Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

TÍTOL DEL TFG: Multiplatform marketplace for Activities and Communication

TITULACIÓ: Doble Grau en Enginyeria d'Aeronavegació i Enginyeria

Telemàtica AUTOR: Krunal Ratan Badsiwal

DIRECTOR: Juan López Rubio

DATA: 20 de octubre del 2022

Title: Multiplatform marketplace for Activities and communication

Author: Krunal Ratan Badsiwal

Director: Juan López Rubio

Date: October 20th 2022

Overview

During the past years a rapid change has occurred in our society and in the world due to covid which has affected humans of all ages from children to elderly.

Everyone has been adapting to these changes in their daily life with different approaches which includes remote work, remote education, online shopping, etc.

Most of the human population has been successful in adapting to these changes, except the elderly who have not been able to find a solution to their daily life such as activities which included meeting with each other safely in an organized and prudent manner without jeopardizing their own health or of the others.

There are already existing tools which tackle this problem, but none specifically designed for accessibility as foremost significance. As such government has proposed a plan to create an application with this particular problem to reach the elderly.

As such the main goal of this project is to create an application which can be used on any platform(device) and have the same UI (User Interface) and UX (User experience) everywhere with ease of accessibility of foremost preeminence.

Finally, this project offers a dissertation on Aerospace appliances of the Marketplace of activities and communication and future possibilities.

INDEX

FIGURES

INTRODUCTION

The Terrassa Health Consortium has seen in the past few years and rapid change in user access requirements for people to be on the internet. And most of the population has been successfully adapted to this situation with more or less comfortability.
But the elderlies have seen in a rise in psychological issues which has given rise to trauma and health consequences.

One of the side effects of being old age is the inability to learn and adapt to changes quickly as of humans of other ages. And Internet is not much different, it’s really hard to get used to the Internet and more so to be able to navigate and do anything without a steep learning required to do so.

As a solution to this problem, The Terrassa Health Consortium had proposed a solution to tackle the health issues and the steep learning curve for the elderlies to tackle this online world. They proposed a Single application Multiplatform Marketplace with communication, news, videos and a personal identity which might resolve the need to change to various different tools(applications) resulting in distraction and confusion.

The proposed solution also required the software to look and function same on all of the platform without changing the design or User Interface to not complicate the understanding and accomplishment of the task to be carried out.

As such the goals were defined and restrictions were put in place, but everything had to be dependent on the user feedback and testing throughout the development cycle. As such the project goal to be successful depended on the user opinion and user testing.

The dilemma of developing a project which depends on the user feedback and user testing report is a paradox in itself. This is not the first time; it has occurred in the software development industry and many theories have been built upon to quantify the predicament and objectively create a solution.

Intrinsically, a widely used solution will be used known as Continuous Integration and Continuous Development (CI/CD), Agile methodologies with objective goals. This project will also go in the through process of the methodology’s implementation for the project fulfillment.

In essence, this thesis has been divided in 3 parts. The first part will define objective the preeminent goals which are essential to the project success defined in **CHAPTER 1. Project Proposal**.
The second part will go upon the selection between native solution or multiplatform frameworks and its advantages and disadvantages contained in **CHAPTER 2. Multiplatform Technologies for Frontend**.
The third part will go on the project implementation for both the backend and frontend part of the project. Which will be spread out in **CHAPTER 3. Backend Development** and **CHAPTER 4. Frontend Development**.

In the last part of this project, direct applications to Aerospace industry and Future appliances of this developed software to modernize it.
This will be explained thoroughly in the **CHAPTER 5. Future Appliances to Aerospace Industry**.

And to conclude the findings, results and research will be defined thoroughly in **CHAPTER 6. CONCLUSIONS**.

CHAPTER 1. Project Proposal

The project has already been defined with abstract and ideal goals, but these goals are too hypothetical to be used as implementation base to rely upon during the development phase.

As such these goals must be translated to objective language with a more precise definition. This chapter will translate the goals and separate it into 3 parts.
Project Requirements, Project Definitions and Project Design.

1. 1 Project Requirement and Constraints

The project requirements for the project are straight-forward and seem simple on the surface. But the simpler a goal looks, the harder it is to develop for, and this applies to any scientific field. And software engineering is no alien to it either.

As such we will break down the abstract goals one by one to be easily digestible for project definitions and requirements.

