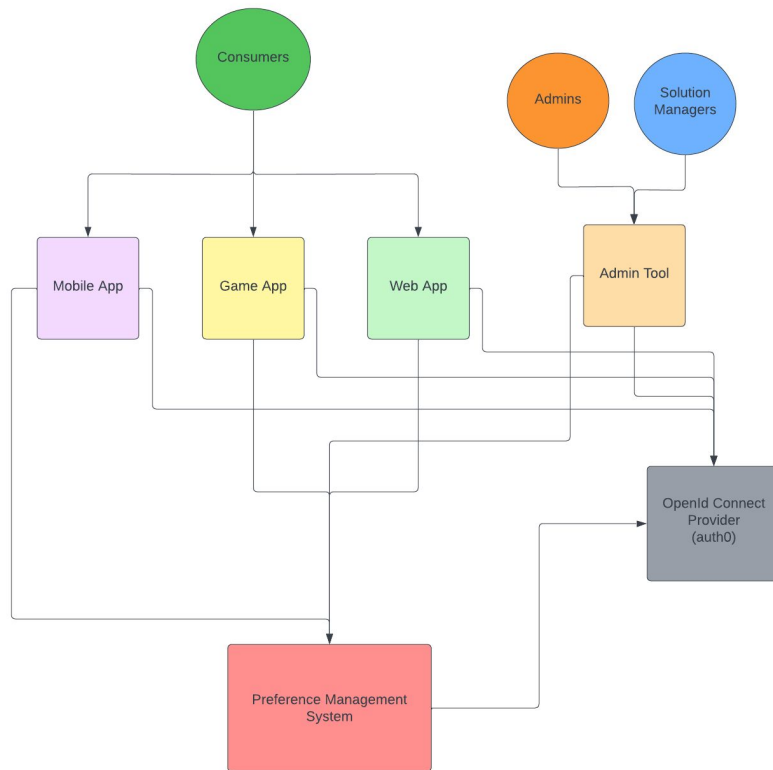


The background is a solid red color with a pattern of overlapping triangles in various shades of red, orange, and yellow. Some triangles are solid, while others are outlined in white. The triangles are arranged in a way that creates a sense of depth and movement, with some appearing to be in the foreground and others receding into the background.

# Preference Manager

**Nikos Larios**

# Preference Manager Use Case

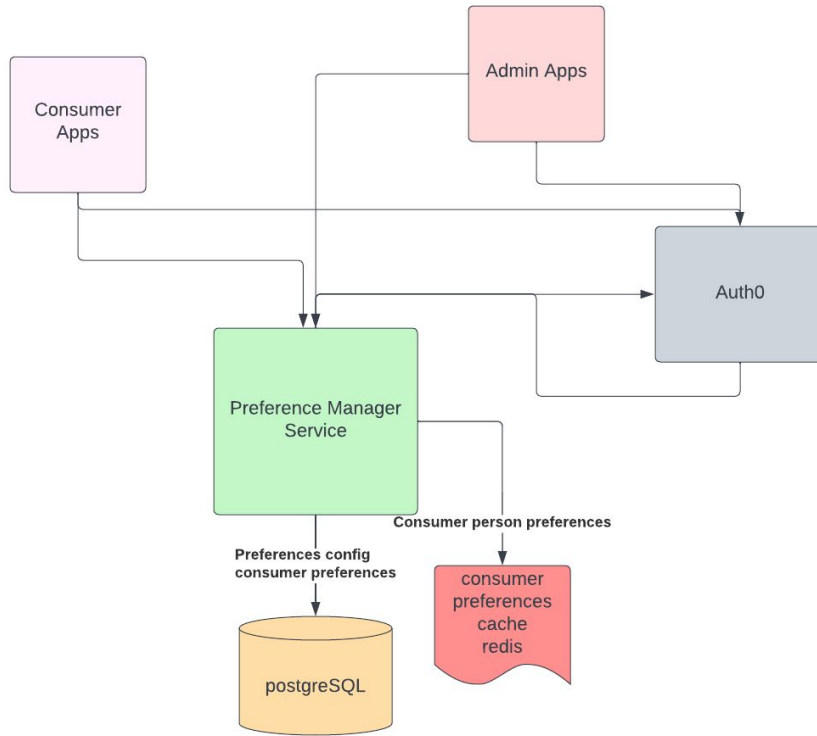


Centralized platform for preference management.

