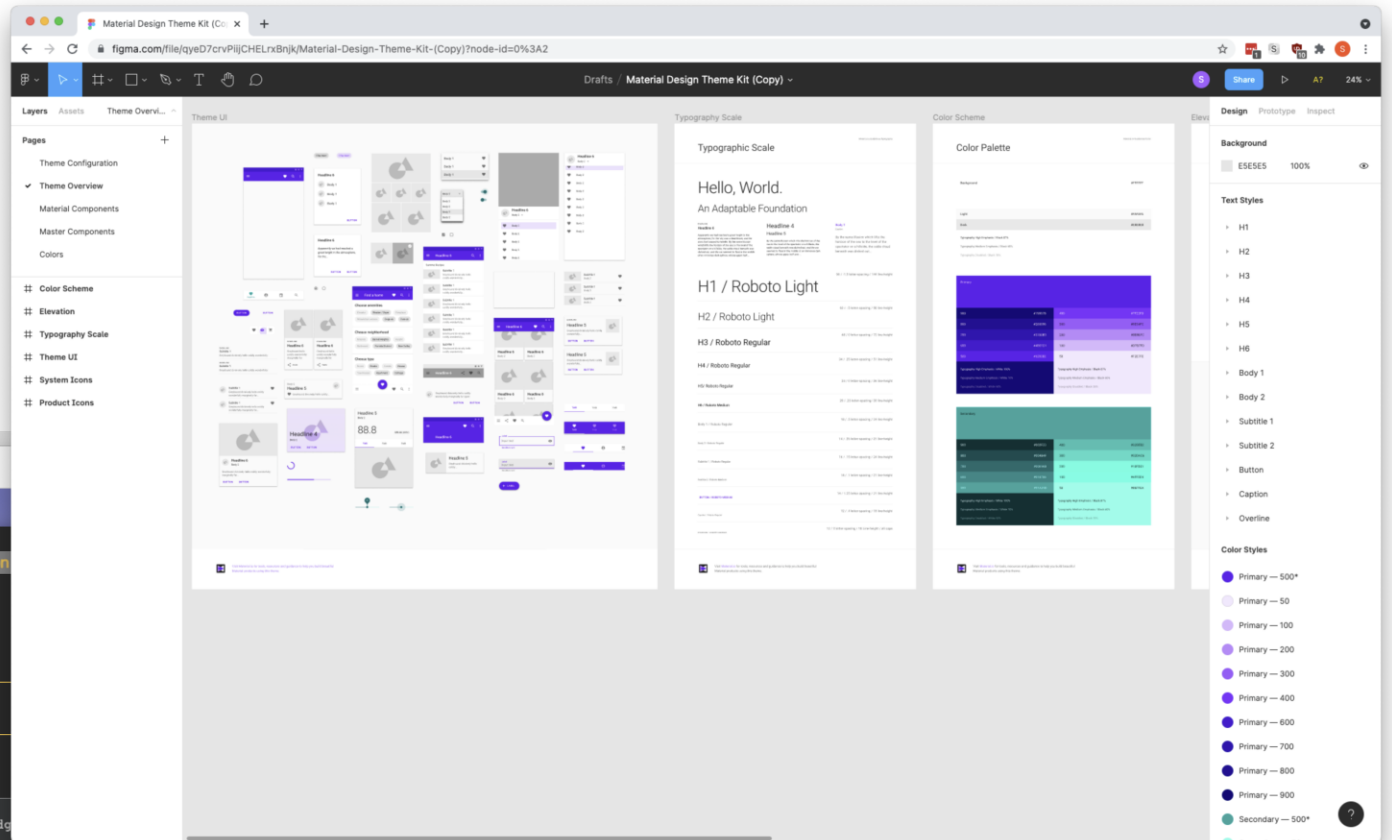
```
    public Docket productApi() {  
        return new Docket(DocumentationType.SWAGGER_2)  
            .apis(RequestHandlerSelectors.basePackage(this.getClass().getPackage()))  
            .build();  
    }
```

```
    @GetMapping("/hello")  
    public String hello(@RequestParam(value = "name", required = true) String name) {  
        return String.format("Hello %s!", name);  
    }
```