1.1.1 Goal #1: Multiplatform and looks same everywhere

In objective terms, the definition of Multiplatform means, a software which is able to run on multiple environments which could be hardware as follows

- Desktop
- Mobile
- Smart wear
- TV

And even for each of the hardware there exists multiple software platform which differ wide substantially to be considered one platform in terms of development
For Desktop there exists

- MacOS
- iOS
- Windows
- ChromeOS
- Web Browsers (HTML + JavaScript)

For Mobile hardware platforms there also exists

- Android
- Ubuntu Touch
- Microsoft mobile OS
- Symbian.

For Smart wear on the wrist there also exists

- WearOS
- Blackberry OS
- Tizen OS
- WatchOS

As such for this wide variety of platforms in both software and hardware space, it would be akin to impossible to develop for all of them and any change would require immense workforce and/or time to accomplish any task.

The number of platforms currently available to population in so huge, if we were to strive to reach all of them. We will never finish, and as such we will be restricting the tally of platforms for the application to be developed for to a much-reduced quantity.

The selection of platforms from the hardware side will only be developed for Desktop and Mobile as this are the only ones which already encompasses all 100% of the population digital connectivity through Internet and this includes the elderly and people with partial disabilities.

On the other hand, for the software development constraints per hardware platform requires a thorough study.

As to single out a select few software platforms, we will be restricting the development to only 2 software platforms per hardware platform at most. This is done to have an upper limit for the constraints.

The first study we will be executed for the Desktop environment. Which is further divided in operating system and web.
If we take a look at the (Figure 1) just for the Operating systems and their usage just for Desktop environment.

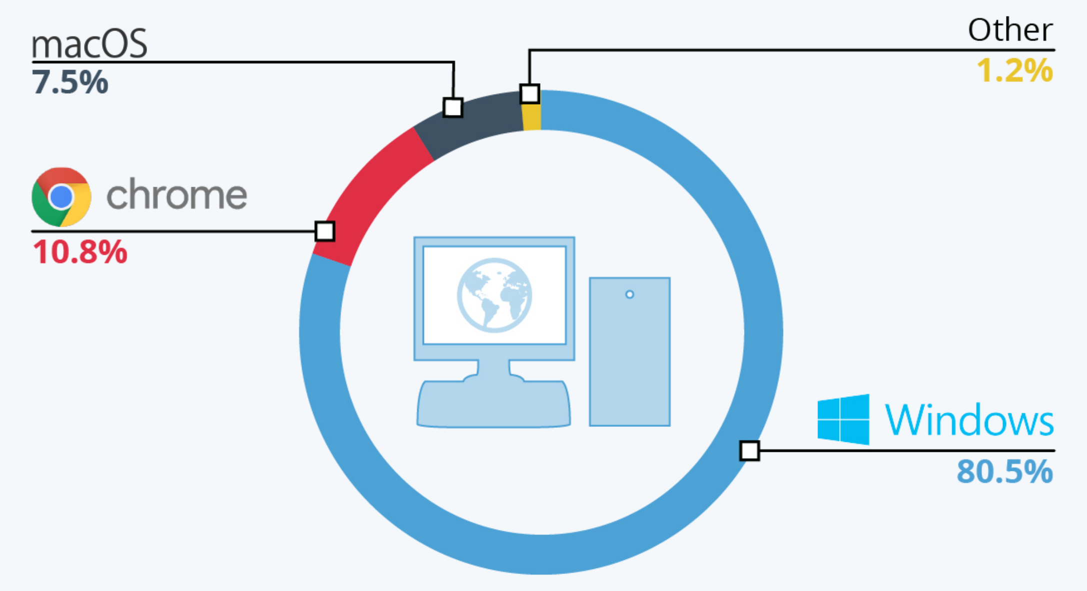


Figure 1. Desktop Environment Market Share Statistics (see reference [1])

We can see that, Windows+Mac+ChromeOs have a market share of 98.8%. As such if we had to choose software environment for the Desktop. We will need to develop for all three of them altogether.

Though due to their existing a software environment called Web which already covers all of the desktop environments. If we were to develop for the Web, we would be able to cover all of them in the market.

As such one of the first requirements for the software development has been defined, we need to develop software for the Web to cover desktop environments.

