



PedalaMi Acceptance Test Report

Version 1.0

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Revision History

Date	Version	Description	Author
2022-01-14	1.0	First Version	Lorenzo Italiano, Emanuele Triuzzi

1. Introduction	3
1.1 Purpose of this document	3
1.2 Document organization	3
1.3 Intended Audience	3
1.4 Scope	3
1.5 Definitions and acronyms	3
1.5.1 Definitions	3
1.5.2 Acronyms and abbreviations	4
2. Background	5
3. Overview of the Testing Process	5
3.1 Requirements	5
4. Unit Testing	7
4.1 Description	7
4.2 Framework	7
4.3 Tests	7
4.4 Results	8
5. Integration Testing	9
5.1 MongoDB Service Test	9
5.2 External APIs Service Test	10
6. User Acceptance Testing	11
6.1 Outcome	11
6.2 Registration and Login	11
6.3 Navigation Bar	12
6.4 Rides	13
6.5 Profile Page	14
6.6 Teams	16
6.7 Events	20
6.8 Badges and Rewards	23
6.9 Web Application	24
6.10 Add-ons	25

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

1. Introduction

1.1 Purpose of this document

The purpose of this document is to show the results of the testing process of the PedalaMi system. The goal is to verify that the system works correctly and satisfies the requirements presented in the Requirements Document.

1.2 Document organization

The document is organized as follows:

- Section 1, *Introduction*, describes the contents of this document.
- Section 2, *Background*, briefly introduces the background of the PedalaMi project.
- Section 3, *Overview of the Testing Process*, gives an overview of the testing process, listing all of the tests performed.
- Section 4, *Unit Testing*, describes the unit testing approach and how it is used to test the back-end of the application, showing the units tested and the overall coverage.
- Section 5, *Integration Testing*, reports the integration testing approach enumerating all the functionalities tested.
- Section 6, *User Acceptance Testing*, provides a series of test cases performed during the acceptance testing process and their relative results.

1.3 Intended Audience

- **Team members**, the development team should have full access to this document, which provides the results of the tests made.
- **Supervisors**, the professors from POLIMI and MDH should have full access to this document, to evaluate and grade the progress of the development as well as to advise the team where it is needed.
- **Customer**, Deloitte Digital, to whom deliver the final product. They should have full access to this document, to verify if the system meets all the requirements and works correctly.

1.4 Scope

This document addresses the results of the acceptance tests. Guidelines are provided for each test on how to replicate all the tests made. It also addresses unit testing, integration testing and their relative results.

1.5 Definitions and acronyms

1.5.1 Definitions

Keyword	Definitions
User	Synonym for <i>registered user</i>
User profile	A term indicating all the information linked to a user, such as its username, its profile picture, its teams and its personal statistics

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Navigation Bar	The bottom part of the application, it is used to move between the main pages
Nav Bar	Abbreviation of Navigation Bar
Tab	A button at the Navigation Bar
Ride data	A term indicating the group of the path, time, burned calories and average speed of a ride.
Record a ride	Collect ride data
Public event	Synonym for both <i>individual public events</i> and <i>team public events</i>
Private event	Synonym for <i>team private events</i>
Badge	Special rewards that can be unlocked after completing specific tasks. They are visible in the user profile.
Reward	The prize that the user can claim in exchange for earned points
Individual public event	An event proposed by the system admins that every user can join individually.
Exclusive Event	An individual public event whose access is limited to users who hold one or more specific badges, selected by the event creator.
Team private event	An event proposed by team admins, in which only two teams can participate.
Team public event	An event created by system admins that could have been proposed by team admins, in which an unlimited number of teams can participate.
User Acceptance Testing	Type of software testing, where an end-user follows predetermined procedures using the product.
Unit Testing	Type of software testing, where individual units or components are tested.
Integration Testing	Type of software testing, where multiple modules are tested as a group

1.5.2 Acronyms and abbreviations

Acronym or abbreviation	Definitions
NTR	Nothing to Report. There is no information on a specific topic available or necessary.
POLIMI	Politecnico di Milano
MDH	Mälardalen University
UI	User Interface
UX	User Experience
App	Mobile Application
Web App	Web Application
UAT	User Acceptance Testing
NPM	Node Package Manager
HTTP	HyperText Transfer Protocol

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

1.6 References

- *Requirements definition document:*
https://www.fer.unizg.hr/download/repository/Requirements_Definition_Document%5B8%5D.pdf
- *Acceptance Test Planning document:*
https://www.fer.unizg.hr/download/repository/Acceptance_Test_Planning.pdf
- *Jest:* <https://jestjs.io/>
- *SuperTest:* <https://www.npmjs.com/package/supertest>

2. Background

PedalaMi is an application created for promoting the use of bikes as a green and sustainable alternative to the non-zero-emissions means of transport, which still represent the most preferred choice in the field.

