Private Kit Service
Author: Prithvi K Padmanabhan

Context

This document illustrates the backend service architecture for private kit applications such as safe paths and other applications that want to utilize the backend service provided by private kit. The document provides the deployment architecture as well as the implementation details of the service.

Table of Content

Principles

The private kit service is built upon principles of scalability, maintainability, extensibility and cloud agnostic using relevant tools to make it easier for developers to deploy and upgrade the service. All the services will be exposed via REST API and will have authentication built in for each of the methods

Tools used

Option 1

- *Kubernetes - Container Orchestration*
- *Docker - Container & JIB*
- *Heroku and AWS*

Option 2

- *Micronaut: https://micronaut.io/ (Serverless as well standard server supported)*

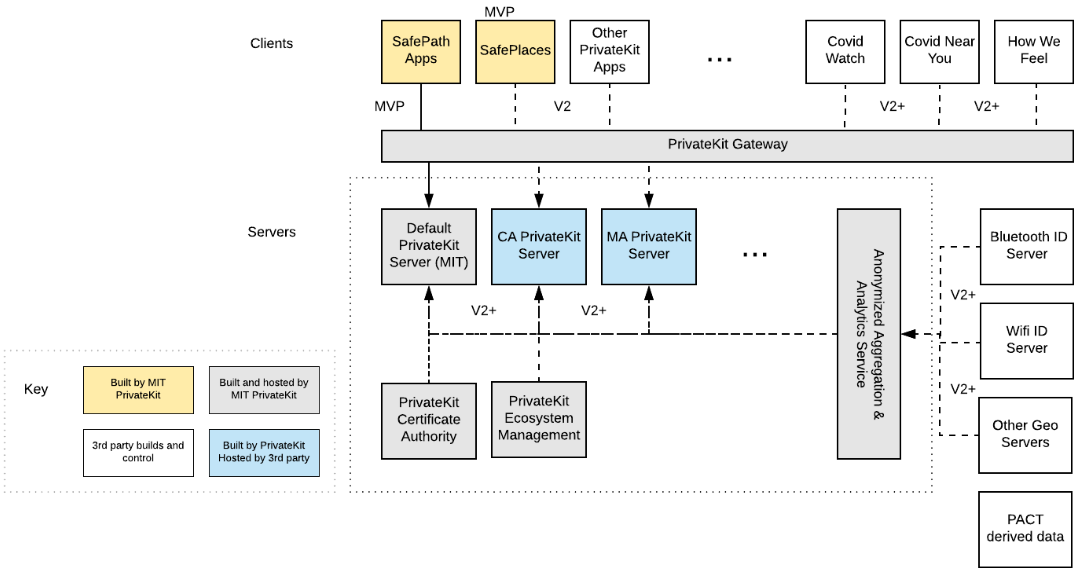
Common

- *Programming Language - Java*
- *Library - Spring Boot, JPA*
- *Health check - Kubernetes Probes*
- *Database - Postgres*

Application Flow

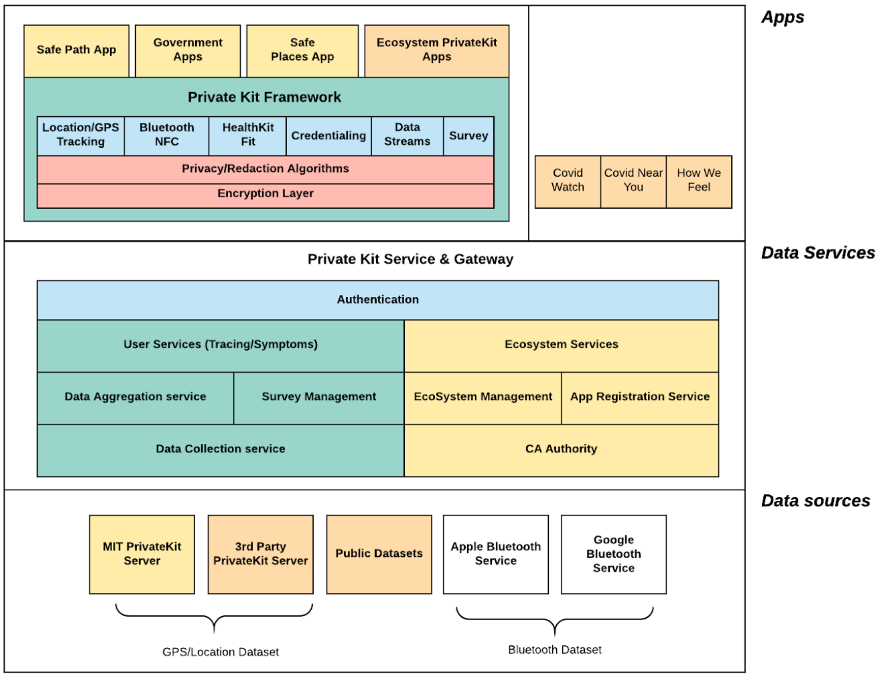
1. Private Kit Admin creates an app in the database (e.g. SafePath App)
2. The App Admin creates the survey (POST Survey API)
3. The App (e.g. SafePath App) gets the survey from the server (GET Survey API) and displays it on the device
4. The user using the app (e.g. SafePath App ios/android) takes the survey and submits it to the server authenticated using HCaptcha (POST Survey Response)