Now moving to the Mobile environment, and looking at the

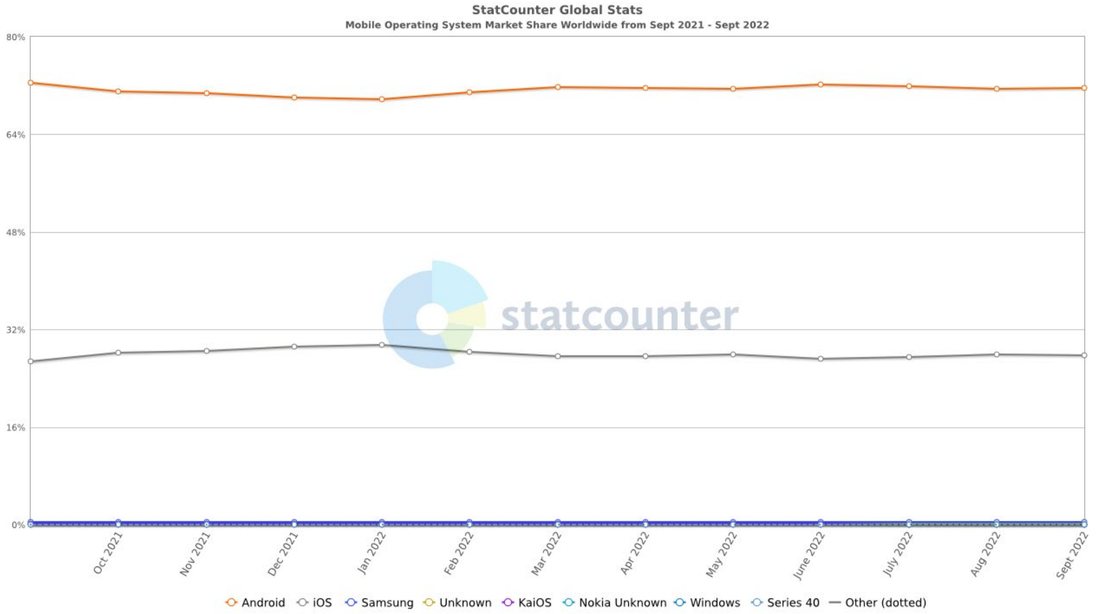


Figure 2. Mobile Environment Market Share Statistics (see reference [2])

Where android has a share of 71.55% and iOS 27.8%. just composing this two together gives a total market coverage of more than 99.35.
And others platforms including Samsung Tizen OS, Linux KaiOS, Microsoft Windows Phone OS and many more only sum up to less than 0.65% which is not favorable enough to add value to the software development.,

But there is still another option we have not talked about, Mobile Web and the market share of the mobile web.
The main underlying fact is the requirement of the application to be on the native platform stores such as Playstore and iOS AppStore.

And stores have banned web apps to be put on their respective platform stores (see reference [5]) and (see reference [6]) as the requirement of mobile Appstore is to be only updatable from their developer interface and should not be able to update itself.

Thus, the only solution to have the app be built for the Mobile Operating System, does not have to be a native solution but cannot be a web app.

1.1.2 Goal #2: Accessible to elderly and people with partial disabilities

One of the challenges of designing any software is the designing phase and how should the application behave under different circumstances and actions.

And one of the goals of this application is to be easily accessible and be effortless to navigate and perform actions.

It's too subjective of what it means to be easily accessible, as such the first task of the goal would be defining this in objective terms.

As the definition of accessible means, a method, approach, arrangement, design, form is easy to reach or get into it. And one of the keywords here in the definition is design and approach. Which is exactly what we are looking for in an application. And Application can be easily accessible if the design and approach of the application is easy to get into or use it.

But the question still remains, what's the definition of a design which is easy and approachable. If we look into the principles of application design and software development. There exist two terms which perfectly fit these definitions previously mentioned.

The terms are User Interface and User Experience are defined as follows

1.1.2.1.a. User Interface: The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website. As such, the types of user interfaces existing currently are numerous and each with their advantages and disadvantages respect to the device it will designed for.

Types of user interfaces available which include

- Graphical user interface (GUI)
  A form of user interface which allows the users to interact with electronic device/hardware through the use of graphical elements such as icons, pointers, images.

- Command line interface (CLI)
  A form of user interface which allows the user to interact with electronic device/hardware through the use of only text-based user inputs.

- Voice user interface (VUI)
  A form of user interface which allows the user to interact with electronic device/hardware through the use of voice for user inputs and to communicate with user. Such as voice back, talk back, etc.