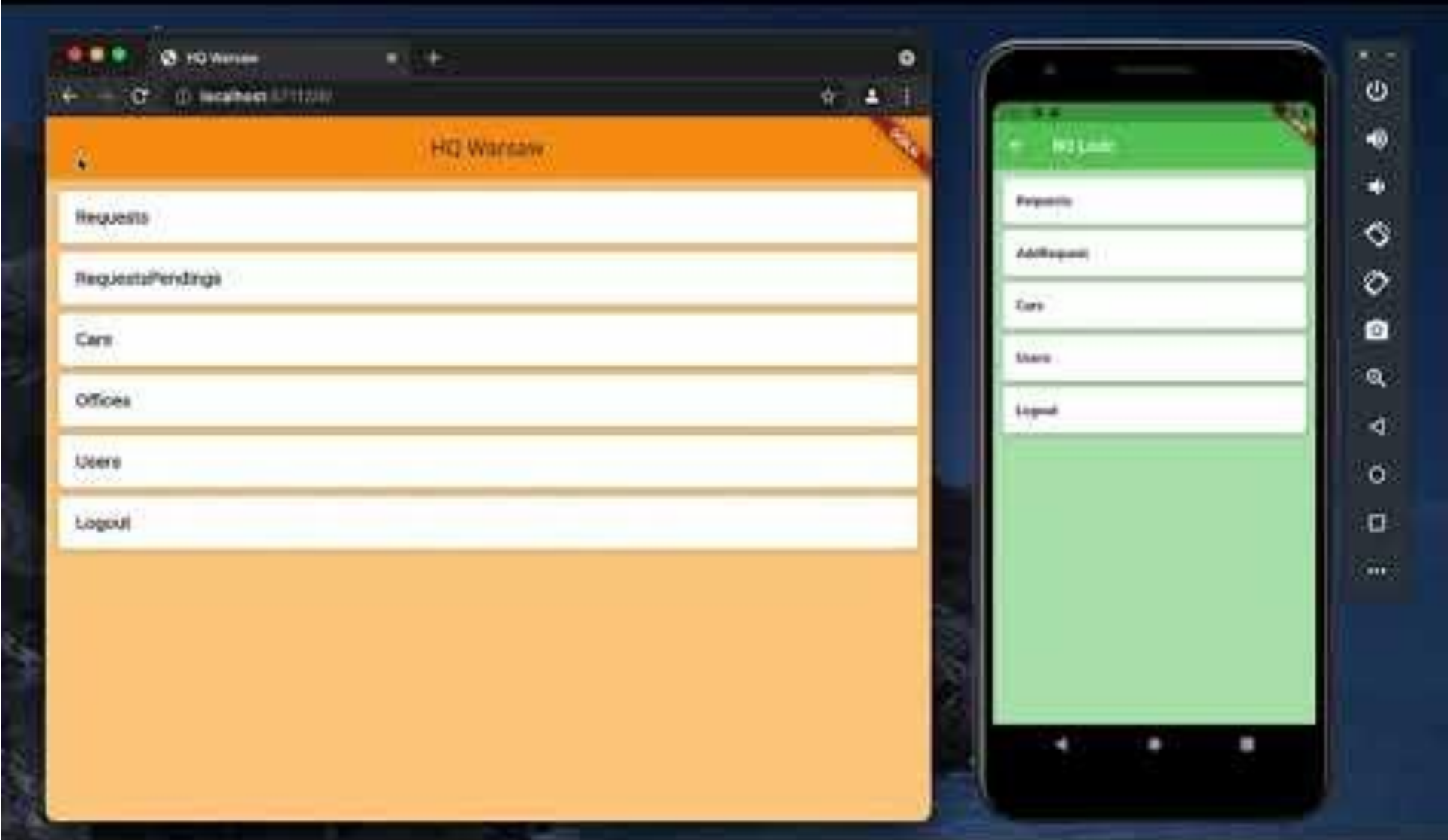
# UI design Figma



# Themes, plugins, kits



# Client in action



# Car Fleet Manager App (frontend)

Features depends on type of office (HQ/BO)

UI adapts to many resolution and screen proportions

UI based on Material Design

Multiplatform (Android, iOS, Web)



# Flutter



Made by [Google](#)

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for [mobile](#), [web](#), [desktop](#), and [embedded](#) devices from a single codebase.

