Ebike application

Requirements and specifications

It is required to create an application that allows users to register themselves and then use one of the electric bikes that are in the environment to start a ride, moving the bike's position.

Users can register themselves and use a bike from a remote GUI, and the administrators can create electric bikes from another remote GUI.

There should be data persistence.

It should be possible to add new functionalities available to the users dynamically.

My specific requirements

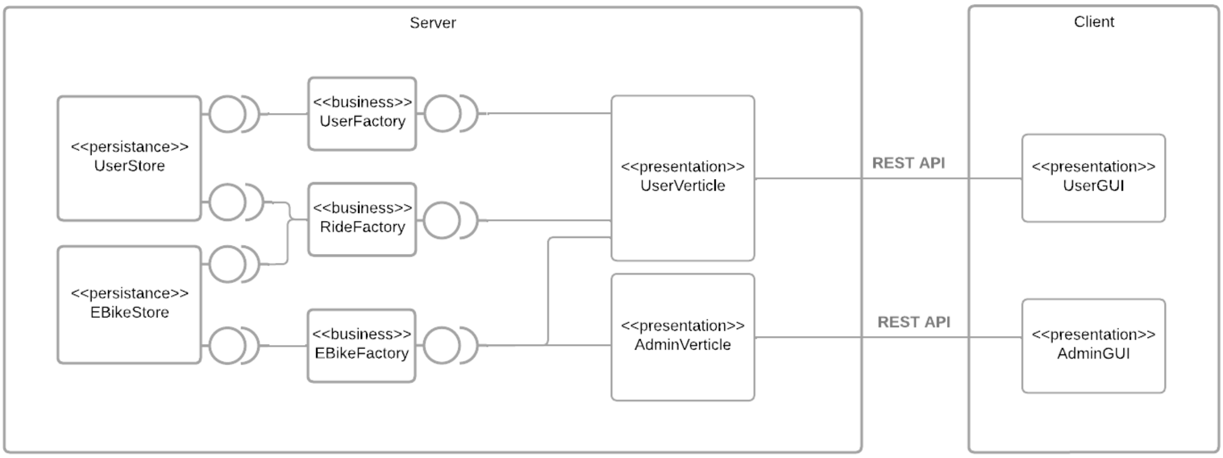
To create persistency, it has been decided to allow the user to save its data, to load its data, and to load and change the electric bikes data (with a ride).

The admin can load, save and change the electric bikes data.

Architecture

The architecture chosen is a layered architecture: there will be a Presentation layer that updates itself when requested by the user or the admin to stay updated with the Business layer, where there will be all the functionalities that will work on the data of the Persistence layer.

The Presentation layer uses a REST API to send data to users and admin and receive commands, like creating an eBike, creating a user, starting a ride or stop it.



Quality attributes scenarios

Availability:

If a user (Source) creates a user already existing (Stimulus) on the system (Artifact) in normal operations (Environment) the system informs the user (Response) with no downtime (Response Measure).

The same as above works with creation of eBikes from the admins.

Consistency, Scalability:

- If more users (Source) start riding some bikes (Stimulus) on the system (Artifact) in normal operations (Environment) the information stored will be the same (Response) regardless of how many bikes or users are being modified (Response Measure).

Fitness functions

It has been used a fitness function to assert that the layered architecture has been followed correctly: The presentation layer can only use the business layer, that can use just the persistence layer that should not use any other layer.