SAFERIDE

Judith Brener, Roy Reinhorn, Eden Barak

ביה"ס למדעי המחשב ע"ש בלווטניק
הפקולטה למדעים מדויקים
ע"ש ריימונד ובברלי סאקלר
אוניברסיטת תל אביב

TAU CS 0368-3527-01
IoT Workshop, Spring 2019
Sivan Toledo and Nir Levi

OUTLINE

The problem

The product

The purpose

App Demo

How does it work

THE PROBLEM

- Electric bikes, nowadays, are used vastly, by all ages
- **There is limited monitoring (if any)**
  - Increasing number of accidents among teenagers
  - **Increasing number of death cases among teenagers**
- Popularity gains: cheaper bikes and better infrastructures (hopefully)

STATISTICS

- Number of casualties rises;
  - 2015- 692
  - 2018- 2185
- Number of deadly accidents rises
  - 2015- 2 deaths
  - 218- 18 deaths (4 under 16)

DEMO VIDEO

https://youtu.be/lQ-5QJiAupc

THE PRODUCT - COMPONENTS

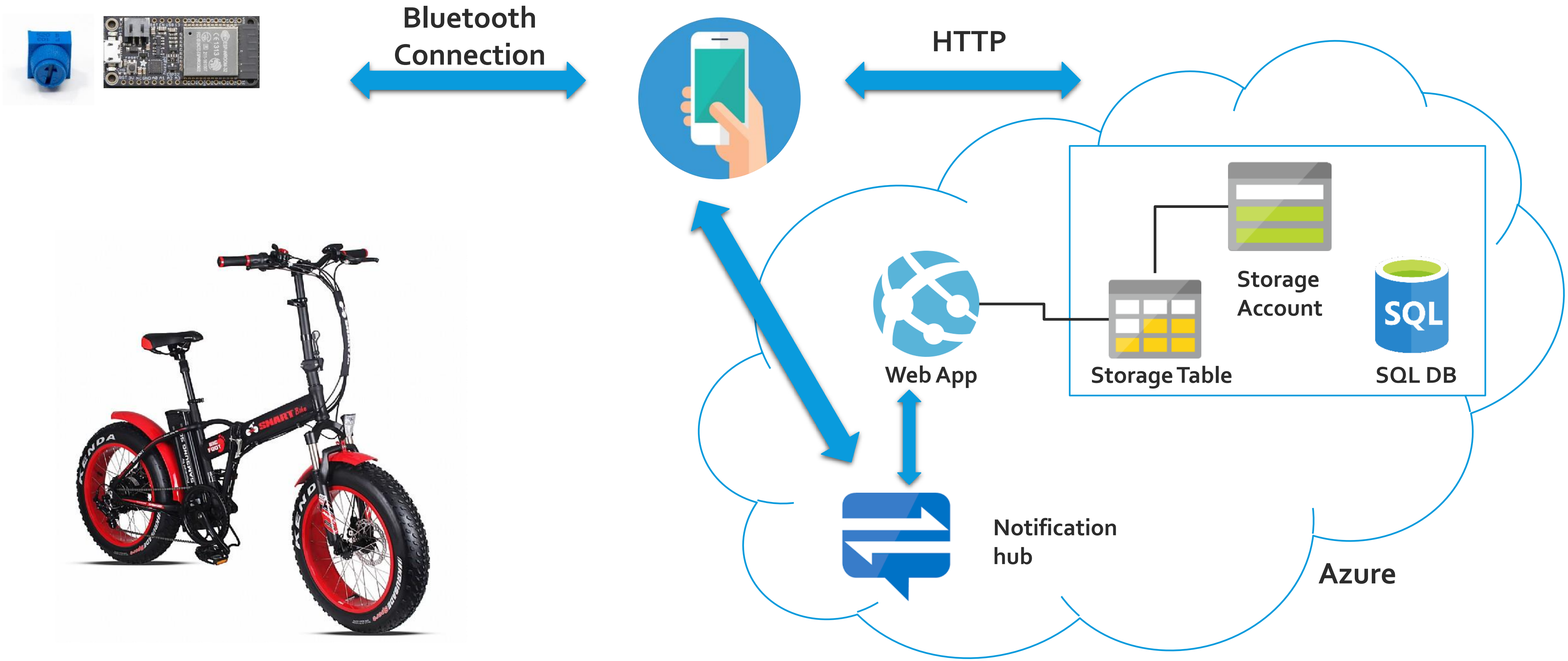
Hardware

- The device consists of Adafruit Feather 32u4 Bluefruit LE board.
- Battery usage, Activate bike battery
- Accelerometer – based on phone (fall back): identify falls
- GPS – based on phone, enables geo-fencing and ride data

Software

- Xamarin based app
- Arduino libraries
- Azure cloud services

HOW DOES IT WORKS - WORKFLOW



Bluetooth Connection

HTTP

Web App

Storage Account

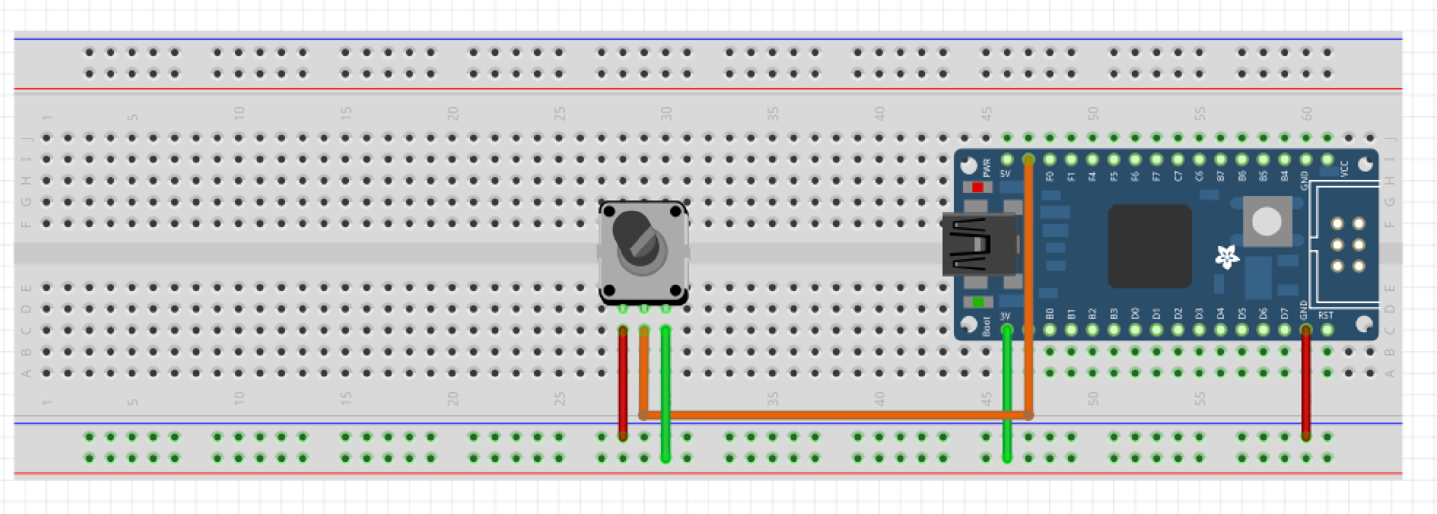
Storage Table

SQL DB

Notification hub

Azure

HARDWARE – THE PARTS



Adafruit Bluefruit 32u4

Bulb

- Mock on/off of the battery

Potentiometer

- Mock the voltage inside the battery

BLE CONNECTION

Using UART service.

- **UUID**: 6E400001-B5A3-F393-E0A9-E50E24DCCA9E

UART connection

- The service simulates basic UART connection over two lines; TX and RX

The service include two characteristics:

- **TX (0x0002)** - written to by the connected Central device.
- **RX (0x0003)** - used to send data from the peripheral device to the connected Central device.

BLE CONNECTION

Central device scans for Bluetooth low-energy device.

- Blinking Red Led symbols BLE existence.

Connects to our Arduino device

- recognizable by unique prefix name
- Blue Led lights when connection established

Central Device invokes event

- listening to data send from the arduino device (battery percentage change).

Upon start/end ride

- central device sends turn on/off bike instruction to the peripheral device, respectfully.