Service Interaction Design

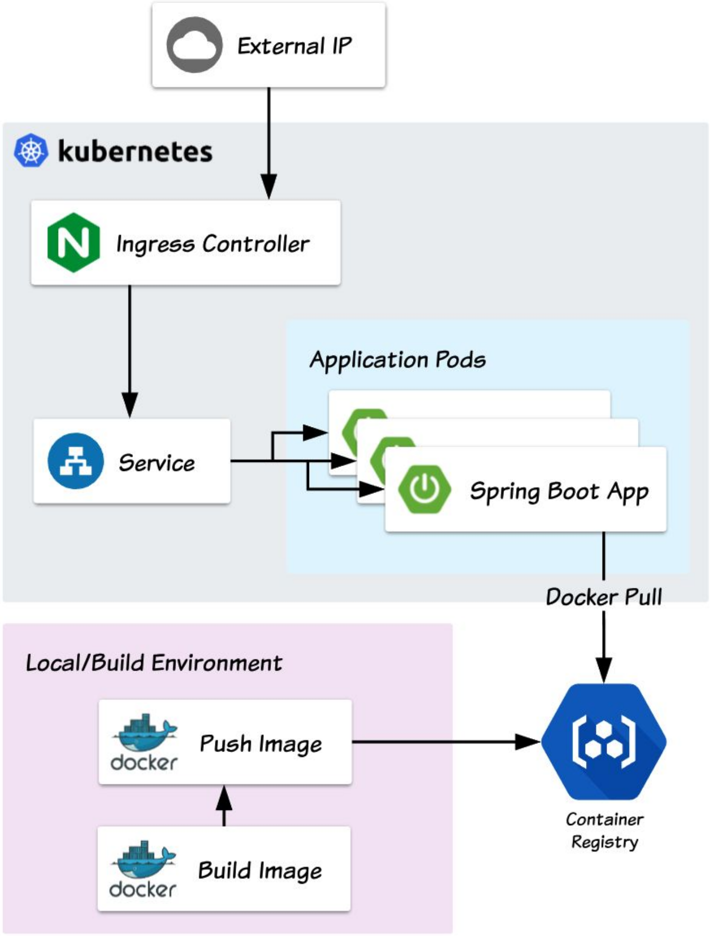


Component Design

Private Kit Cohesive Architecture



Infrastructure Design



Services

The following modules will be part of the Private Kit Service:

- **Administration**
  App Registration: The app registration end point will allow any application developer of the Private kit application to register their app. However all apps when registered will go into pending status. MIT team will review the app and approve the registration.
- **Authentication**
  - For **every** upload, we will launch into captcha flow
  - And API will include Captcha code for the upload API
- **Self Reporting**
  Self Reporting feature provides a wellness check/symptoms check for the user to provide based on a survey. At the end of the survey the user can upload the location trail that is fuzzed to the private kit service. Multiple endpoints will be provided for the user and app administrator to administer the survey and the app to render the survey.

Services Specification (WIP)

Survey API Specification

RAML Spec OAS Spec

/services/v1.0/{app_id}/survey

```
method: GET
auth: JWT
scope: app

response:
{
 "data": [
  {
   “id”: 1234,
   "name": "Symptoms Checker Survey",
   "description": "Symptoms Checker Survey",
   "image": "",
   "questions": [
    {
     "question_key": "1",
     "question_text": "Select your symptoms?",
     "image": "",
     "question_type": "MULTI",
     "required": true,
          "screen_type": "Checkbox",
          "option_key": "option_1",
          "conditions": [
            {
              "response": "Y",
              "jump_to_key": "3"
            },
            {
              "response": "N",
              "jump_to_key": "2"
            }
          ]
        },
        {
          "question_key": "2",
          "question_type": "END",
          "screen_type": "Alert"
        },
        {
          "question_key": "3",
          "question_type": "END"
        }
      ],
      "options": [
        {
          "key": "option_1",
          "values": [
            {
              "label": "Yes",
              "value": "Y",
              "description": ""
            },
            {
              "label": "No",
              "value": "N"
            },
            {
              "label": "Maybe",
              "value": "M"
            }
          ]
        }
      ],
      "screen_types": [
        "Checkbox",
        "911"
      ]
    }
  ]
}
```

```
method: POST
auth: Bearer {CAPTCHA}
scope: app admin

body:
{
  "name": "Symptoms Checker Survey",
  "description": "Symptoms Checker Survey",
  "image": "",
  "questions": [
    {
      "question_key": "1",
      "question_text": "Select your symptoms?",
      "image": "",
      "question_type": "MULTI",
      "required": true,
      "screen_type": "Checkbox",
      "option_key": "option_1",
      "conditions": [
        {
          "response": "Y",
          "jump_to_key": "3"
        },
        {
          "response": "N",
          "jump_to_key": "2"
        }
      ]
    },
    {
      "question_key": "2",
      "question_type": "END",
      "screen_type": "Alert"
    },
    {
      "question_key": "3",
      "question_type": "END"
    }
  ],
  "options": [
    {
      "key": "option_1",
      "values": [
        {
          "label": "Yes",
          "value": "Y",
          "description": ""
        },
        {
          "label": "No",
          "value": "N"
        },
        {
          "label": "Maybe",
          "value": "M"
        }
      ]
    }
  ],
  "screen_types": [
    "Checkbox",
    "911"
  ]
}

response:
{
  "info": "survey was created successfully."
```

```
}
```

/services/v1.0/{app_id}/survey/{survey_id}/response

```
method: POST
auth: Bearer {CAPTCHA}
scope: app

body:
[
  {
    “question_key”: 12345,
    "response": ["Great"],
    "skipped": true
  }
]
```

HCaptcha

https://fathomless-mountain-29102.herokuapp.com/v1.0/captcha