

# ficha

C'est pas fichu !

## REPORT

## I. Context

FICHA is a project which has for purpose to revalorize the selective waste sorting for, in a first time, the students living in university residences. Through recycling and connected boxes and an application, the weight of recycled and sorted waste would allow FICHA's users to obtain rewards, for example: discounts in bakeries, groceries and other locals trading.

The people who initiates this project are students from the Business School of Grenoble (GEM) in sabbatical year for totally devoting to the realisation of FICHA : Vincent HIPAULT, Hubert MENARD and Guillaume FOURCHER. We had meetings with them each week to keep them updated on our advancement.

## II. Purpose

Our participation to this project has consisted in the implementation of a mobile application which will allow the user to connect to a recycling box. The application must display statistics about the sorted waste, the number of points of the user (computed in function of the weight of recycled waste) and also a list of the rewards that the user will ask for once its number of points is attained. The list of rewards may be displayed as a list or as an interactive map with the location of all the different partners stores.

## III. Implementation choices

### 1. Front-end

For the generation and the development of the mobile application, we have chosen the Ionic Framework, which is very complete using several languages. It uses for web implementation HTML, for appearance SCSS and for the functionalities Angular/Typescript.

A graphic chart was furnished by the Product Owners and we tried to respect it as much as possible for the Front-end. It's visible in appendix 1.

### 2. Back-end

For the back-end, we used JHipster to handle the server, the database and the user accounts (creation, connection, information). We never used JHipster before, so it took us considerable time to learn how to use it but we finally achieve our goals.

## IV. Architecture

### A. Front-End

The application pages are located to:



Ficha\_app/src/pages

Each page owns 3 or 4 files:

- [name].html : for the visible part HTML
- [name].scss : for the appearance (colors, fonts...)
- [name].ts : for the functions
- [name].module.ts : for the modules, imports/exports

The application is divided into 6 parts which are visible in appendix 2 (first draft: drawing), 3 (second draft: mockflow) and 4 (finale version: screens from the app).

## 1. Open page

The open page is the first visible page when the user downloads the application. It allows to create its account or to connect. When registering, an email is sent to the user's email address so the user can confirm its account and then use it.

## 2. Home page

Once the user is connected, the user comes to an home page which displays the number of points that the user own. There are also 4 buttons to access the different parts of the application.

## 3. User profile

The user profile is supposed to allow the user to add and modify its information. For example: its surname, name, email address and there's also a disconnect button.

## 4. Statistics page

The statistics page gets from the database the information obtained by recycling and sorted from the connected boxes. It displays the statistics as a graphic, indicating the proportion of each type of waste.

As the recovery of sorted quantities and the sending of these information to the database are made by the IESE students, we only implemented locally this page. The values "Plastique 20, Papier 25, Verre 10 etc" are locally written but should, when the IESE students prototypes will be functional, be retrieved from the distant database.

## 5. QR code page

The QR Code page displays a unique barcode related to the username of the user. This barcode is generated with an Ionic module called NgxQRCodeModule.

The aim of this barcode is to allow the user to connect to the recycling box.

## 6. Rewards page

The reward page is composed of 4 parts:

### B. Reward type

This page sorts the rewards by type: restaurants, shops, hobbies and others. They are clickable buttons to access the list of rewards of the clickable type.

### C. List of the type

All the rewards are clickable so they give access to more information about the reward itself.

### D. Page récompense

The reward page displays all the information about the reward: number of points to get it, button to ask for it, information, description...

### E. Interactive map

Accessible with the button "Ouvrir la carte", it's a map which displays all the partners shops with rewards.

## F. Back-end

We used JHipster for three main features :

### 1. User Accounts Management

Thanks to the JHipster app we managed the subscription of new users, the connection and the sending of emails. When a new user subscribe, an automatic email is sent to his email address so he can confirm his account. We can also validate this new user account through the JHipster app. We can see the interface of the user management in the 5th appendix. For example, the user with ID 4 needs to be activated to be able to use its account : by his email address or directly through this interface by clicking on "Desactivated".

### 2. User points Entity

We had to create a new entity to handle the user points which are used for asking rewards. At the subscription of a new user, a user point entity is automatically created and associated to this account (Appendix 6). At the beginning, the number of point is equal to zero. It will then increase when throwing recycled waste into the connected box. It will also decrease when asking for rewards.