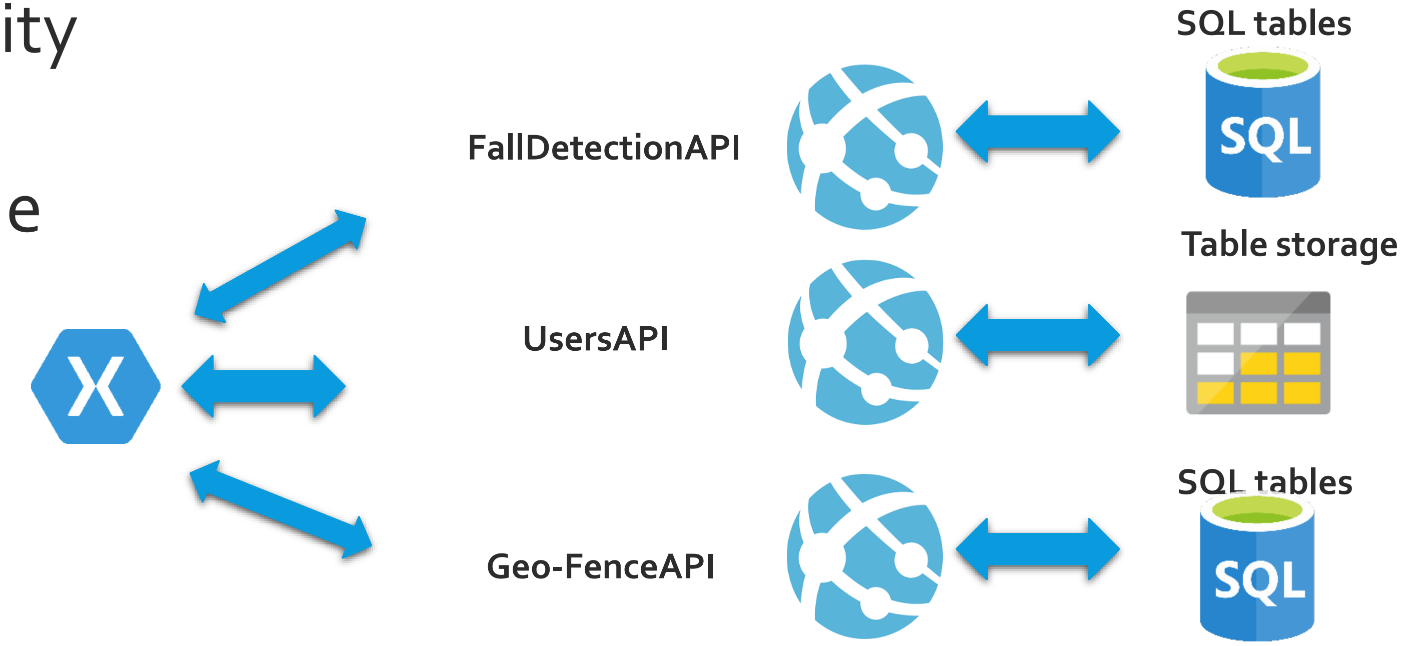
HOW DOES IT WORK – MICRO SERVICES

Each micro-service is:

- Independent - functionality
- Independent - DB
- Independently deployable

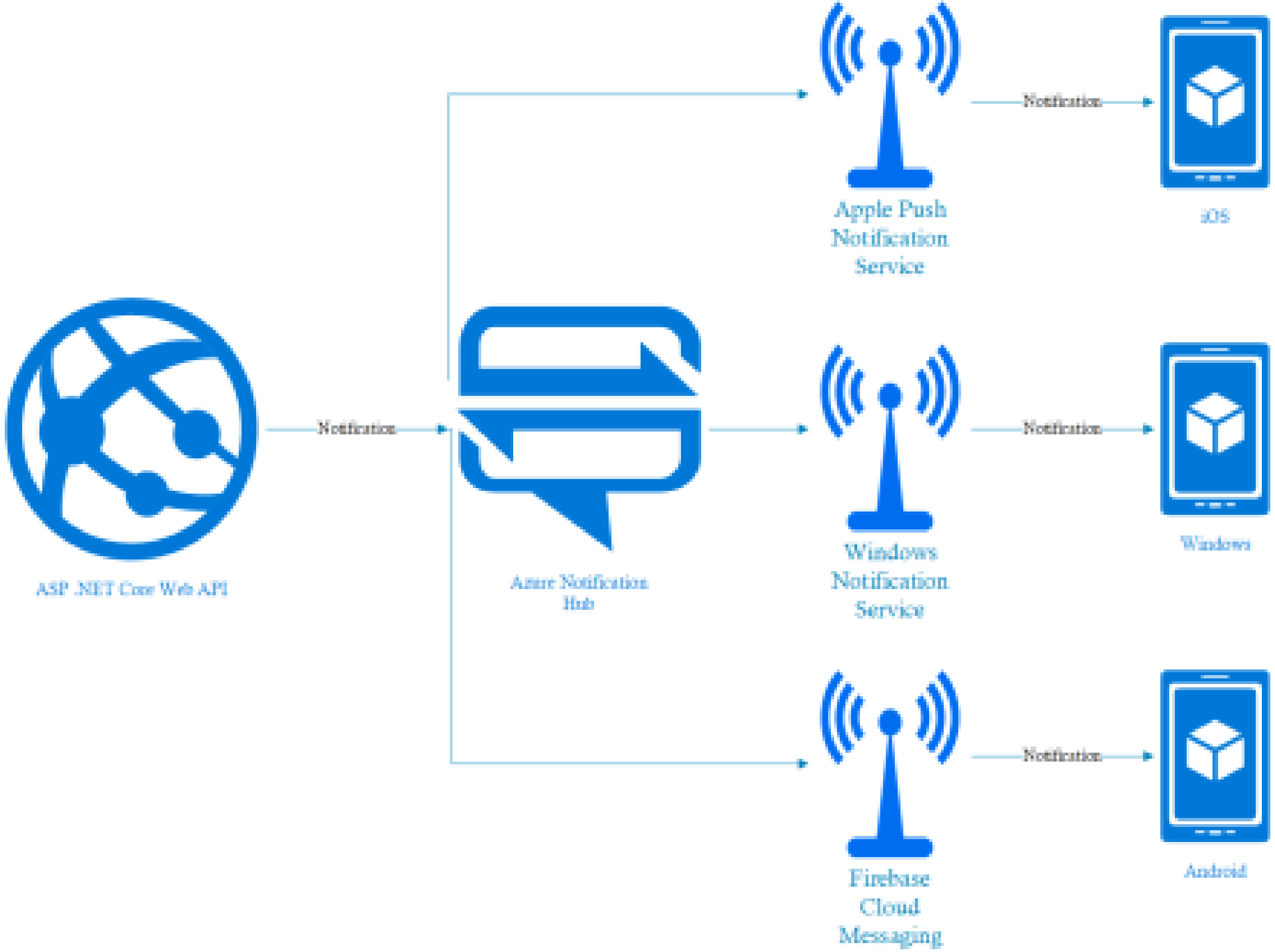
Allows:

- Easy maintenance
- Simplicity
- Fast development



NOTIFICATIONS

- To the “Parent” (admin)
- When fall detected
- When exiting the geofence



SERVICES OUTLINE

User management

Smart lock

Buttery usage

Fall detection

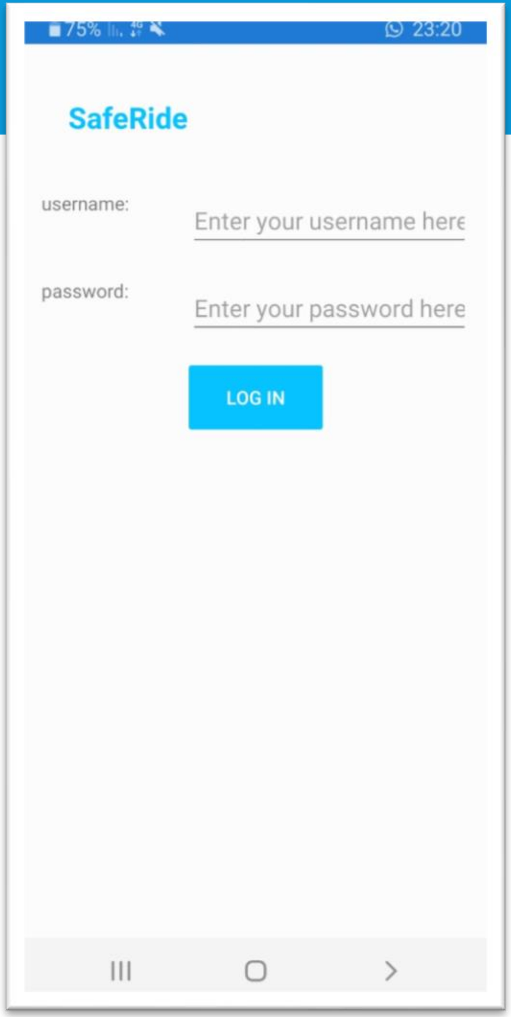
Geofance

USERS MANAGEMENT

- **Multiple users can use the same device**

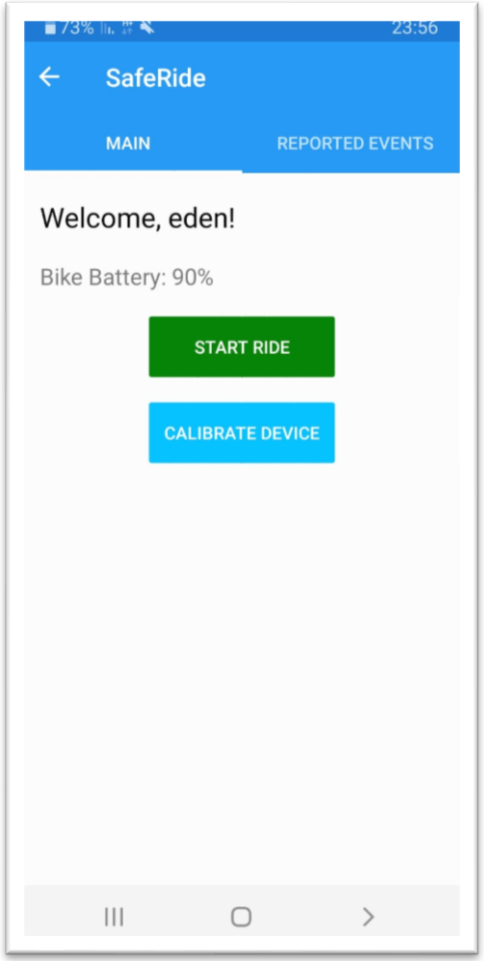
- **“Parent” (Admin) users**
- First Login (no users associated to current device ID) creates new user with admin privileges.

- **Login management:**
- Server Data Base
- Hashed Passwords transmitted (SHA-256)



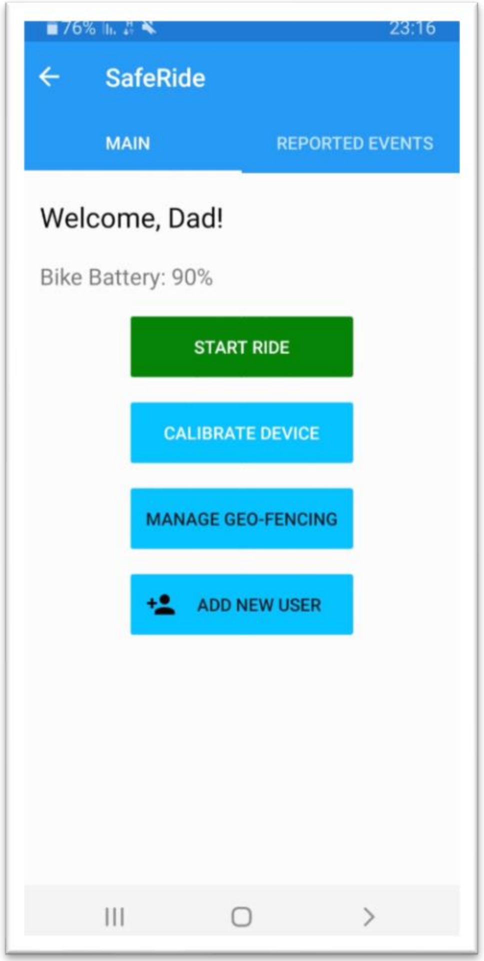
USERS API

- Manages user's tables
- Authorization
- Licenses



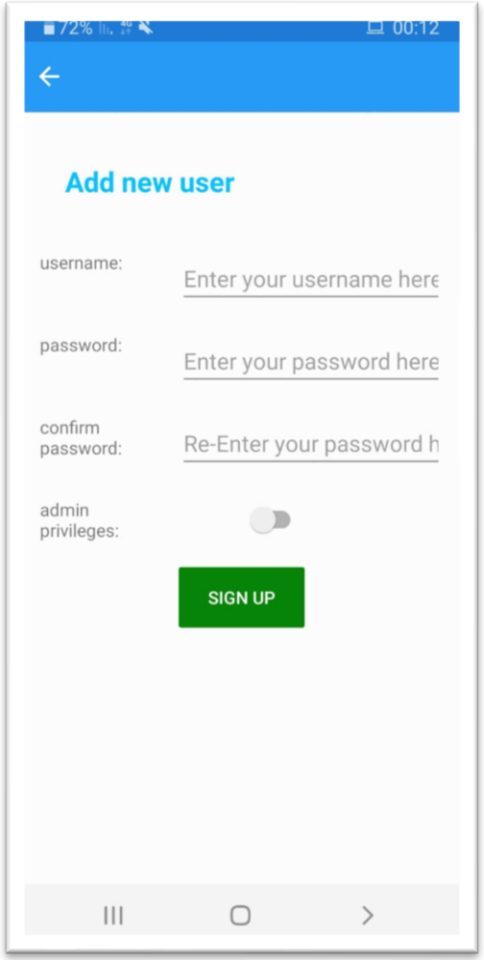
Welcome, eden!

Bike Battery: 90%



Welcome, Dad!

Bike Battery: 90%



Add new user

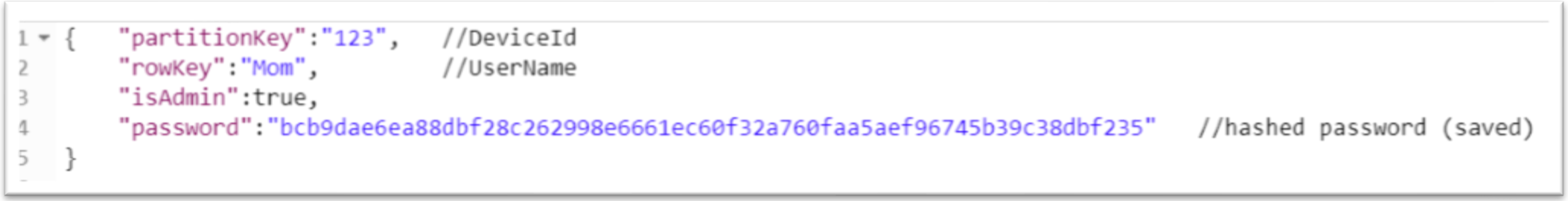
username: Enter your username here

password: Enter your password here

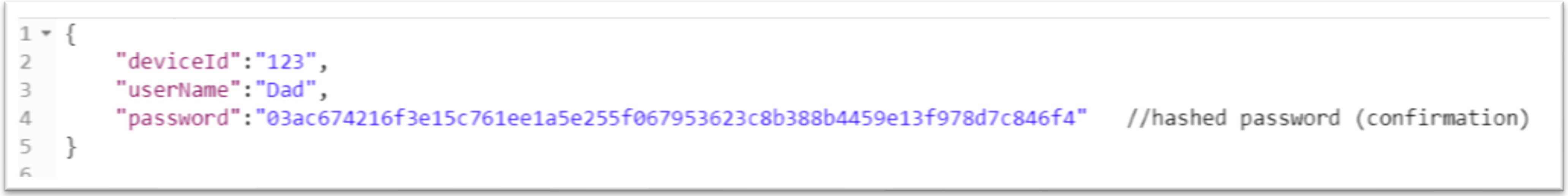
confirm password: Re-Enter your password h

admin privileges:

USERS API - JSONS

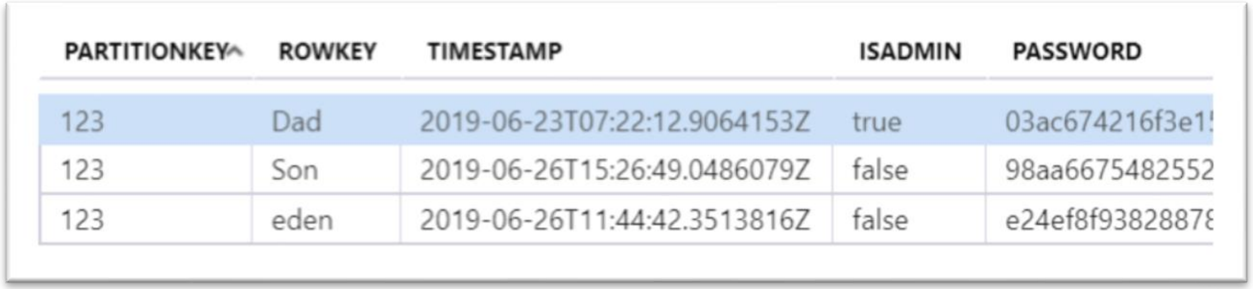


```
{   "partitionKey":"123",    //DeviceId
    "rowKey":"Mom",          //UserName
    "isAdmin":true,
    "password":"bcb9dae6ea88dbf28c262998e6661ec60f32a760faa5aef96745b39c38dbf235"   //hashed password (saved)
}
```



```
{
      "deviceId":"123",
      "userName":"Dad",
      "password":"03ac674216f3e15c761ee1a5e255f067953623c8b388b4459e13f978d7c846f4"   //hashed password (confirmation)
}
```

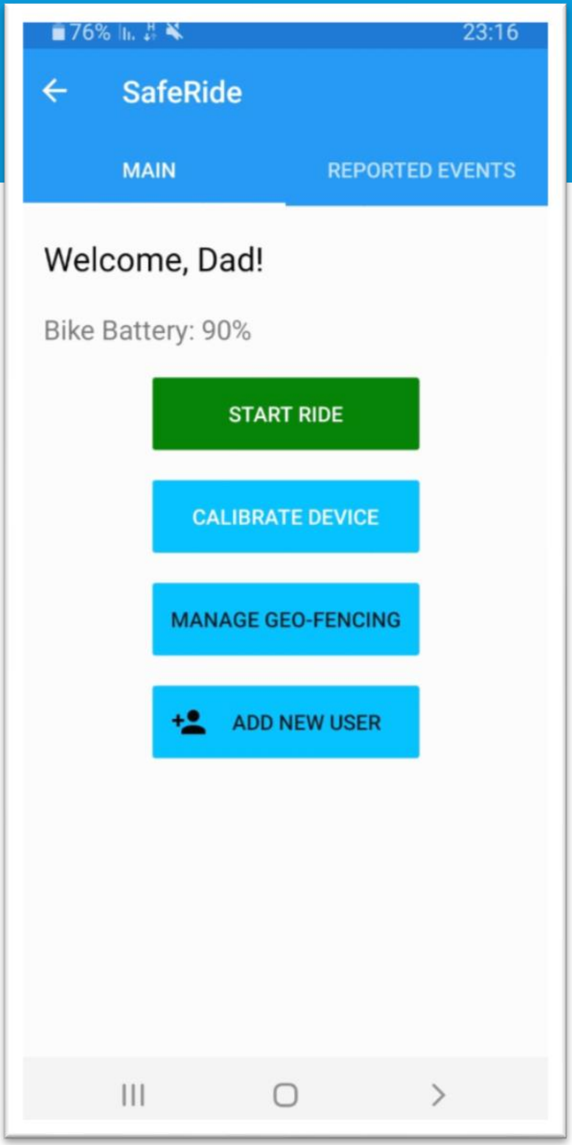
USERS API - TABLES



PARTITIONKEY^	ROWKEY	TIMESTAMP	ISADMIN	PASSWORD
123	Dad	2019-06-23T07:22:12.9064153Z	true	03ac674216f3e15
123	Son	2019-06-26T15:26:49.0486079Z	false	98aa6675482552
123	eden	2019-06-26T11:44:42.3513816Z	false	e24ef8f93828878

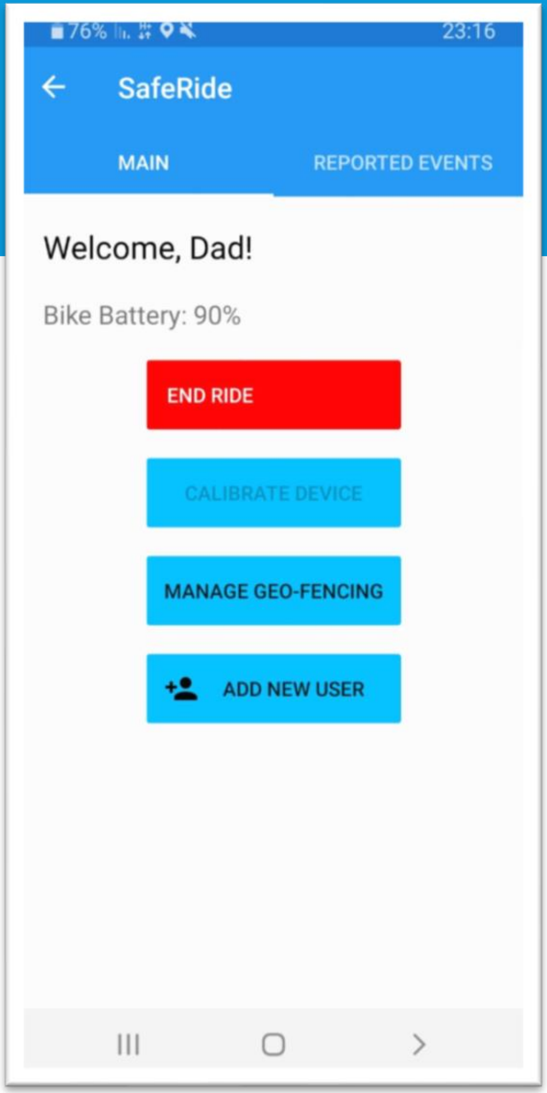
SMART LOCK

- Enable engine via application
- Multiple users
- No need for a key



Welcome, Dad!

Bike Battery: 90%

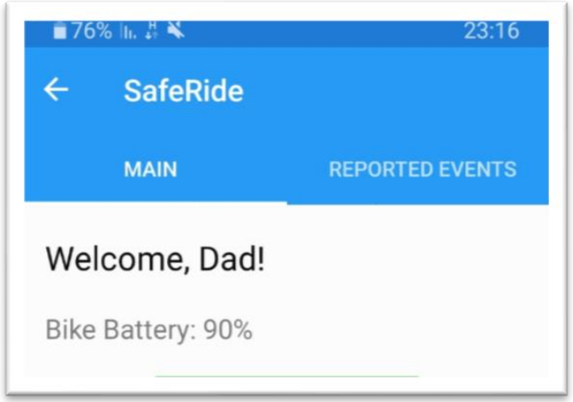


Welcome, Dad!

Bike Battery: 90%

BUTTERY USAGE

- User cab see the buttery percentage
- //Supply estimated distance according to percentage



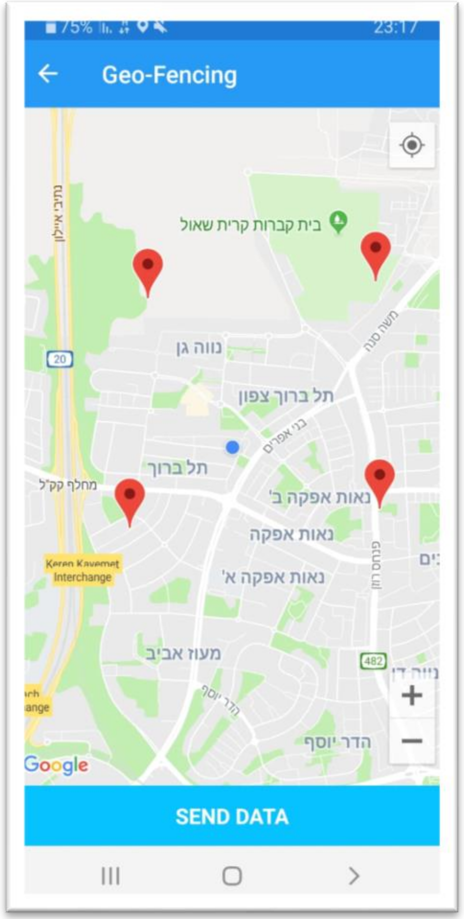
Welcome, Dad!