In our case, we will be solely focusing on the Graphical User Interface and including color and partial blindness incapacity as the predominant focus group.

To design for the visually impaired, as there exists multitude of conditions and it's a grayscale and not exactly classifiable. It's not easy to find a common solution and use that everywhere.

As this is not possible, we will solely focus on the text readability, Graphical User Interface element with contrasts, Alt text for all of the image, convey messages with full descriptive information, minimalistic design and multiple color theme modes in the application.

1.1.2.1.b. User Experience: The User experience depends on the design of the application which we have already discussed in the section 1.1.2.1.a User Interface. But it's not the only dependance, as it also hinges on the following
- Information architecture: Content for which the user is actually trying to use the application/software, and a good information architecture makes the use of the application accessible as to allow the user to focus and perform the given task easily. Even though it might now seem, but it is the backbone of the user interface design. Information architecture is based on cognitive psychology, and as such it can be used to objectively design a product with scientific methods. (see reference [7])

The main roots of the cognitive psychology are
Gestalt principles
Where Gestalt principles explore user's visual perception of objects in relation to each other, including similarity, continuity, proximity, symmetry and closure.

Because User interface principles require the bases of user experience and Gestalt principles dictate how the elements should be established in relation to each other to present the information architecture in a cohesive and sensible manner.

We can use Gestalt principles as a guideline during Project design phase of the User Interface. Of course, we have to take into consideration to not burden the user with cognitive load and keep it to a minimal while providing as much content as possible in as simple form as feasible.

1.2.3 Goal #3: Health data to be analyzable

The platform needs to support variable data input from user in form of notification or login question input based on the time of the day or individual performance on the application itself. As such, one of the key project design considerations during the development phase will require a variable and dynamic screen based on the user historic data and/or custom question input.

1.2.4 Goal #4: Easily updatable and changeable application during the development

As the application did not have any defined features set or requirements, and everything depended on the context of development and user needs as we progressed through testing and development cycle.

The most important goal, actually for the development side was actually it to be easily flexible and versatile. So that any new feature could be added and coupled together with the existing feature every new cycle. (see reference [8])

1.2 Project Methodology

Any new project may it be in real-world or software world, all of them require a special attention to the work methodology for the project development cycle.
As this provides the stability a project requires for it to be a success and be able to strive and achieve the desired goals it had set in first place.

As such, we will begin with a small study of available methodology available to us for the software application projects and which suits the best for us.

We will only be analyzing the top architectures used in 99.999% of any projects which are most common. Of course, each company can tweak and modify this software architecture methodologies to best fit their circumstances even more closely to their needs. (see reference [9])

1.2.1 Agile Software Development Methodology

Agile software development methodology is a disciplined software management process to minimize risk which includes bugs, cost overruns and too variable changing requirements when adding any new functionality or feature. The development cycle of Agile Software development Methodology is called iterations and they last from a week to anywhere a month.

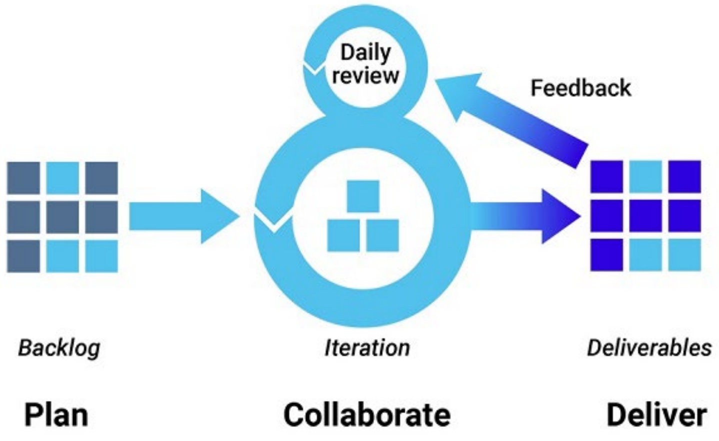


Figure 3. Agile Software Methodology

As we can see, the planning phase must always occur before any iteration to the software and any new iteration must be completed and be delivered before moving onto next features.
Thus, the advantages of the Agile Software Development Methodology are that it allows the software to be release in iterations. Iterative release improve efficiency by allowing teams to find and fix defects and align expectation early on. They also allow users to realize software benefits earlier, with frequent incremental improvements.