[Get started](#)

# Flutter

Free

Open Source

Dart framework

Null-safety

Hot reload

Multiplatform

- Mobile (Android, iOS)
- Web (Chrome engine)
- Desktop (Windows, Linux, MacOS, FuchsiaOS)

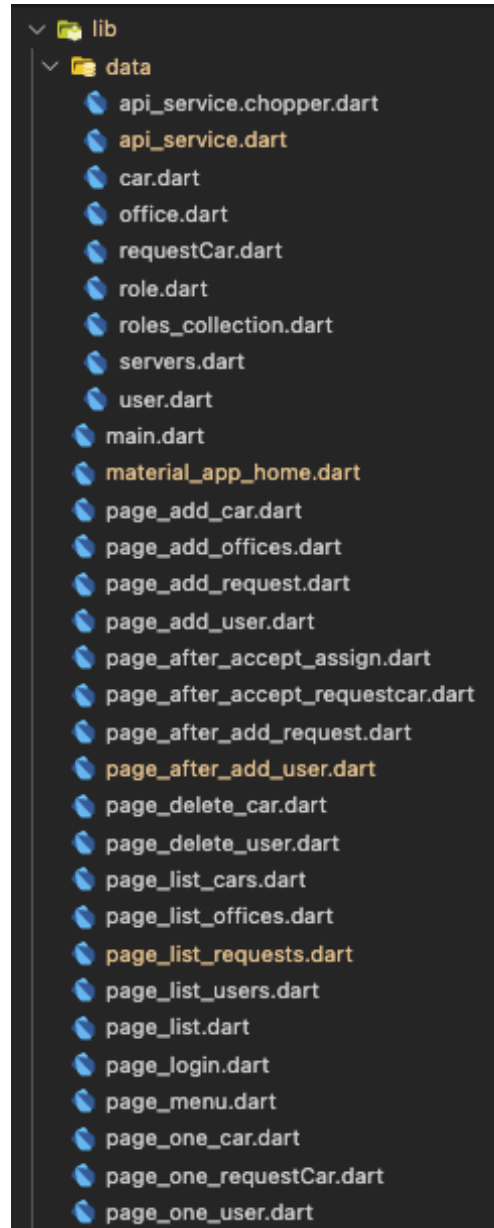
# Flutter vs. React Native: In a nutshell

	Flutter	React Native
What is it?	A portable UI toolkit for building natively-compiled apps across mobile, web, and desktop* from a single codebase	A framework for building native applications using React
Official release	December 2018, Google I/O	March 2015, F8 Conference
Created by	Google	Facebook
Free and open source	Yes	Yes
Programming language	Dart	JavaScript
Popularity	120,000 Stars on Github (May 2021)	95,300 stars on Github (May 2021)
Hot Reload	Yes	Yes
Native performance	Great	Great