Bike Battery: 90%

GEOFANCE

Safety

- Limits riding area

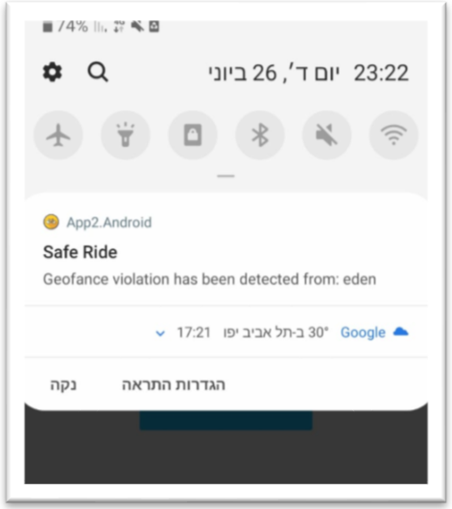


Hardware

- Device GPS service

Cloud

- SQL DB
- API
- Notification



74%

23:22 יום ד', 26 ביוני

App2.Android

Safe Ride

Geofance violation has been detected from: eden

Google 30° ב-תל אביב יפו 17:21

הגדרות התראה נקה

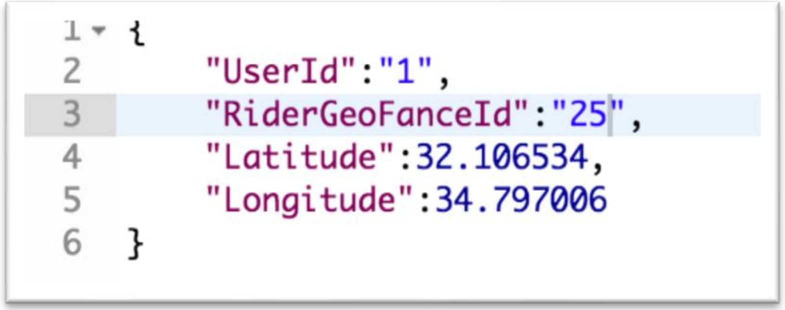
Algorithm

GEOFANCE API

- StartRide/SaveRiderPlot/FinishRide/SaveRiderGeoFancePlots
- Uses
  - Google maps API
- Logics:
  - Geofence – geometry of space
  - Notifications
  - Rides summary – distance, average speed, etc

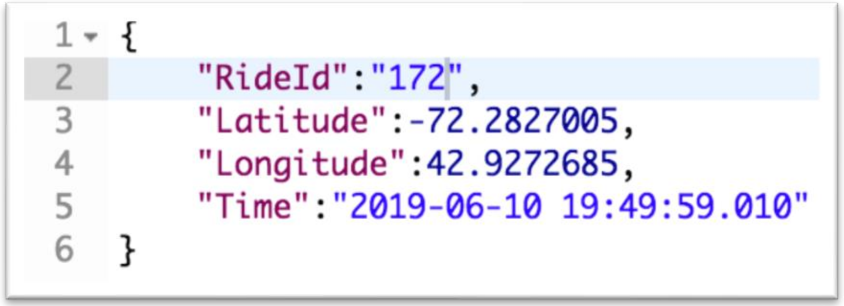
GEOFANCE API - JSONS

StartRide



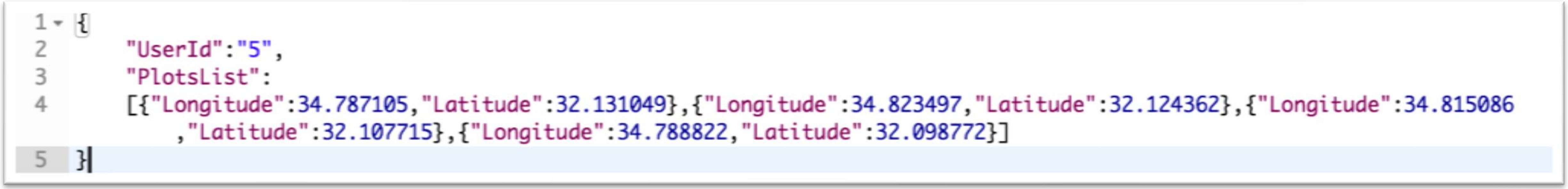
```
{
    "UserId":"1",
    "RiderGeoFanceId":"25",
    "Latitude":32.106534,
    "Longitude":34.797006
}
```

SaveRiderPlot/FinishRide



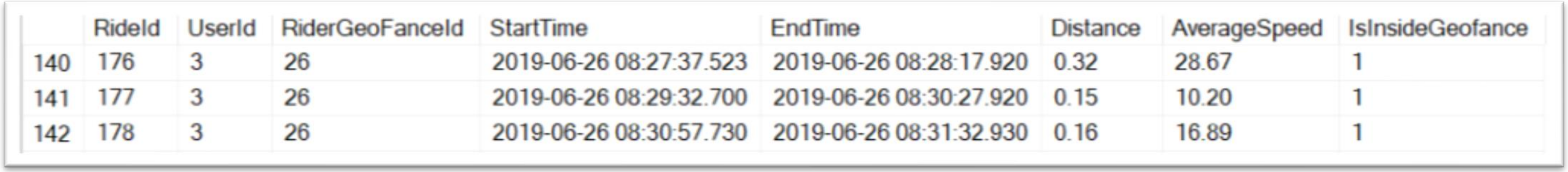
```
{
    "RideId":"172",
    "Latitude":-72.2827005,
    "Longitude":42.9272685,
    "Time":"2019-06-10 19:49:59.010"
}
```

SaveRiderGeoFancePlots

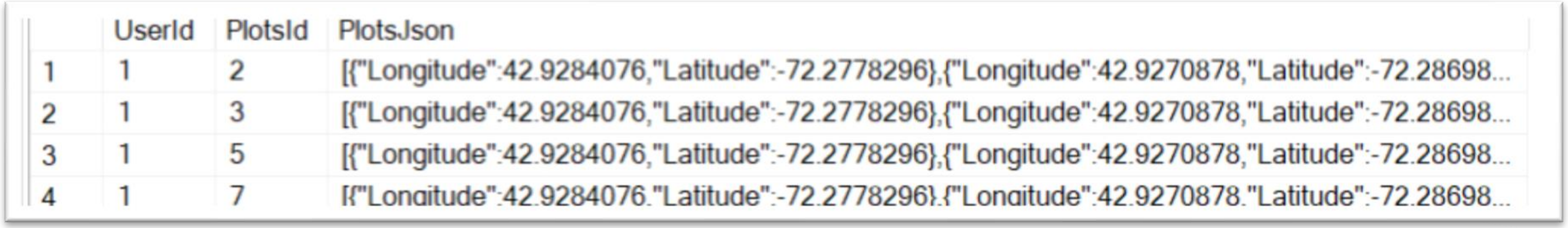


```
{
    "UserId":"5",
    "PlotsList":
    [{"Longitude":34.787105,"Latitude":32.131049},{"Longitude":34.823497,"Latitude":32.124362},{"Longitude":34.815086,"Latitude":32.107715},{"Longitude":34.788822,"Latitude":32.098772}]
}
```

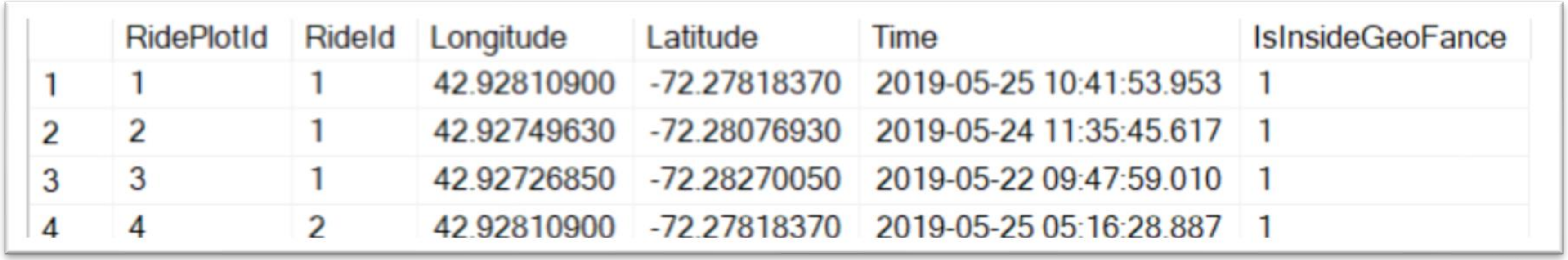
GEOFANCE API - TABLES



	RideId	UserId	RiderGeoFanceId	StartTime	EndTime	Distance	AverageSpeed	IsInsideGeofance
140	176	3	26	2019-06-26 08:27:37.523	2019-06-26 08:28:17.920	0.32	28.67	1
141	177	3	26	2019-06-26 08:29:32.700	2019-06-26 08:30:27.920	0.15	10.20	1
142	178	3	26	2019-06-26 08:30:57.730	2019-06-26 08:31:32.930	0.16	16.89	1