Though the Agile development methodology relies on real-time communication and thus any new member or change with external client require a intensive communication to reach to the current project cycle or heavy use of documentation of each cycle from each developer and everybody involved in the project.

1.2.2 Scrum Development Methodology

Scrum methodology prioritizes customer needs and accurate and precise record of the user requirements before any real technical development begins. And any change into user requirements is only possible in the next cycle of the scrum development cycle and has to start the process from the head.

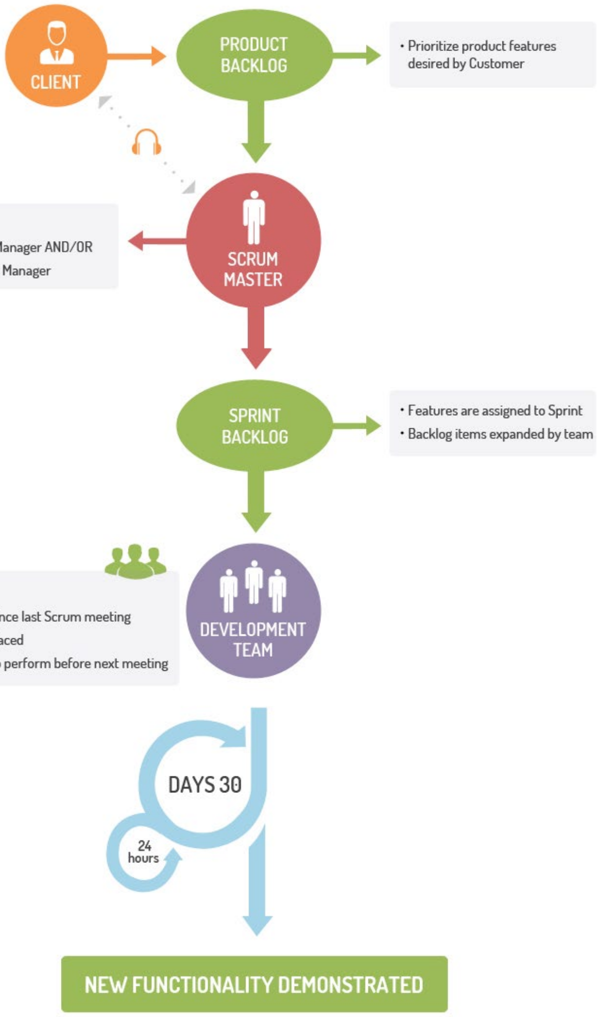

CLIENT
PRODUCT BACKLOG
• Prioritize product features desired by Customer
CAN BE
• A product Manager AND/OR
• A Technical Manager
SCRUM MASTER
SPRINT BACKLOG
• Features are assigned to Sprint
• Backlog items expanded by team
SCRUM
• Activities since last Scrum meeting
• Obstacles faced
• Activities to perform before next meeting
DEVELOPMENT TEAM
DAYS 30
24 hours
NEW FUNCTIONALITY DEMONSTRATED

Figure 4. Scrum Development Methodology

Thus, the advantages of the Scrum development cycle are its static and precise language and error proof development cycles. The cycle is entirely in the hands of superiors and control the pace and work. The user requirements are also managed by the scrum masters and is discussed by the team, before any work begins. The work is managed and the requirements are easily definable to customer or business.

And the disadvantages of the Scrum are the time and pace. As there is no causal pace of the work defined and nothing can be changed during a cycle development phase, if the work has already begun. It’s really hard to change any feature after having pass through review in the initial phase.

1.2.3 Waterfall Methodology

Waterfall methodology has a similar structure to Scrum methodology except for the fact, there is no Development team input till the second half of the cycle. As such, the higher decisions of the project development are all taken by the project manager and development team input is only taken into consideration after the project planning and architecture design of the feature already being completed by the project manager.