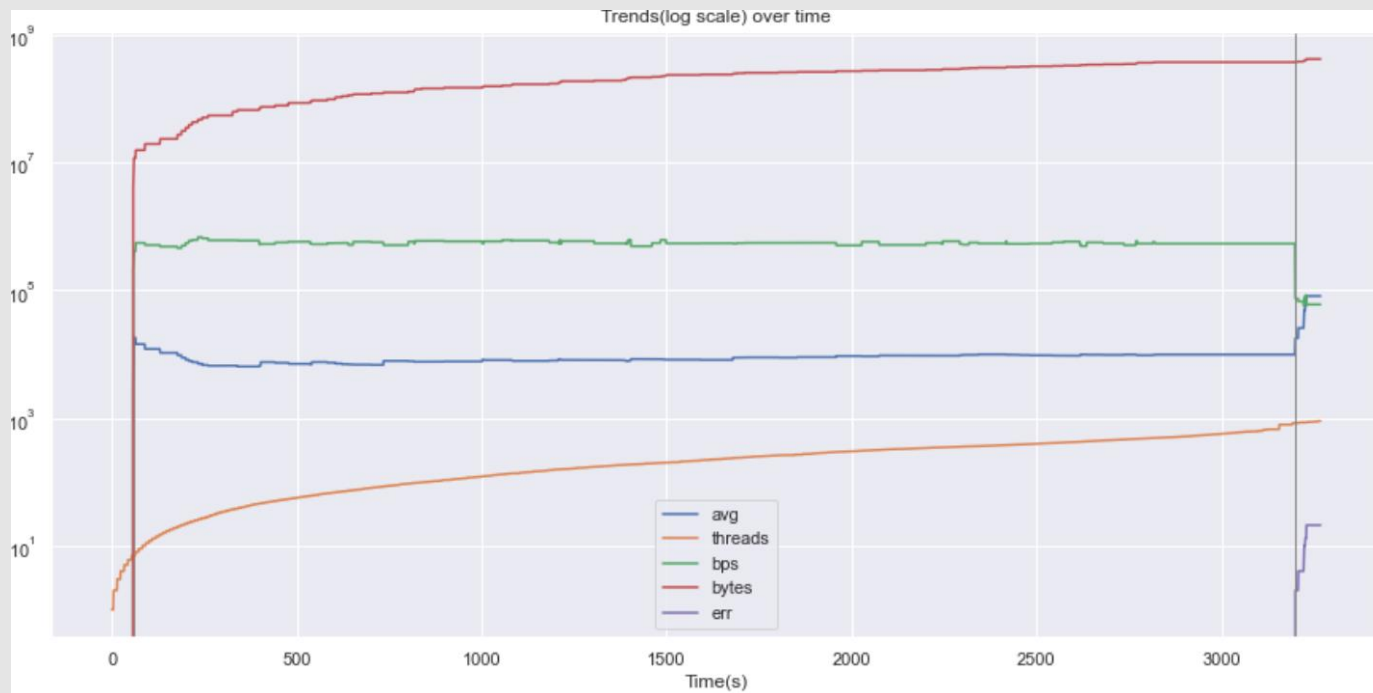
# Important frontend libraries

- Provider – state management
- Chopper – API consumption
- Build runner – to generate Chopper file
- Material Design – UI, controls



```
dependencies:  
  flutter:  
    sdk: flutter  
  flutter_datetime_picker: ^1.5.1  
  intl: ^0.17.0  
  chopper: ^4.0.1  
  provider: ^5.0.0  
  
  cupertino_icons: ^1.0.2  
  restart_app: ^1.0.3  
  
dev_dependencies:  
  flutter_test:  
    sdk: flutter  
  chopper_generator: ^4.0.1  
  build_runner:
```

# Performance Analysis



- Testing Tool: SoapUI 5.6.0
- Testing Computer: 8GHz RAM and Core i5-4258U CPU @2.4GHz 2.4GHz
- Testing Strategy: Thread
- Virtual users/Threads: 836
- Data volume sent: 360 GB
- Speed: 72 mbps
- Average response time: 9.6s

# Deployment

```
docker login sosnuscontainers.azurecr.io
```

```
# login: *****
```

```
# password: *****
```

```
docker build -t sosnuscontainers.azurecr.io/iap-warsaw-hq .
```