The main idea on PedalaMi is based on nudge and gamification techniques: a user, riding a bike, can obtain badges and collect points to redeem rewards. Users can also join teams and participate in private or public events.

This objective is pursued by providing the users with an interactive system that encourages cycling thanks to a fun reward and challenge-based system, in order to enrich the standard riding experience to make it more appealing to casual bikers.

The customers of the project are Guido Perrucchini and Giuseppe Valerio Gramazio, who are employed by **Deloitte Digital** as developer consultants.

The project team consists of six people from POLIMI and three from MDH, with one supervisor from each university.

3. Overview of the Testing Process

The *User Acceptance Testing* (UAT) is performed by an end-user and the test cases described in this document are created to test the overall user interaction with both the application and the web application.

These tests are designed to cover all of the functional requirements of the application with a focus on the front-end.

The back-end is tested using the Unit Testing approach, while the Integration Testing is performed using flutter.

The planning of the tests was made in accordance with the customer. In order to stick with his advice, a shared document with the customer was created and it includes all the tests.

The first UAT was performed internally, while the official one was made in a meeting with the customer.

All the UATs performed have the following specifications:

- **User:** Ita97
- **OS:** Android v10
- **Browser:** Google Chrome v97.0.4

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

3.1 Requirements

ID	Description
R1	The application must have authentication, allowing users to login
R2	The application must allow users to record bike rides
R2.1	The app should allow users to see the rides made
R3	The application must collect users' rides statistics and display them
R4	The application must allow users to edit their username and their profile picture
R5	The application must allow users to earn points from their rides
R6	The application must assign badges to the users as a reward, based on their rides' statistics
R7	The app should allow users to redeem rewards
R8	The app should display information about the redeemed rewards
R9	The application should allow users to create teams and become team admins
R10	The application should allow users to search for teams and join them
R11	The application should allow team admins to create private team events
R12	The application should allow system admins to create individual public events
R13	The application should allow system admins to create exclusive public events
R14	The application should allow a Team admin to propose a public team event that has to be approved by a System Admin
R15	The application should display to the user the available events and allow users to enrol for them
R16	The application could allow team admins to invite other teams to join private events
R17	The app should give extra points to the user if they bike in adverse weather
R18	The web app should display all the user's statistics

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

R19	The app could alert the user when the air quality is low
------------	--

4. Unit Testing

4.1 Description

Unit Testing is a type of software testing, where single units are tested independently. The purpose is to validate that each unit of the software performs as expected.

As it is intended to be, unit testing was made during the development phase. When a new feature is developed, it is tested and only after that, the new feature is integrated into the application. This procedure was made according to the standards using Github. Each feature is developed in a new branch, when the development and the testing part is done the branch is merged into the master branch.

4.2 Framework

The back-end is developed using Node.js and the team decides to perform the unit tests of the components exploiting a new, very popular testing framework called **Jest** (v. 27.4), which is well integrated with Node.js. Jest can safely run tests in parallel thanks to the unique global state, and it makes the testing very quick. Moreover, this framework can easily collect the code coverage of the tests from the entire project without additional setup.

Moreover, in order to provide a high-level abstraction for the HTTP testing, **SuperTest** is used, which is an npm module that makes easy assertions via *superagent*, a client-side HTTP request library.

4.3 Tests

The components tested are the following:

- EventHandler.js
- GamificationController.js
- ProfileController.js
- RewardController.js
- RideHandler.js
- TeamController.js

The tests are performed using the following command:

```
npm test
```

Which is a shortcut for:

```
npx jest --coverage
```

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

4.4 Results

File	% Stmts	% Branch	% Funcs	% Lines
All files	85.4	82.2	77.04	85.98
pedalami	71.05	75	0	71.05
server.js	71.05	75	0	71.05
pedalami/backend	100	100	100	100
schemas.js	100	100	100	100
pedalami/backend/components	85.77	82.28	80.34	86.4
eventHandler.js	79.47	83.58	66.12	80.53
gamificationController.js	98.76	74.66	100	98.76
profileController.js	89.74	80	100	89.74
rewardController.js	90.19	90	85.71	90.19
rideHandler.js	91.3	80	85.71	91.3
teamController.js	93.47	88.23	100	93.47
Test Suites: 6 passed, 6 total Tests: 115 passed, 115 total Snapshots: 0 total Time: 17.663 s Ran all test suites.				