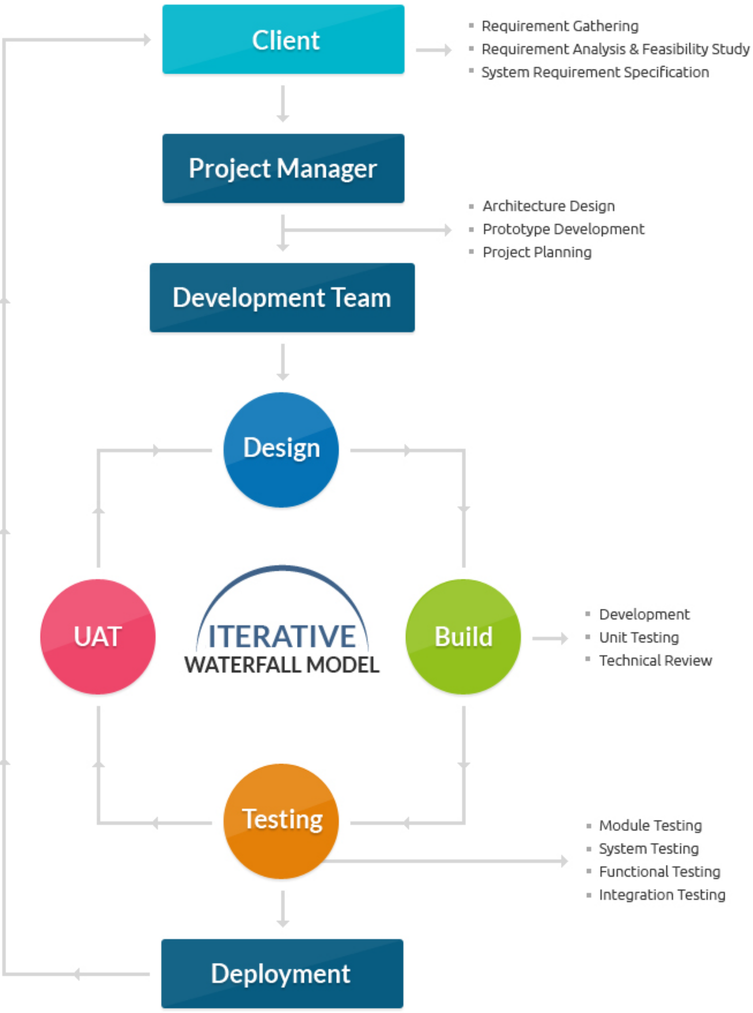


Figure 5. Waterfall Methodology

As such, the waterfall methodology is best suited in environments where the business decisions are well separated from the developer and no input is required.
This gives advantages in situation where the business solely depends on the client needs and does not have a set time to develop thy feature.
Though this gives rise to disadvantages as such as, it's impossible to give a time frame a respective feature implementation or project changes required to do so.

1.2.4 Prototype Methodology

In the Prototype methodology is based on the rapid lifecycle methodology where the product or application being developed is just for testing. May it be for market test or internal testing, but this will never be the final product reaching the user in first place. As such, this type of methodology is generally used in Minimum Viable Product software cases, where just a minimum product is required for certain cases.