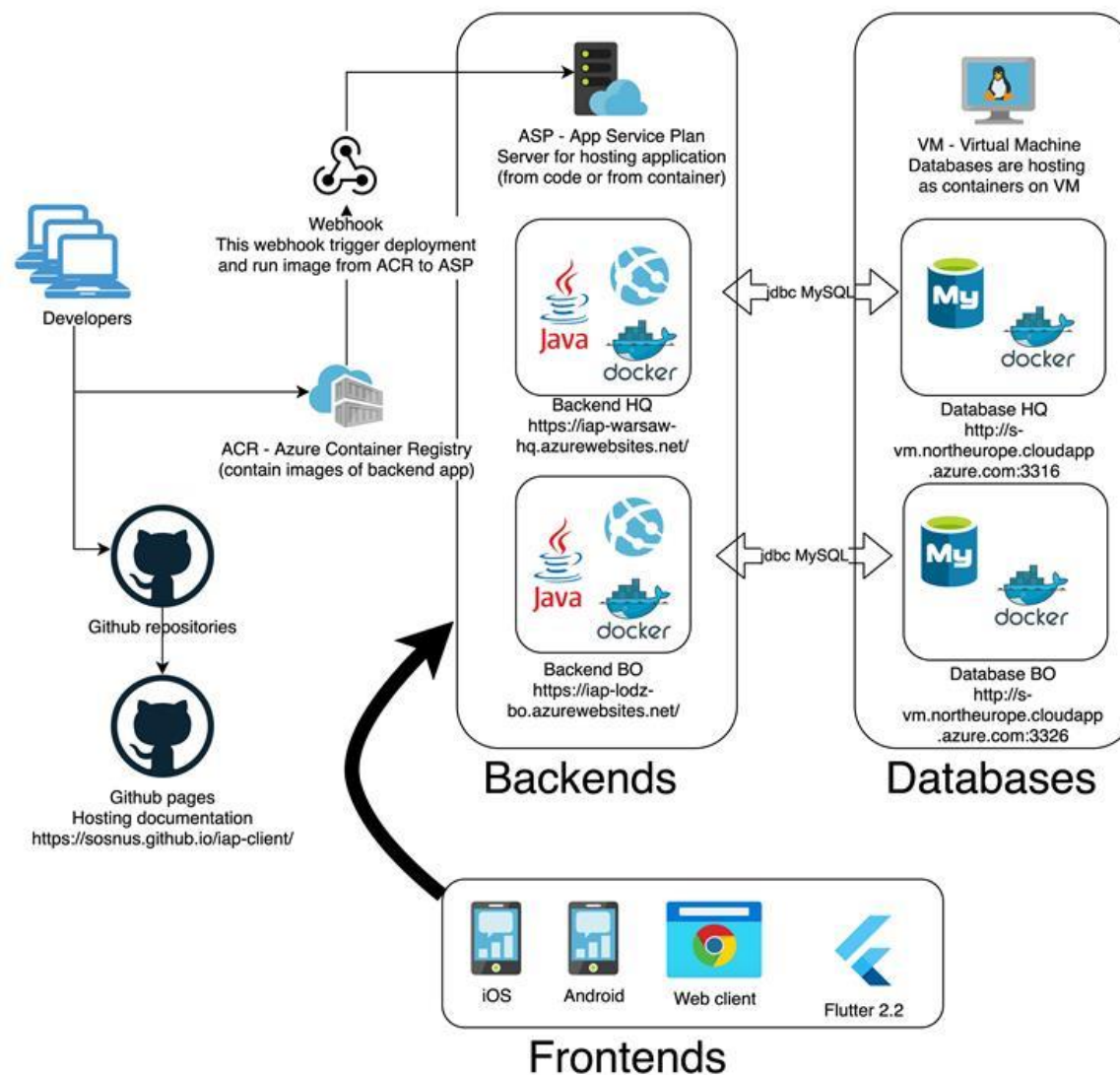
```
docker push sosnuscontainers.azurecr.io/iap-warsaw-hq
```

Steps:

- TODO: Docker push on every commit on Github (using Github Actions)
- Docker push (to ACR)
- Webhook run deployment process (send new image to WebApplication)
- Done!