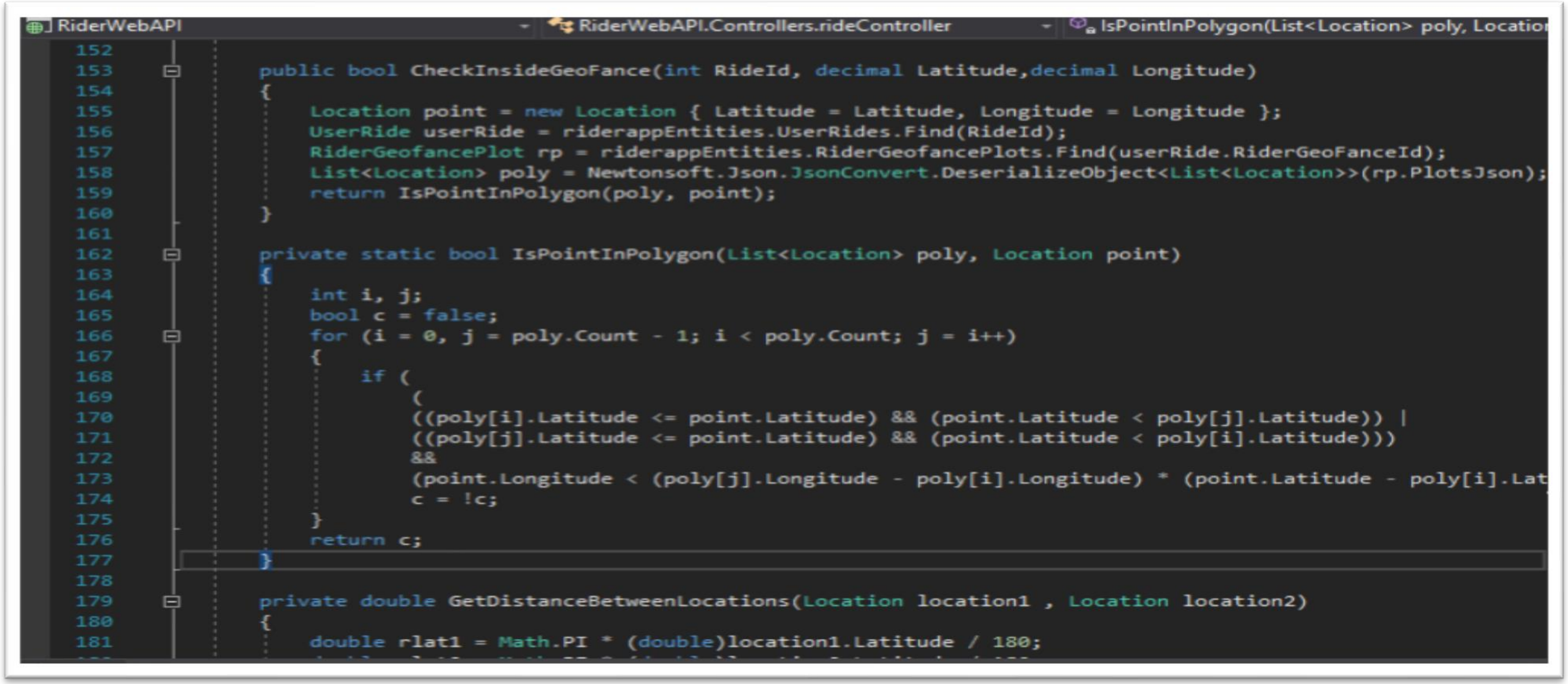
	UserId	PlotsId	PlotsJson
1	1	2	[{"Longitude":42.9284076,"Latitude":-72.2778296},{"Longitude":42.9270878,"Latitude":-72.28698...
2	1	3	[{"Longitude":42.9284076,"Latitude":-72.2778296},{"Longitude":42.9270878,"Latitude":-72.28698...
3	1	5	[{"Longitude":42.9284076,"Latitude":-72.2778296},{"Longitude":42.9270878,"Latitude":-72.28698...
4	1	7	[{"Longitude":42.9284076,"Latitude":-72.2778296},{"Longitude":42.9270878,"Latitude":-72.28698...



	RidePlotId	RideId	Longitude	Latitude	Time	IsInsideGeoFance
1	1	1	42.92810900	-72.27818370	2019-05-25 10:41:53.953	1
2	2	1	42.92749630	-72.28076930	2019-05-24 11:35:45.617	1
3	3	1	42.92726850	-72.28270050	2019-05-22 09:47:59.010	1
4	4	2	42.92810900	-72.27818370	2019-05-25 05:16:28.887	1

GEOFANCE API – ALGORITHM

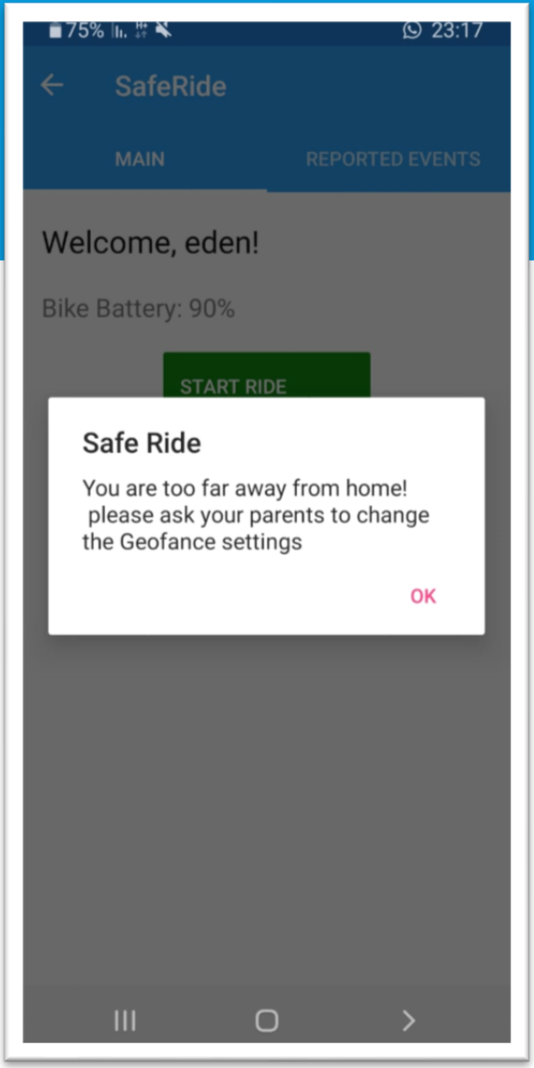
- Is point inside polygon?


RiderWebAPI
RiderWebAPI.Controllers.rideController
IsPointInPolygon(List<Location> poly, Locatio

```
    public bool CheckInsideGeoFance(int RideId, decimal Latitude,decimal Longitude)
    {
        Location point = new Location { Latitude = Latitude, Longitude = Longitude };
        UserRide userRide = riderappEntities.UserRides.Find(RideId);
        RiderGeofancePlot rp = riderappEntities.RiderGeofancePlots.Find(userRide.RiderGeoFanceId);
        List<Location> poly = Newtonsoft.Json.JsonConvert.DeserializeObject<List<Location>>(rp.PlotsJson);
        return IsPointInPolygon(poly, point);
    }

    private static bool IsPointInPolygon(List<Location> poly, Location point)
    {
        int i, j;
        bool c = false;
        for (i = 0, j = poly.Count - 1; i < poly.Count; j = i++)
        {
            if (
                (
                ((poly[i].Latitude <= point.Latitude) && (point.Latitude < poly[j].Latitude)) |
                ((poly[j].Latitude <= point.Latitude) && (point.Latitude < poly[i].Latitude)))
                &&
                (point.Longitude < (poly[j].Longitude - poly[i].Longitude) * (point.Latitude - poly[i].Lat
                c = !c;
        }
        return c;
    }

    private double GetDistanceBetweenLocations(Location location1 , Location location2)
    {
        double rlat1 = Math.PI * (double)location1.Latitude / 180;
```



Welcome, eden!