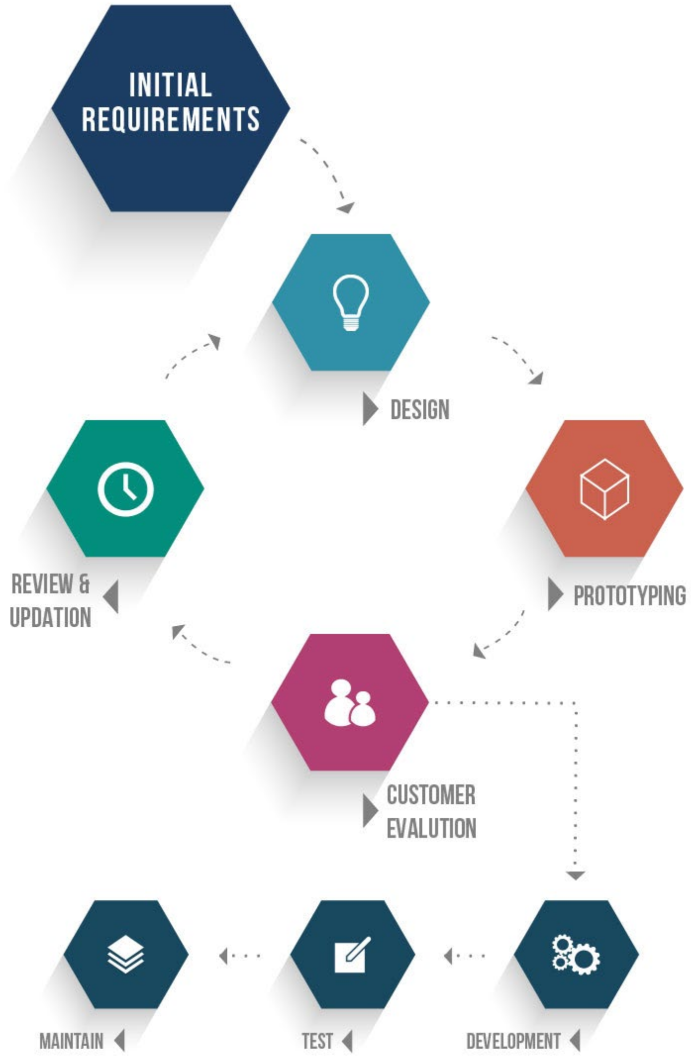


Figure 6. Prototype Methodology (see reference [3])

The advantages of a Prototype methodology are the fact that, it does not require the application to be stable and as such changes can be made disregarding errors and

problems per specific case and just concern about the features.
The disadvantage of such methodology is the fact that the time spent on developing for such a product will in the end be wasted and never be recovered for the actual software application that would be required to be built after the MVP as the prototype will never be usable in the real world under real world usage.

1.2.5 Rapid Application Development

Rapid Application development takes a step further into removing any of the extra steps for a software development cycle and concentrates into a short cycle of development and features.

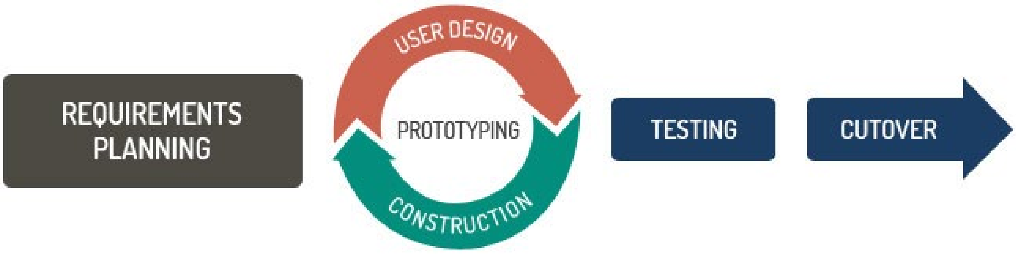


Figure 7. Rapid Application Development

The advantage of such approach is the rapid development cycle and feature integration. No planning phase which usually slows down the process of new feature implementation and changes.

The disadvantage of such cycle is no planning phase, which allows the developer and project manager to be able to review it properly without stress and pressure. More bugs are able to escape to testing and customer due to no testing phase not closely loop back into prototyping phase to improve the final product delivered to users.

1.2.6 Conclusions

After having understood thoroughly the software architecture methodology we can start to understand the advantages, disadvantages and the nuances of the product to which its best suited for.

Though in our case, not a single particular software methodology suited our needs for the product development cycle. But which most closely resembled our need was Rapid Application development methodology but it needed to be modified to make it fit perfectly for our edge cases and development needs.

As such we modified the Rapid Application development to fit our needs in the following way *Figure 8. Custom Development Methodology*.