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

5. Integration Testing

Integration testing is the phase in which multiple software modules are tested as a group. It was conducted to assess the compliance of a component and its interactions with other components, in order to comply with the functional requirements.

The integration testing was carried out following the unit test of the individual components. In particular, the integration test carried out mainly focuses on the interactions of the client with the Heroku server, and the interaction of the server with the MongoDB database.

Furthermore, the interaction with external APIs was tested, such as the service to obtain the air quality given the coordinates.

5.1 MongoDB Service Test

This group of tests was crucial for development: since the team is divided into frontend and backend, the backend simply implemented and tested the main functionalities in dart, in such a way to provide practical examples for the frontend team.

In this way, it was much easier to identify errors and fix them. In particular, when changes have been made on the backend, the integration test on the frontend were done, and if errors occurred, they would have been immediately fixed by the backend team. E.g., if some fields were changed on some schema of the database, the tests would fail if the corresponding changes weren't made also on the model in the dart files.

In particular, the integration testing was performed using the `test` function of the `flutter_test` package, and consisted of:

- **Record a ride testing:** if the ride is saved successfully, then the server should not return `null`
- **Ride history testing:** if, given a user, the rides are retrieved correctly, the server should not return `null` as a response
- **Redeem a reward testing:** if the reward is redeemed correctly, then the server should not return `null`
- **Get user's rewards testing:** if, given a user, the rewards are retrieved correctly, then the server should not return `null`, instead, it must return a list of rewards of the user
- **Get team testing:** given an existing `teamId`, the server should return a team and the client must be able to access the list of its users
- **Private team event creation testing:** given a hosting team id, an invited team id, and the admin id of the first one, if the private event can be created, the server should return an Event which is not `null`
- **Private team event invite acceptance testing:** given an event id, a team id, and its admin id, if the acceptance of the invite to the private event is performed, the server should return `true`
- **Private team event invite rejection testing:** given an event id, a team id, and its admin id, if the rejection of the invite to the private event is performed, the server should return `true`
- **Public team event creation testing:** given a hosting team id and its admin id, if the

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

public event is created, the server should return an Event which is not null

- **Public team event enrolling testing:** given an event id, a team id, and its admin id, if the enrolling to the public event is performed, the server should return true
- **Event search testing:** given a team id, its admin id, and a title, the server should return a list of events (that are not already enrolled) that contains the title specified and should not return null
- **Event join testing:** given a team id, the id of one of its members, and an event id, the server should return true if the user can join the event
- **Get joinable event testing:** given a user id, the server should return a list of events the user can join, and not null

5.2 External APIs Service Test

In order to test external services APIs, proper testing has been made. Since it is an external service, unit testing cannot be performed by us.

The tests were done still in a dart file, since the usage of this API is done only from the client, and is not intercepted by the server.

In particular, the tests were made using the test function of the flutter_test package. The tests consisted of:

- **Air Quality testing:** the air quality index is retrieved given some coordinates. The result of this test is an integer corresponding to the air quality index, and should not be -1, a non-valid index.
- **Weather testing:** a weather id (a code representing the current weather) is retrieved given some coordinates. The result of this test is an integer corresponding to the weather id, and should not be -1, a non-valid index.

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

6. User Acceptance Testing

6.1 Outcome

- *OK*: test passed;
- *OK+NOTE*: test passed, but with a note (the note is reported in a new row);
- *KO*: test failed;
- *DELETED*: the feature was deleted and cannot be tested;
- *NEED DEPLOY*: the feature is still a work in progress, and cannot be tested at this moment.

