Bike

+ ID: int
+ _Bikes: list<Bike>_
- location: Coordinate
- isLocked: bool
+ time: long

- connectToInternet(): void
+ _closestBike(user: Coordinate): Bike_
- openLock(): void
+ isLocked(): bool
+ getPosition(): Coordinate
+ reserve(): void
+ calculateCharge(type: int): double
+ notifyTimingOut(): void
- freeBike(): void

Coordinate

+ longitude: double
+ latitude: double

+ _distance(a: Coordinate, b: Coordinate): double_

User

- location: Coordinate
+ username: string
- password: string
- paypal: Account
+ typeofPlan: int

+ register(username: string, password: string, acc, Account): void
- validateMail(username: string): bool
- validatePassword(password: string): bool
+ login(): void
+ choosePlan(type: int): void
- pay(quantity: double): void
- chooseBike(): int
- leaveBike(): void
- takeBike(): void

Account

+ card: string
- nip: int

+ makePayment()

Luis Eduardo
Robles
Jiménez

1   1   1   1   0..1   0..1   1   1

Luis Eduardo Robles Jiménez

Open Bikes

Registered user

Non-registered user

- Access
- Register account
- Choose plan
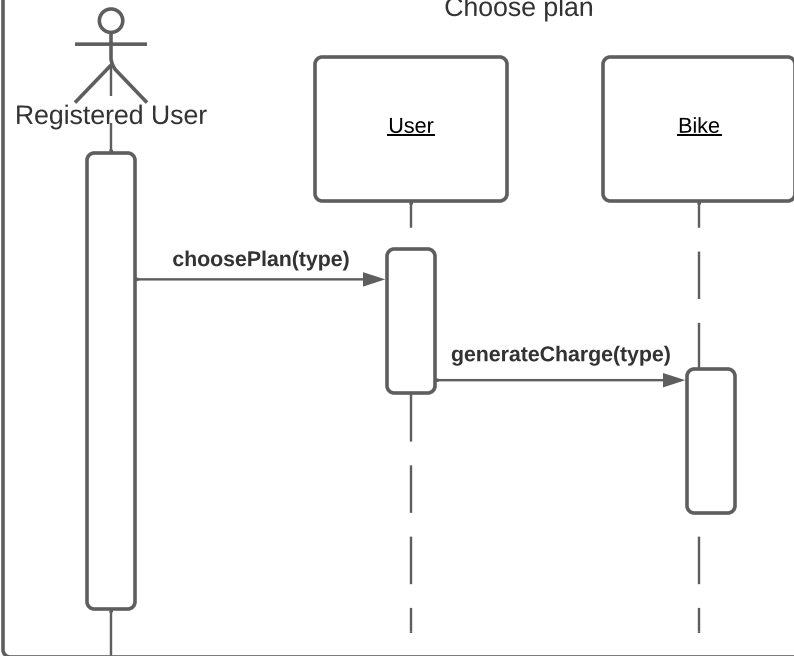- Find bike
- Drop bike

User
- Registered User
- Non-registered user

Nombre: Access
Autor: Luis Eduardo Robles Jiménez
Fecha: 04 de Marzo del 2022
Descripción: Starts the account of a user.
Actores: Registered User
Precondiciones: The user needs an existing account in the platform.
Flujo Normal: An user provides their credentials and the system opens the account
Flujo Alternativo: If the user doesn't have an account, the page is redirected to register.
Poscondiciones: Logging out is a safe practice.

Nombre: Register Account
Autor: Luis Eduardo Robles Jiménez
Fecha: 04 de Marzo del 2022
Descripción: Creates new account of the user.
Actores: Non-registered user.
Precondiciones: The user has to be new in the platform.
Flujo Normal: A new email address and a verified password are provided and the system creates a new account.
Flujo Alternativo: If the user already exists, the session will log in instead.
Poscondiciones: It's not neccesary to use this function again.