Bike Battery: 90%

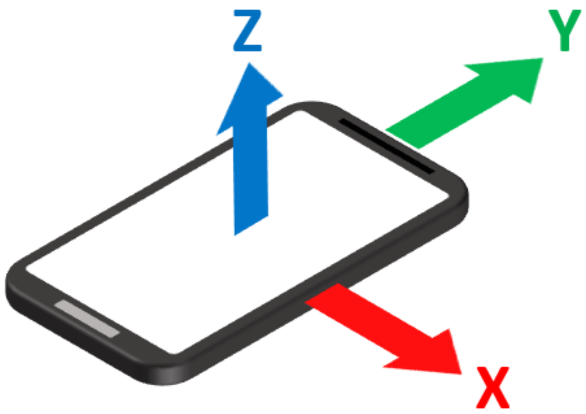
Safe Ride

You are too far away from home!
please ask your parents to change
the Geofance settings

FALL DETECTION

Safety

- Alert device users



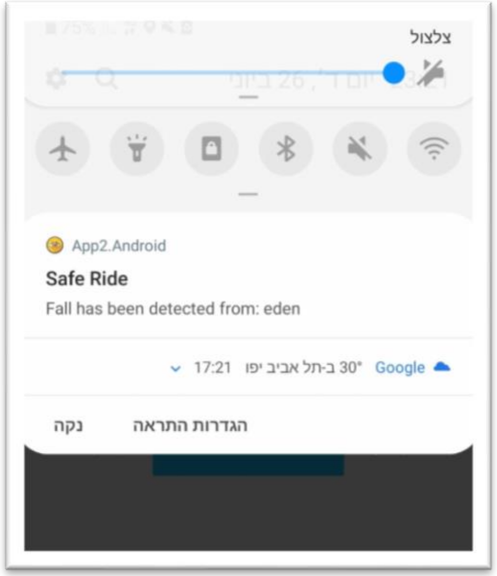
Hardware

- Phone 3- axis accelerometer

Cloud

- SQL DB
- API
- Notification

Algorithm


צלצול

App2.Android

Safe Ride

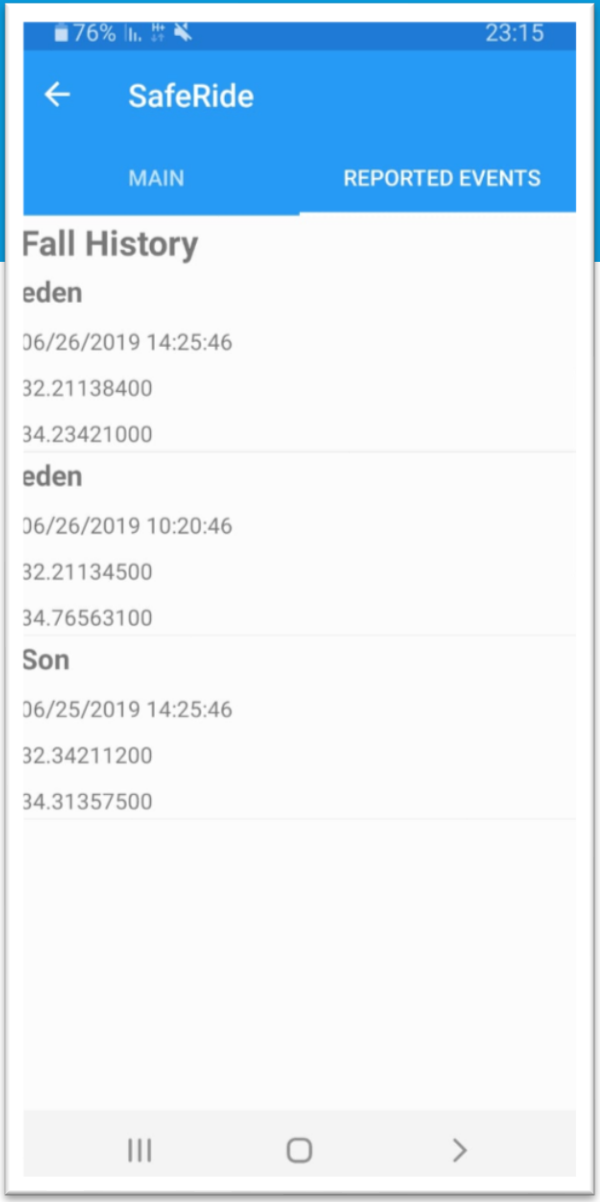
Fall has been detected from: eden

Google 30° ב-תל אביב יפו 17:21

הגדרות התראה נקה

FALL DETECTION API

- Analyze Accelerometer data
- Calibration
- Handles SQL database
- Users History



Fall History

eden

06/26/2019 14:25:46

32.21138400

34.23421000

eden

06/26/2019 10:20:46

32.21134500

34.76563100

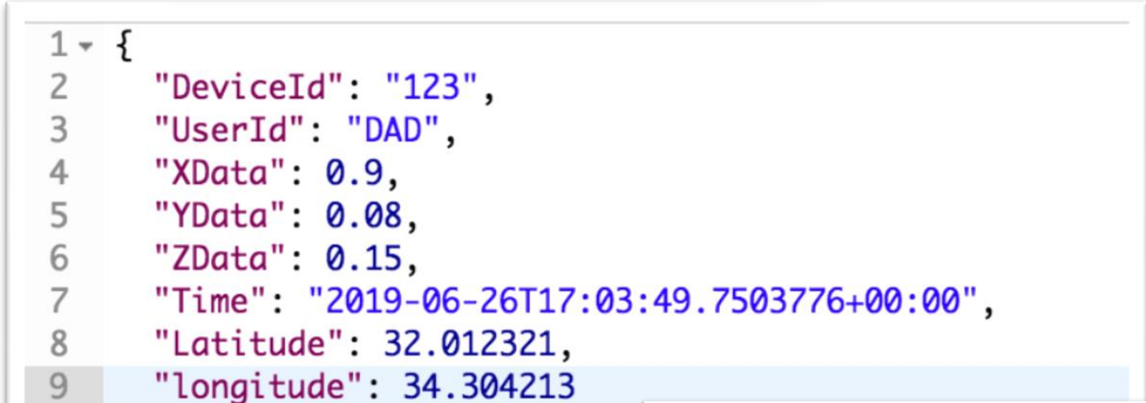
Son

06/25/2019 14:25:46

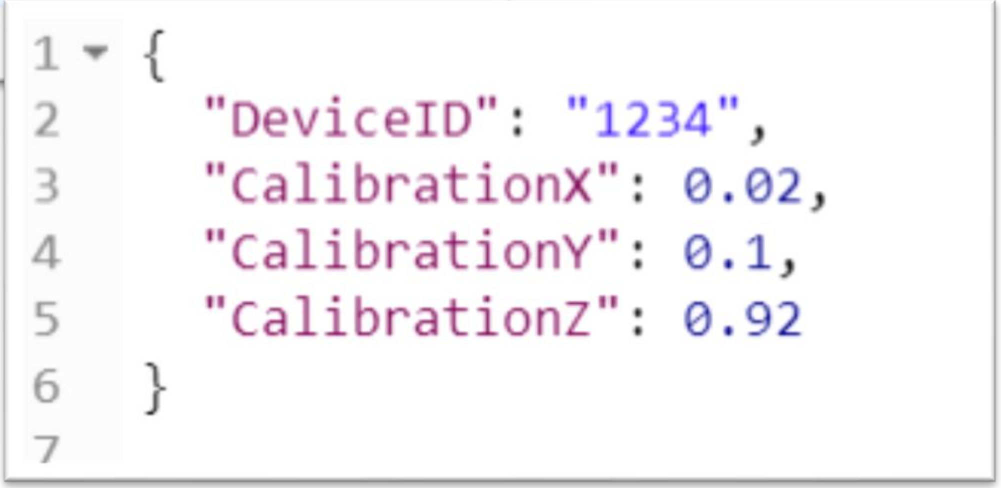
32.34211200

34.31357500

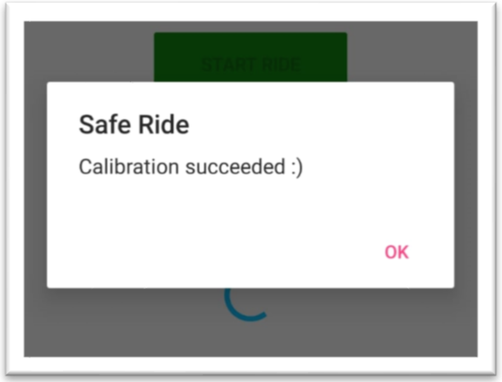
FALL DETECTION API - JSONS



```json
{
    "DeviceId": "123",
    "UserId": "DAD",
    "XData": 0.9,
    "YData": 0.08,
    "ZData": 0.15,
    "Time": "2019-06-26T17:03:49.7503776+00:00",
    "Latitude": 32.012321,
    "longitude": 34.304213
}
```