# Solution Architecture

## Fleet Management System Internet Application Programming



# ACR – Azure Container Registry

The screenshot displays the Microsoft Azure portal interface for the 'sosnuscontainers' Container Registry. The left sidebar contains navigation links for Home, Container registries, sosnuscontainers, Repositories, Access control (IAM), Tags, Quick start, Events, Settings (Access keys, Encryption, Identity, Networking, Security, Locks), and Services (Repositories, Webhooks, Replications, Tasks). The main content area shows the 'iap-warsaw-hq' repository with options to Refresh and Delete repository. Below this, the 'Essentials' section displays repository details: iap-warsaw-hq, Tag count 1, Last updated date 6/23/2021, 10:35 PM GMT+2, and Manifest count 2. A search bar for tags is present. The 'Tags' section shows a list of tags, with 'latest' selected. A context menu is open over the 'latest' tag, showing options: Create webhook, Delete, Untag, Run instance, and Deploy to web app.

Microsoft Azure

Search resources, services, and docs (G+/I)

Home > Container registries > sosnuscontainers >

**sosnuscontainers | Repositories**  
Container registry

Search (Cmd+/) Refresh

Access control (IAM)

Tags

Quick start

Events

Settings

Access keys

Encryption

Identity

Networking

Security

Locks

Services

Repositories

Webhooks

Replications

Tasks

Search iap

Refresh Delete repository

**iap-warsaw-hq**  
Repository

Refresh Delete repository

JSON View

Repository iap-warsaw-hq Tag count 1

Last updated date 6/23/2021, 10:35 PM GMT+2 Manifest count 2

Search to filter tags ...

Tags ↑↓

latest

Create webhook

Delete

Untag

Run instance

Deploy to web app

ASP— App Service Plan

Name	Type	Resource Group	Status
asp1	app,linux,container	rg-s	Running
asp2	app,linux,container	rg-s	Running
asp3	app,linux,container	rg-s	Running
asp4	app,linux,container	rg-s	Running
asp5	app,linux,container	rg-s	Running
asp6	app,linux,container	rg-s	Running
asp7	app,linux,container	rg-s	Running
asp8	app,linux,container	rg-s	Running
asp9	app,linux,container	rg-s	Running
asp10	app,linux,container	rg-s	Running
asp11	app,linux,container	rg-s	Running
asp12	app,linux,container	rg-s	Running
asp13	app,linux,container	rg-s	Running
asp14	app,linux,container	rg-s	Running
asp15	app,linux,container	rg-s	Running
asp16	app,linux,container	rg-s	Running
asp17	app,linux,container	rg-s	Running
asp18	app,linux,container	rg-s	Running
asp19	app,linux,container	rg-s	Running
asp20	app,linux,container	rg-s	Running
asp21	app,linux,container	rg-s	Running
asp22	app,linux,container	rg-s	Running
asp23	app,linux,container	rg-s	Running
asp24	app,linux,container	rg-s	Running
asp25	app,linux,container	rg-s	Running
asp26	app,linux,container	rg-s	Running
asp27	app,linux,container	rg-s	Running
asp28	app,linux,container	rg-s	Running
asp29	app,linux,container	rg-s	Running
asp30	app,linux,container	rg-s	Running
asp31	app,linux,container	rg-s	Running
asp32	app,linux,container	rg-s	Running
asp33	app,linux,container	rg-s	Running
asp34	app,linux,container	rg-s	Running
asp35	app,linux,container	rg-s	Running
asp36	app,linux,container	rg-s	Running
asp37	app,linux,container	rg-s	Running
asp38	app,linux,container	rg-s	Running
asp39	app,linux,container	rg-s	Running
asp40	app,linux,container	rg-s	Running
asp41	app,linux,container	rg-s	Running
asp42	app,linux,container	rg-s	Running
asp43	app,linux,container	rg-s	Running
asp44	app,linux,container	rg-s	Running
asp45	app,linux,container	rg-s	Running
asp46	app,linux,container	rg-s	Running
asp47	app,linux,container	rg-s	Running
asp48	app,linux,container	rg-s	Running
asp49	app,linux,container	rg-s	Running
asp50	app,linux,container	rg-s	Running
asp51	app,linux,container	rg-s	Running
asp52	app,linux,container	rg-s	Running
asp53	app,linux,container	rg-s	Running
asp54	app,linux,container	rg-s	Running
asp55	app,linux,container	rg-s	Running
asp56	app,linux,container	rg-s	Running
asp57	app,linux,container	rg-s	Running
asp58	app,linux,container	rg-s	Running
asp59	app,linux,container	rg-s	Running
asp60	app,linux,container	rg-s	Running
asp61	app,linux,container	rg-s	Running
asp62	app,linux,container	rg-s	Running
asp63	app,linux,container	rg-s	Running
asp64	app,linux,container	rg-s	Running
asp65	app,linux,container	rg-s	Running
asp66	app,linux,container	rg-s	Running
asp67	app,linux,container	rg-s	Running
asp68	app,linux,container	rg-s	Running
asp69	app,linux,container	rg-s	Running
asp70	app,linux,container	rg-s	Running
asp71	app,linux,container	rg-s	Running
asp72	app,linux,container	rg-s	Running
asp73	app,linux,container	rg-s	Running
asp74	app,linux,container	rg-s	Running
asp75	app,linux,container	rg-s	Running
asp76	app,linux,container	rg-s	Running
asp77	app,linux,container	rg-s	Running
asp78	app,linux,container	rg-s	Running
asp79	app,linux,container	rg-s	Running
asp80	app,linux,container	rg-s	Running
asp81	app,linux,container	rg-s	Running
asp82	app,linux,container	rg-s	Running
asp83	app,linux,container	rg-s	Running
asp84	app,linux,container	rg-s	Running
asp85	app,linux,container	rg-s	Running
asp86	app,linux,container	rg-s	Running
asp87	app,linux,container	rg-s	Running
asp88	app,linux,container	rg-s	Running
asp89	app,linux,container	rg-s	Running
asp90	app,linux,container	rg-s	Running
asp91	app,linux,container	rg-s	Running
asp92	app,linux,container	rg-s	Running
asp93	app,linux,container	rg-s	Running
asp94	app,linux,container	rg-s	Running
asp95	app,linux,container	rg-s	Running
asp96	app,linux,container	rg-s	Running
asp97	app,linux,container	rg-s	Running
asp98	app,linux,container	rg-s	Running
asp99	app,linux,container	rg-s	Running
asp100	app,linux,container	rg-s	Running