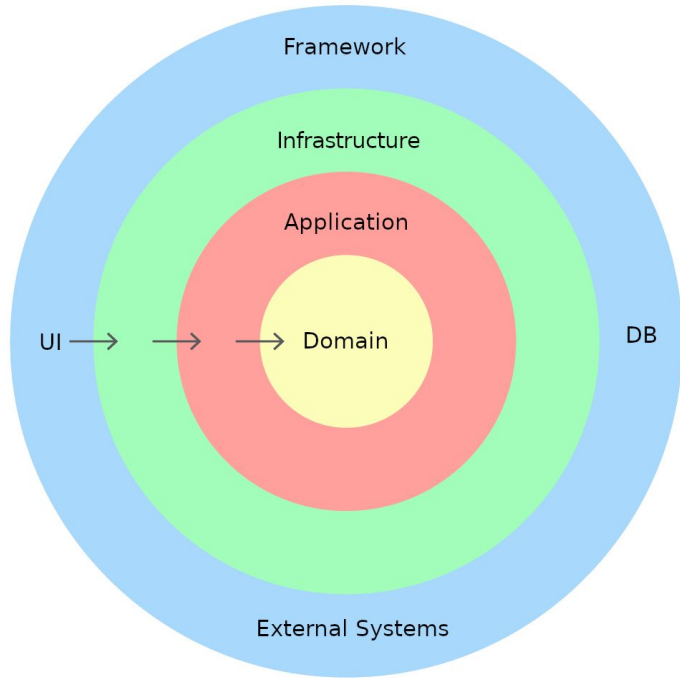
The system provides an API that allows authorised Apps to create Preferences configurations and also fills these preferences with Consumer data.

Client services authenticate the users and use JWT tokens to access preference-manager

# Architecture



# Service architecture



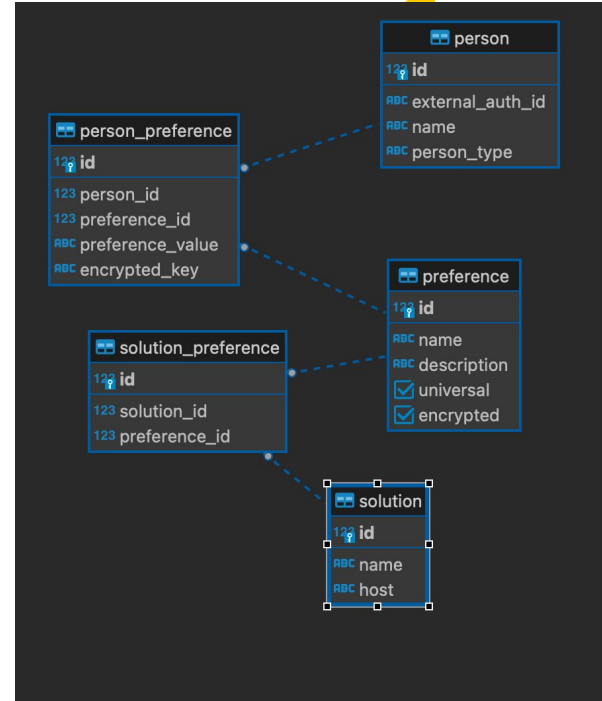
It has the following layers (class libraries):

- API
- UseCase (Application)
- Domain
- Infrastructure

# Database schema and Domain

## DOMAIN

- Preference (preference configuration)
- Person (User)
- Solution (Solution that preferences apply to)
- PersonPreference (Consumer preferences with values)



# API design

## PersonPreference

GET	/persons/{externalAuthId}	✓	🔒
POST	/person-preferences	✓	🔒
PUT	/person-preferences/{id}	✓	🔒

## SolutionManagerPreferences

POST	/solution-preferences	✓	🔒
POST	/solution-preferences/{id}	✓	🔒
GET	/solution-preferences/solutions/{id}	✓	🔒

## UniversalPreference

GET	/universal-preferences	✓	🔒
POST	/universal-preferences	✓	🔒
PUT	/universal-preferences/{id}	✓	🔒



# Scopes and Roles

## List of Permissions (Scopes)

These are all the permissions (scopes) that this API uses.