Nombre: Choose plan
Autor: Luis Eduardo Robles Jiménez
Fecha: 04 de Marzo del 2022
Descripción: The user is able to choose between 3 plans depending on the time they are going to use the bike.
Actores: Registered User
Precondiciones: The information of the current account is already validated.
Flujo Normal: The user chooses one between the three plans through the application and it calculates the respective charge.
Flujo A\lternativo:
Poscondiciones: The prices may change if the time is exceeded.
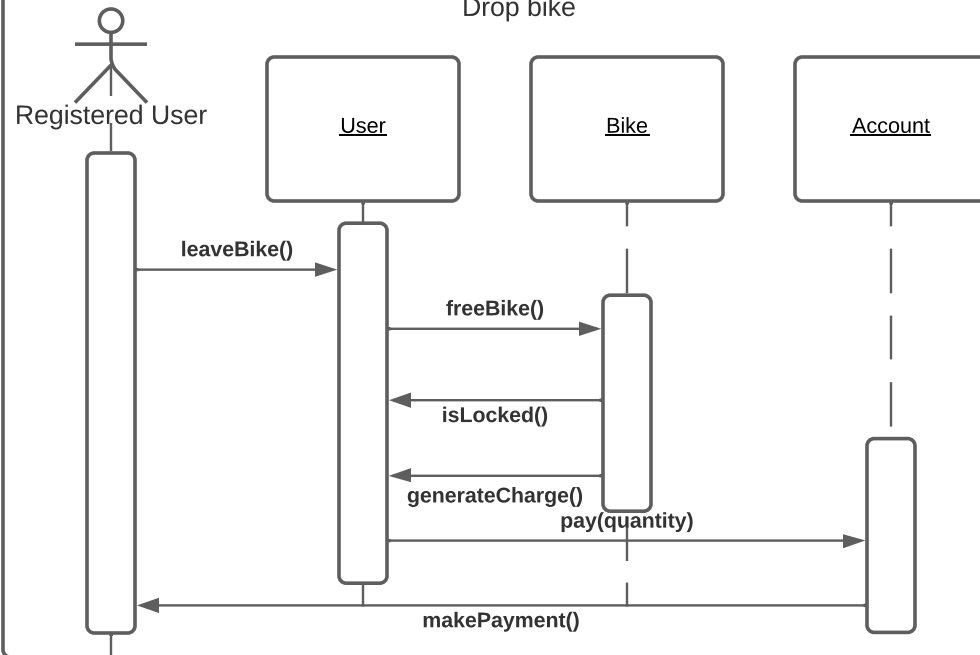

Nombre: Find bike
Autor: Luis Eduardo Robles Jiménez
Fecha: 04 de Marzo del 2022
Descripción: Once the user joins the app, the closest bike shows up.
Actores: Registered user
Precondiciones: A plan of use has already been chosen.
Flujo Normal: The closest bike is shown in the map for the user and they choose it.
Flujo Alternativo: Once the user finds the bike, they are able to reserve it for 5 minutes.
Poscondiciones:

Nombre: Drop bike
Autor: Luis Eduardo Robles Jiménez
Fecha: 04 de Marzo del 2022
Descripción: Required actions when the user stops using the bike
Actores: Registered user
Precondiciones: The user has a bike that is not locked.
Flujo Normal: After the bike is manually locked, the system asks for confirmation of leaving the bike.
Flujo Alternativo: If the user doesn't lock the bike, more charges will be applied.
Poscondiciones:

Choose plan

Registered User

| User | Bike |

Registered User → User: **choosePlan(type)**

User → Bike: **generateCharge(type)**

Find Bike

Registered User

| User | Bike | Coordinate |

Registered User → User: **chooseBike()**

User → Bike: **closestBike(location)**

Bucle
Iterate bikes

Bike → Coordinate: **distance(bLoc, uLoc)**

Coordinate → Bike: **Bike**

Bike → User: **IDofBike**

Drop bike

Registered User

| User | Bike | Account |

Registered User → User: **leaveBike()**

User → Bike: **freeBike()**

Bike → User: **isLocked()**

Bike → User: **generateCharge()**

User → Account: **pay(quantity)**

Account → Registered User: **makePayment()**