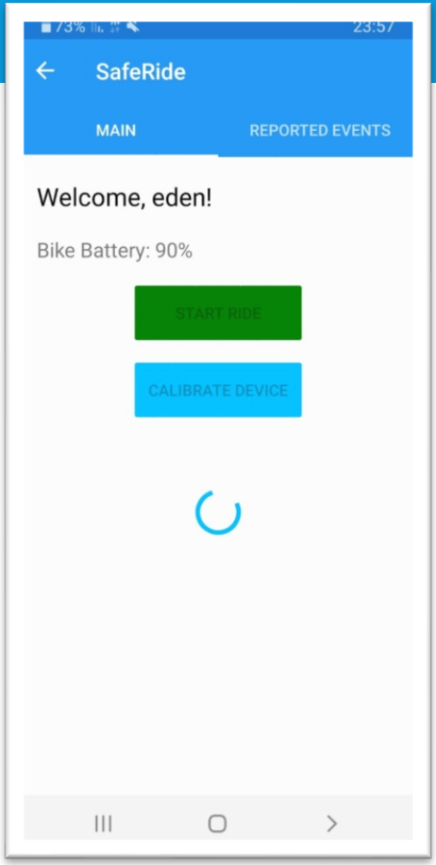


```json
{
    "DeviceID": "1234",
    "CalibrationX": 0.02,
    "CalibrationY": 0.1,
    "CalibrationZ": 0.92
}
```



Safe Ride

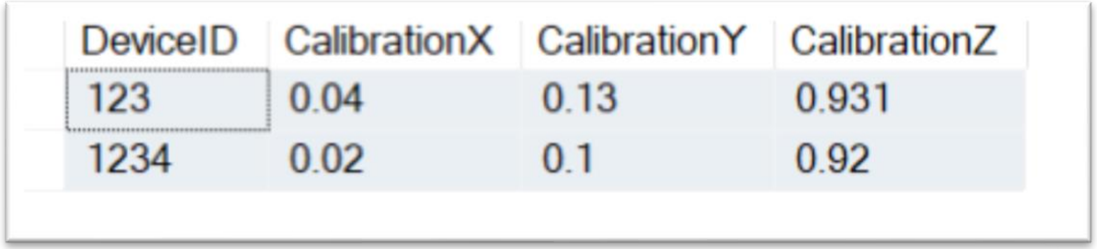
Calibration succeeded :)



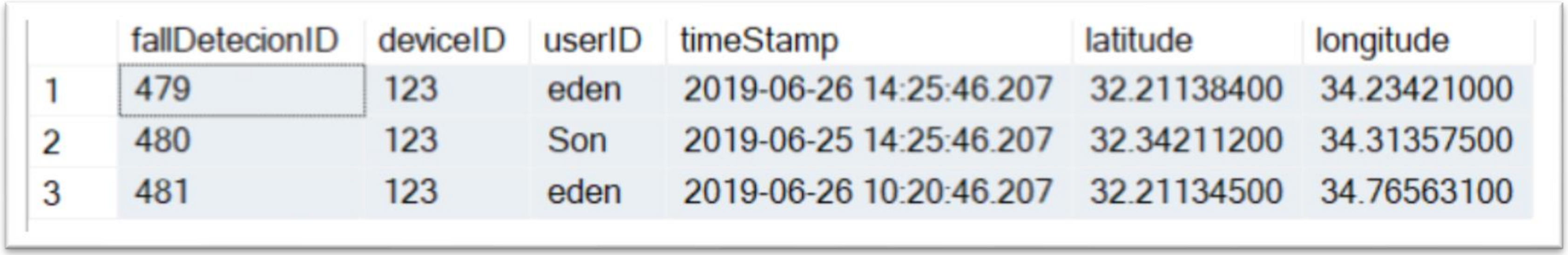
Welcome, eden!

Bike Battery: 90%

FALL DETECTION API - TABLES



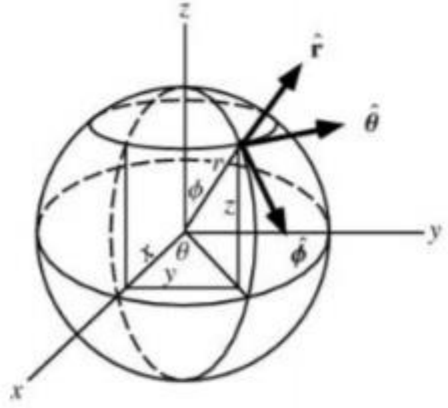
DeviceID	CalibrationX	CalibrationY	CalibrationZ
123	0.04	0.13	0.931
1234	0.02	0.1	0.92

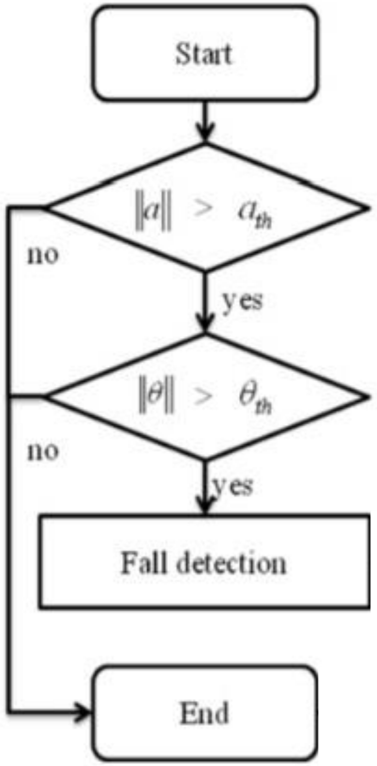


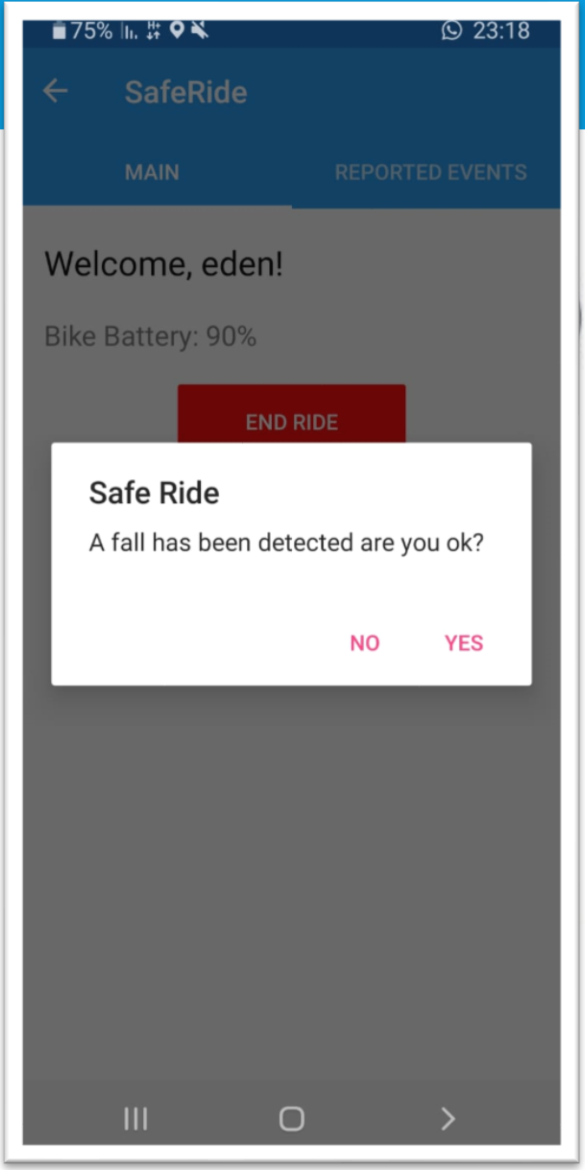
	fallDetecionID	deviceID	userID	timeStamp	latitude	longitude
1	479	123	eden	2019-06-26 14:25:46.207	32.21138400	34.23421000
2	480	123	Son	2019-06-25 14:25:46.207	32.34211200	34.31357500
3	481	123	eden	2019-06-26 10:20:46.207	32.21134500	34.76563100

FALL DETECTION API - ALGORITHM

- Fall detecting algorithm







Welcome, eden!

Bike Battery: 90%

Safe Ride

A fall has been detected are you ok?

THOUGHTS AHEAD

- **Gyroscope – more accurate fall detection**
- **SMS service**
- **Emergency service**
- **Multithreaded – allows falls detections and**
- **Performance**