### 3. Rewards Entity

The rewards need to be manually added through the JHipster app. There is a form that need to be fill in (Appendix 7), with for example the name of the merchant, the address, the phone number, the website, the description of the reward or also the number of points to get it.

## V. Prospects for improvement

### 1. Merchant profile

We have implemented a checkbox in the register page to allow the user to specify if its a merchant or a simple user.

In the case where the checkbox is checked, that would allow to display specific options for merchants such as the creation of rewards from the application and the update of information (opening hours, address, website...). In order to do that we successfully implemented different roles in JHipster to differentiate the client from the merchant. Then what needs to be done is to use these roles in the front-end to allow the merchant to access to the entity page in which they can add new entities like a new reward. However, we started to implement this features but we had some issues and thus, it is not in the final version because it wasn't working well.

### 2. User profile page

The user profile doesn't work because we didn't achieve to implement the saving of the data in the input. Indeed, we didn't find a way to allow users to modify userDTO's data located in the back-end from the front-end. Apparently, the only way to do that is to proceed the same way we did with points : adding an entity with the fields we need with a one-to-one relationship to the user entity.

### 3. Sending the reward

The functionality "Demander cette récompense" which is in the reward page is not implemented yet, even if it still decrease the amount of point the user have. Thereafter, the goal would be to send an email with the discount code automatically generated and usable on the website or any other type of offer usable in the shop directly. But the issue we faced was that we didn't manage to know how to retrieve the discount codes, because it should be the merchant that provide those one and then we could use them in the app by sending an email with a new template describe in JHipster.

### 4. Interactive map

The interactive map works correctly for local data (if we give manually the latitude and longitude of each merchant) but for some unknown reasons, we didn't achieve to access the latitude and longitude from the database.

The interactive map is shaded with the text “For development purposes only” because we use the Google Maps API which is now a paid service. As developers and for project, we decided to use only a free account and token. For further development and finale exportation, the developers will have to get a paid account to remove this annoying text. Another solution would be to re implement this part using Open Street Map.

## 5. Adding rewards directly from the app

It should be very useful to have the possibility to add new rewards directly from the app, without having to connect to the JHipster app. We added a checkbox in the subscription page to indicate whether the user is a client or a merchant, and in case the user is that last type of user he should be able to open a page dedicated to the creation of rewards : the entity page as we said earlier.

## 6. Application exportation

Once the application is finished, we will need to find a way to export the application so anyone can download and use it from any device connected to the internet.

We succeeded to export the application on an Android Device thanks to Ionic but without the JHipster/server part. As the management of the users account is made with our JHipster server, that we need to launch with our computers, the server is not accessible on mobile device. So we can't connect/register outside our computer.

To export the application for the final use, we will need to export the database and import it on the final online server. Unfortunately, we haven't find a way to do that with JHipster.