6.2 Registration and Login

Test Name	Registration
ID	T1
Requirements	R1
Prerequisites	User has an active Google account User installs and opens the app
PASS / FAIL criteria	User joins with their Google account
Procedure	<ul style="list-style-type: none"> - User installs the application - User clicks on Google authentication button - User enters a username - User allows/declines access to location
Outcome	OK

Test Name	User Login
ID	T2
Requirements	R1
Prerequisites	User has registered in the app
PASS / FAIL criteria	User gets redirected to Map Tab
Procedure	<ul style="list-style-type: none"> - User starts the application - User clicks on Google authentication button - User allows/declines access to location
Outcome	OK

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

6.3 Navigation Bar

Test Name	Navigate to Profile Tab
ID	T3
Requirements	R3
Prerequisites	User has logged in
PASS/FAIL criteria definition	System shows information about Points, Rewards, Statistics, Earned Badges and Ride History
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar
Outcome	OK

Test Name	Navigate to Map Tab
ID	T4
Requirements	R2
Prerequisites	User has logged in
PASS/FAIL criteria definition	System shows user's location on map
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar
Outcome	OK

Test Name	Navigate to Teams Tab
ID	T5
Requirements	R9
Prerequisites	User has logged in
PASS/FAIL criteria definition	System allows user to search for Teams and shows information about Joined Teams
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar
Outcome	OK

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Test Name	Navigate to Rewards Tab
ID	T6
Requirements	R8
Prerequisites	User has logged in
PASS/FAIL criteria definition	System shows list of all available Rewards to be redeemed
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Rewards Tab on Navigation Bar
Outcome	OK

Test Name	Navigate to Events Tab
ID	T7
Requirements	R15
Prerequisites	User has logged in
PASS/FAIL criteria definition	System shows Joined Events and allow user to search for Available Events
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar
Outcome	OK

6.4 Rides

Test Name	Start a Ride
ID	T8
Requirements	R2
Prerequisites	User is in Map Tab
PASS/FAIL criteria definition	The Stop button is displayed on the screen
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar - User clicks on "Start Ride" button
Outcome	OK

Test Name	Perform a Ride
ID	T9
Requirements	R2
Prerequisites	User has started a ride
PASS/FAIL criteria definition	The location on the map changes
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

	<ul style="list-style-type: none"> - User click on "Start Ride" button - User performs the ride
Outcome	OK

Test Name	Stop a Ride
ID	T10
Requirements	R5
Prerequisites	User is performing a ride, User is in the Map Tab
PASS/FAIL criteria definition	System shows information related to the ride and points earned
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar - User click "Start Ride" button - User starts biking - User clicks "Stop Ride" button
Outcome	OK

Test Name	View the Previous Rides
ID	T11
Requirements	R2.1
Prerequisites	User is in the Profile Tab, User has performed at least one ride
PASS/FAIL criteria definition	System shows path of the ride on map
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User scrolls vertically the white part of the screen - User reaches Previous Rides section - User clicks on one of the rides
Outcome	OK

6.5 Profile Page

Test Name	Scroll Profile Page vertically
ID	T12
Requirements	R3
Prerequisites	User is in the Profile Tab
PASS/FAIL criteria definition	System shows Profile Page scrolled to

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User scrolls vertically the white part of the screen
Outcome	OK

Test Name	Scroll Badges list horizontally
ID	T13
Requirements	R6
Prerequisites	User is in the Profile Tab, User has earned several badges
PASS/FAIL criteria definition	<i>IF</i> badge list does not fit in his section <i>THEN</i> System shows Earned Badges not visible before
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User scrolls vertically the white part of the screen - User reaches Earned Badges section - User scrolls this section horizontally to view the badges
Outcome	OK + NOTE
Note	Tested after T33 and completed OK

Test Name	Edit Profile
ID	T14
Requirements	R4
Prerequisites	User is in the Profile Tab
PASS/FAIL criteria definition	User is redirected to profile editing view
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User clicks on "Edit Profile" button
Outcome	OK

Test Name	Change username with an NOT existing one
ID	T15.a
Requirements	R4
Prerequisites	User is editing the Profile
PASS/FAIL criteria definition	The System displays the new Username typed

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User clicks on "Edit Profile" button - User enters a new username - User clicks on "Change Username" button
Outcome	OK

Test Name	Change username with an existing one
ID	T15.b
Requirements	R4
Prerequisites	User is editing the Profile
PASS/FAIL criteria definition	The System displays the new Username typed
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User clicks on "Edit Profile" button - User enters an existing username - User clicks on "Change Username" button
Outcome	OK

Test Name	Change profile picture
ID	T16
Requirements	R4
Prerequisites	User is editing the Profile
PASS/FAIL criteria definition	The System displays the new picture selected
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Profile Tab on Navigation Bar - User clicks on "Edit Profile" button - User clicks on "Change Profile Picture" button - User chooses a picture from gallery or takes a new one
Outcome	OK