Show data for last: 30 minutes 1 hour 6 hours 12 hours 1 day 3 days 7 days 30 days

Failed requests

Failed requests (Count)  
appinsightsall  
95

Server response time

Server response time (Avg)  
appinsightsall  
140.28 ms

Server requests

Server requests (Count)  
appinsightsall  
52.62 k

# Documentation

Markdown

Export HTML/PDF

Git

Github

Github pages

Read The Docs

# Tools

## - sum-up

---

VS Code, Eclipse

---

DBeaver, MySQL

---

Docker, Android emulator

---

Azure – ASP, ACR

---

Swagger, Chopper

---

Postman, SoapUI

---

Figma, Material Design

---

Git, Github, G. Desktop, G. Pages



# Future Works



Implement a new business process



Integrate more branch offices



# Conclusion

"During project realization I had used Figma application in order to design views, which was a new experience for me. Due to Figma's useful prototyping features, I plan to use this tool in the future as well."



# Conclusion

"It was a joyous moment to realize data exchange and synchronization between two APIs and observing that the setup works as it is supposed to work despite the fact that, I never programmed in java spring boot."



# Conclusion

"Swagger is an amazing tool, and this is more than "swagger-ui.html" which I knew before implementing this project. Swagger hub can help manage our API for different applications and different versions of API. Using Swagger we can design API in YAML before development stage. Swagger can generate not only client code, moreover can generate server side code."



# Conclusion

"Spring boot is a nice framework for API development and backend applications as it has made easier to fast create a web application. Using `swagger-ui.html` endpoint we can rapid test backend service inside browser - we do not need postman, curl or SoapUI for it. "

# Acknowledgment



Special thanks to Mr. Wiktor Wandachowicz for his guidance, conveyed lectures and support during the project realization.



Thanks for questions and feedback from the other groups.