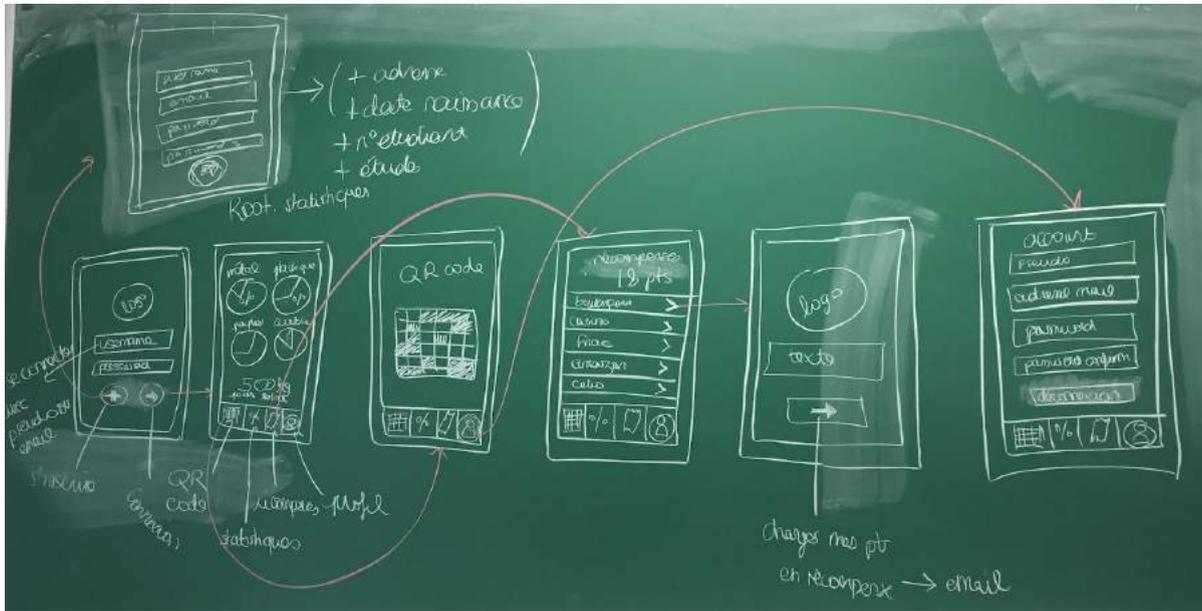
## VI. Appendix

**APPENDIX 1 : GRAPHIC CHART**

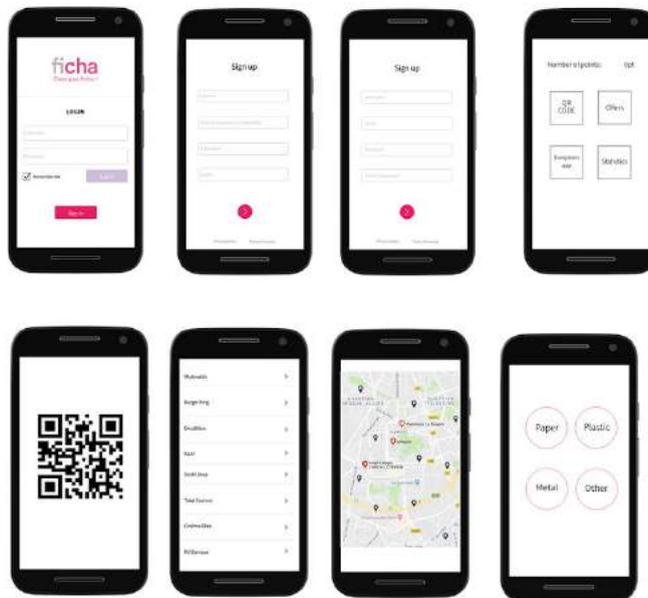
**COULEURS PRINCIPALES**

**COULEURS SECONDAIRES**

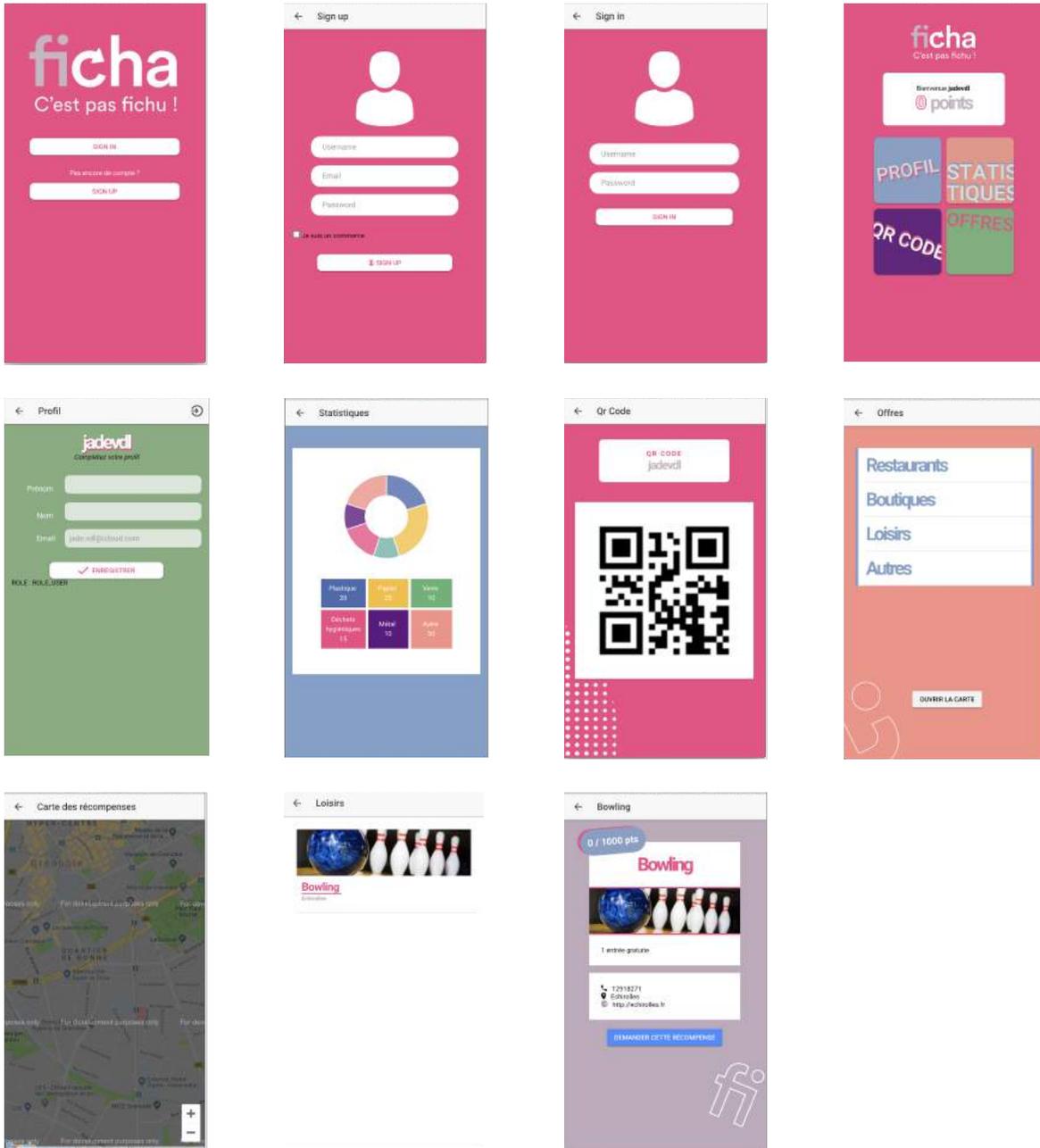

APPENDIX 2 : MOCKFLOW DRAWINGS



APPENDIX 3 : MOCKFLOW



APPENDIX 4 : FINALE VERSION, SCREENS FROM THE APPLICATION



**APPENDIX 5 : JHipster application - User management**

ID	Login	Email	Language	Profiles	Created date	Modified by	Modified date
1	system	system@localhost	en	ROLE_USER ROLE_ADMIN	system		
3	admin	admin@localhost	en	ROLE_USER ROLE_ADMIN	system		
4	user	user@localhost	en	ROLE_USER	admin	09/04/19 10:53	

**APPENDIX 6 : JHipster application - Point Entity**

ID	Point	User Extent
5	340	jadevcl

**APPENDIX 7 : JHipster application - Reward Entity**

ID	Type	Name	Address	Phone	Website	Image	Description	Points	Longitude	Latitude
1	LOISIRS	Bowling center	19 Avenue de Grugiasco, 38130 Échirolles	123456789	echirolles.bowlingcenter.fr	https://www.luztime-72.fr/wp-content/uploads/2016/03/bg-men-600x400.jpg	1 boisson offerte pour l'achat d'une pizza	100	5.716895	45.750753
2	RESTAURANTS	Pizza Hut	46 Avenue Albert Ter de Belgique, 38000 Grenoble	04 78 48 30 40	pizzahut.fr	https://f.roooin.com/images/menus/93025/header-image.jpg?width=1200&height=630&auto=webp&format=jpg&fit=crop&w=1541&h=6633	La 2e pizza a -25%	30	5.7525035	45.78469
4	BOUTIQUES	Fnac	4 Rue Félix Poulat, 38000 Grenoble	0 825 02 00 20	fnac.com	http://myfnac.com/y3t449e	-10% à partir de 50€ d'achat.	30	5.726269	45.700378

The screenshot shows a web application interface for creating or editing a reward. The page title is "Create or edit a Reward". The interface includes a navigation bar at the top with "Home", "Entities", "Administration", and "Account" links. A user menu is visible in the top right corner, showing "User Extra" and "Reward" options. The main form contains the following fields:

- Type (dropdown menu)
- Name (text input)
- Address (text input)
- Phone (text input)
- Website (text input)
- Image (text input)
- Description (text input)
- Points (text input)
- Longitude (text input)
- Latitude (text input)

At the bottom of the form, there are "Cancel" and "Save" buttons.