6.6 Teams

Test Name	Create Team
ID	T17
Requirements	R9
Prerequisites	User is in the Team Tab
PASS/FAIL	User notified with Team created successfully and the new team is

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

criteria definition	visible in the list
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User clicks on a Button with Plus icon - User enters a Team name - User enters short Team description - User clicks on Create team button
Outcome	OK

Test Name	Search Existing Team
ID	T18.a
Requirements	R10
Prerequisites	User is in the Teams Tab
PASS/FAIL criteria definition	System shows User's search results
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User clicks on input field, keyboard shows up - User types in possible team name - User clicks on search icon on keyboard
Outcome	OK

Test Name	Search Not ExistingTeam
ID	T18.b
Requirements	R10
Prerequisites	User is in the Teams Tab
PASS/FAIL criteria definition	User is notified with "no teams found"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User clicks on input field, keyboard shows up - User types a not existing team name - User clicks on search icon on keyboard
Outcome	OK

Test Name	Scroll Team Search
ID	T19
Requirements	R10
Prerequisites	User is in the Teams Tab
PASS/FAIL criteria definition	IF the list of teams does not fit the current page THEN teams not visible before should now be visible

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User search for a team
Outcome	OK

Test Name	Join Team Success
ID	T20
Requirements	R10
Prerequisites	User has searched a Team
PASS/FAIL criteria definition	User notified with team joined successfully and the team appears in "My Teams" list
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User search for a team - User click join team
Outcome	OK

Test Name	Team Admin cannot Leave the Team
ID	T21.a
Requirements	R10
Prerequisites	User is Team Admin
PASS/FAIL criteria definition	User is notified with "Admin cannot leave his team"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User click on a Team Created - User click leave team
Outcome	OK

Test Name	Leave Joined Team
ID	T21.b
Requirements	R10
Prerequisites	User has joined a Team
PASS/FAIL criteria definition	User notified with "team left successfully"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User click on a Team Joined - User click leave team
Outcome	OK

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Test Name	View Joined Teams
ID	T22
Requirements	R10
Prerequisites	User has joined at least one team
PASS/FAIL criteria definition	System shows list of Teams joined
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User navigate to joined teams
Outcome	OK

Test Name	Scroll Joined Teams
ID	T23
Requirements	R10
Prerequisites	User has joined several teams
PASS/FAIL criteria definition	System shows Joined Teams not visible before
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User navigates to My Teams - User scrolls teams
Outcome	OK

Test Name	View Team Information
ID	T24
Requirements	R10
Prerequisites	User is in the Team Tab, User has joined at least one Team
PASS/FAIL criteria definition	System shows information about Team
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User finds team who which to learn about - User clicks on the Team - User reads data about Team

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Outcome	OK
----------------	----

Test Name	View Team Member List
ID	T25
Requirements	R10
Prerequisites	User is inside a Team he joined
PASS/FAIL criteria definition	System shows information about Member list
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Teams Tab on Navigation Bar - User finds team who which to learn about - User clicks on the Team - User clicks on "Team Members" - User looks through members of Team
Outcome	OK

6.7 Events

Test Name	Search for an existing Event
ID	T26.a
Requirements	-
Prerequisites	User is in Events Tab
PASS/FAIL criteria definition	System shows a list of available events
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User looks through events list - User types a name for the event
Outcome	OK

Test Name	Search for a NOT existing Event
ID	T26.b
Requirements	-
Prerequisites	User is in Events Tab
PASS/FAIL criteria definition	System shows "No events found"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User looks through events list - User types a name for the event

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Outcome	OK
----------------	----

Test Name	Join Event
ID	T27
Requirements	R15
Prerequisites	User is in Events Information
PASS/FAIL criteria definition	System shows the event in the "Joined Events" list
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User search for events and looks through events list - User clicks on "Join" button in an event
Outcome	OK

Test Name	Leave Event
ID	T28
Requirements	-
Prerequisites	User is in Events Tab
PASS/FAIL criteria definition	User notified with "Event left successfully"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User looks through events list joined - User clicks on "Leave" button in an event
Outcome	DELETED
Note	Client confirm that is not required