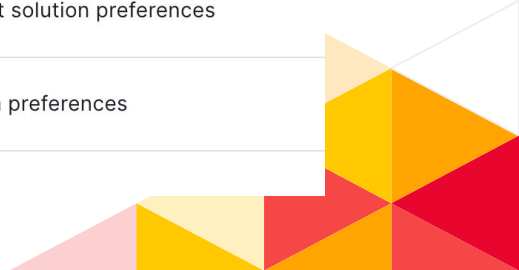
Permission	Description
<code>read:universal_preferences</code>	Read universal preferences
<code>read:preferences</code>	Read preferences
<code>edit:universal_preferences</code>	Read and edit universal preferences
<code>read:consumer_preferences</code>	Read values of consumer preferences
<code>edit:consumer_preferences</code>	Read and edit consumer preferences
<code>edit:solution_preferences</code>	Read and edit solution preferences
<code>read:solution_preferences</code>	Read solution preferences

## Roles

- UniversalAdmin
- SolutionManager
- Consumer

I used only permissions to handle authorization

But in a future work can work with role, once the authentication is handled by preference-manager service



# Acceptance Tests

The screenshot displays the Postman interface with the 'Runner' tab selected. The left sidebar shows a collection named 'Acceptance Tests Preference Manager' with three sub-items: 'Admin', 'Solution Manager', and 'consumer'. The main area shows the 'RUN ORDER' list with five tests, all of which are checked for execution:

- ☒ > **POST** Authenticate admin
- ☒ > **POST** Add universal preference
- ☒ > **GET** get all universal preferences
- ☒ > **POST** Authenticate solution manager
- ☒ > **POST** Add solution preference
- ☒ > **GET** Add person preference

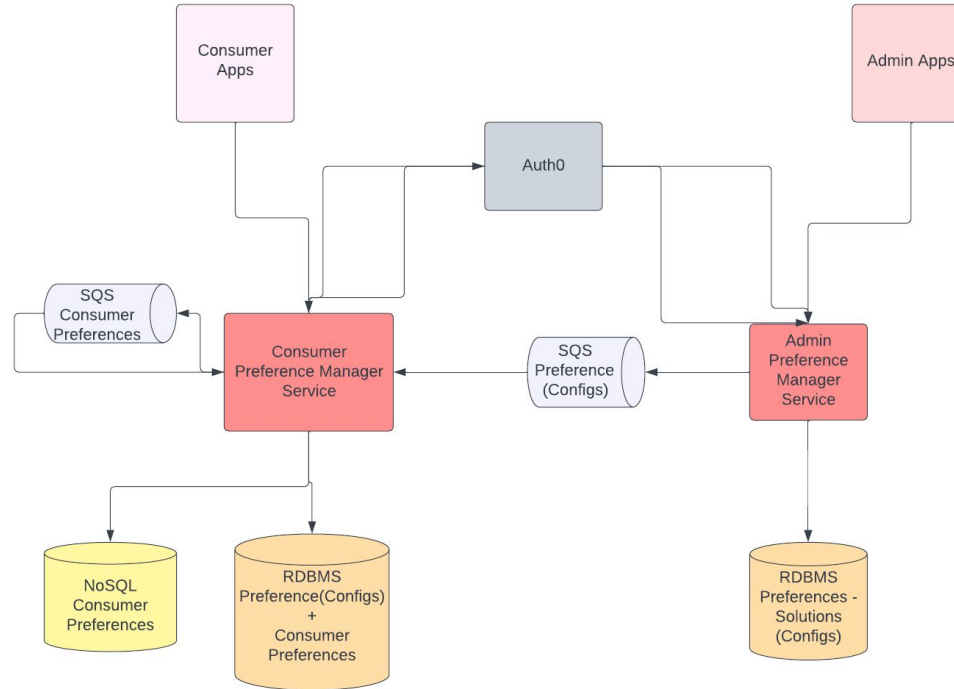
On the right, the 'Run Settings' panel is visible, showing the following configuration:

- Iterations:** 1
- Delay:** 0 ms
- Data:** Select File
- Advanced settings:**
  - ☐ Save responses ⓘ
  - ☒ Keep variable values ⓘ
  - ☐ Run collection without using stored cookies
  - ☒ Save cookies after collection run ⓘ

A red button at the bottom right of the 'Run Settings' panel reads 'Run Acceptance Tests Preferenc...'.



# Future work and Optimisation



**Thank you**