Test Name	Create an Individual Public Event
ID	T29
Requirements	R12
Prerequisites	User has logged in, User is a System Admin
PASS/FAIL criteria definition	System shows "Public Event created successfully", A new Individual Public Event is displayed in the page
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User looks through events list - User clicks "Create Public Event" - User provides the information needed - User clicks "Create"

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Outcome	<i>OK + NOTE</i>
Note	<i>You need to be a PedalaMi system admin. Not performed from UI, just from the back-end.</i>

Test Name	Create an Exclusive Event
ID	T30
Requirements	R13
Prerequisites	User has logged in, User is a System Admin
PASS/FAIL criteria definition	System shows "Public Event created successfully", A new Exclusive Event is displayed in the page
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Events Tab on Navigation Bar - User looks through events list - User clicks "Create Public Event" - User check the "exclusive event" box - User select a badge - User provides the information needed - User clicks "Create"
Outcome	<i>DELETED</i>

Test Name	Create Private Team event
ID	T31
Requirements	R11, R16
Prerequisites	User has logged in, User is a Team Admin
PASS/FAIL criteria definition	System shows "Private Event created successfully"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Team Tab on Navigation Bar - User clicks on "Create Event" button - User fills out information - User selects "Private" event - User selects an adversarial team - User clicks on Confirm
Outcome	<i>OK</i>

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Test Name	Propose Public Team event
ID	T32
Requirements	R14
Prerequisites	User has logged in, User is a Team Admin
PASS/FAIL criteria definition	System shows "Public event sent correctly"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Team Tab on Navigation Bar - User clicks on "Create Event" button - User fills out information - User selects "Public" event - User clicks on Confirm
Outcome	OK

6.8 Badges and Rewards

Test Name	Achieve Badges
ID	T33
Requirements	R6
Prerequisites	User is performing a ride
PASS/FAIL criteria definition	System shows earned Badges
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar - User click "Start Ride" - User Starts biking - User clicks "Stop Ride" - User achieve a badge - User navigates to Profile Page
Outcome	OK

Test Name	Check Available Rewards to Redeem
ID	T34
Requirements	R8
Prerequisites	User is in the Reward Tab
PASS/FAIL criteria definition	System shows all available rewards to be redeemed
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Rewards Tab on Navigation Bar - User looks through available rewards to choose from

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

Outcome	OK
----------------	----

Test Name	Redeem Reward
ID	T35
Requirements	R7
Prerequisites	User is in the Reward Tab User has enough points to redeem a reward
PASS/FAIL criteria definition	System notifies the user with "You have redeemed your reward"
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Rewards Tab on Navigation Bar - User looks through available rewards to choose from - User clicks in the button near the reward and then GET - User clicks on YES in the confirmation pop-up
Outcome	OK

Test Name	Check Redeemed Rewards
ID	T36
Requirements	R8
Prerequisites	User is in the Reward Tab, User has redeemed at least one reward
PASS/FAIL criteria definition	System shows all redeemed rewards
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Rewards Tab on Navigation Bar - Clicks on "Redeemed Rewards" button
Outcome	OK

6.9 Web Application

Test Name	Display statistics on Web Dashboard
ID	T37
Requirements	R3, R18
Prerequisites	User is registered in the app
PASS/FAIL criteria definition	System should display the statistics of the user
Procedure	<ul style="list-style-type: none"> - User opens the web page
Outcome	OK

Test Name	Display the ride history on Web Dashboard
-----------	---

PedalaMi	Version: 1.0
Acceptance Test Report	Date: 2022-01-14

ID	T38
Requirements	R18
Prerequisites	User is registered in the app
PASS/FAIL criteria definition	System should display all the recorded rides
Procedure	- User opens the web page
Outcome	OK + NOTE
Note	Map plug-in doesn't support web app, yet.

6.10 Adds-on

Test Name	Air Quality Alert
ID	T39
Requirements	R19
Prerequisites	User has logged in
PASS/FAIL criteria definition	System shows a pop-up with the air quality of the current location
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar - User Starts a ride
Outcome	OK

Test Name	Adverse Weather
ID	T40
Requirements	R17
Prerequisites	User is performing a ride in adverse weather
PASS/FAIL criteria definition	The points earned should be higher
Procedure	<ul style="list-style-type: none"> - User opens the app - User clicks on Map Tab on Navigation Bar - User clicks "Start Ride" - User starts biking in adverse weather condition - User clicks "Stop Ride"
Outcome	OK + NOTE
Note	Could not be tested directly because of the weather.