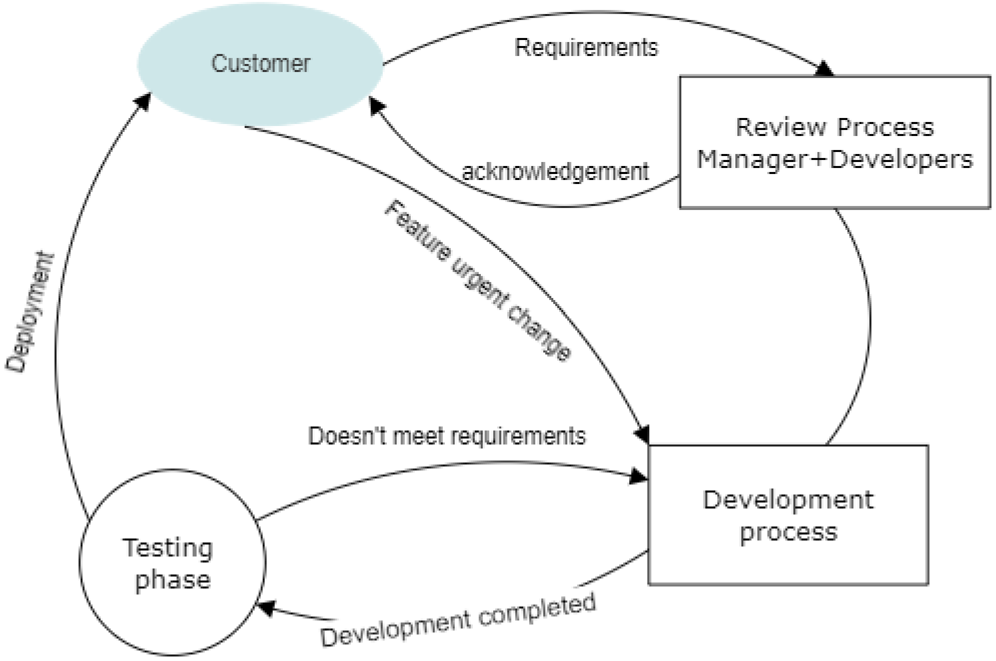


Figure 8. Custom Development Methodology

The foremost importance is given to customer during any phase of the development process. As such, the customer in this case the CST (Consorci Sanitari de Terrassa) could disrupt the development process and ask for change. As such, all of the phases can be exited early and disrupted to jump to next cycle.

Even further so, there is no real cycle, even though the cycle follows a traditional direction of requirements →review →development → testing → next cycle. All of the cycles actually just follow a new feature request development process. Intrinsically, this means the testing phase is the only reliable measure of quality and value. And time to time, errors would escape to users which would require further changes to code and the vicious cycles repeats.

But the main advantage of such a cycle is the closed denseness of feature abundance in really short time with full control on the features and the developer have perception of the changes and time required to implement the feature with full control.

But few of the disadvantage is the requirement of complete project codebase and requirements through the whole development process and cycles. Which makes it impossible for new developers to join in and start adding value. As there exists no documentation beforehand about the project.
Nevertheless, due to the fact that the team was only composed of just 2 developer and later on just composed of 1. This made the implementation and pursuance of this methodology possible.

1.3 Project Design

The project design will consider the project requirements and goals and transform it into project skeleton for the developers to understand. But as during the lifecycle of the project, the requirements kept changing. We will be using the final features implemented and write them from the perspective of the end user or customer.

1.3.1 Use cases

Use cases are a description of the ways in which a user interacts with a system or product. A use case can establish both success and failure scenarios for a given interaction between user and product in our case software application. (see reference [10])

All of the available use cases or in user experience perspective, the user stories are defined in the ANNEX 1. Use Cases, user **stories**.

1.3.2 Platform models

The platform models or also known as data models are a visual representation of data elements and the connection between them. The attributes or data elements inside data models represent the reality as they usually record people, places, objects, events. This data models also describe the way data is stored and retrieved. (see reference [12])

But the main use case of Data models is to facilitate communication between business and technical development by accurately representing the requirements and constraints of the system or project goals and accurately representing the response of the information system.

Data models are composed of the following main components:

- Data Sources: Data source are elements of the data model that describe real world sources of data in the software or application.

- Data Types: Actual Data attribute or type. The data type can be simple such as number, text, boolean, List or can be abstract such as another data model.

- Data Items: Elements of the data model that represent real units of data stored in a data source.

- Link or relationship: relationship between data types and data items. As the data items can be just singular and direct or multiple composed of multiple data types.

- Even Sources: Event sources are special types of data sources. Each event source represents an application that stores and manages events.

Now that we have a clear understanding of Data models, we need to define one for ourself based on the given requirements and constraints. But before that we can decide, we have to select the type of Data model we require.

There exist three types of Data models Types:

Conceptual Data Model: This type of Data model defines WHAT the system contains. This model is typically created by Business stakeholders and Data Architects. The purpose is to organize, scope and define business concepts and rules.

Logical Data Model: This type of Data model defines HOW the system should be implemented regardless of the Database management system. This model is typically created by Data Architects and Business Analysts. The purpose is to developed technical map of rules and data structures.

Physical Data Model: This type of Data Model describes HOW the system will be implemented using a specific DBMS system. This model is typically created by DBA and developers. The purpose is actual implementation of the database.

In our case, we will define the data models of type Logical Data model type. The main reason of jumping from Conceptual Data model type directly to Logical Data model is due to the fact how simple our system requirements are and basically the actual representation from Logical Data Model type and Physical Data model type would have been the same.

To define the Logical Data model Type for our different Use case and requirements, we will be using the Unified Model Language also known as UML.

In UML, the data models are defined as follows

Class Name	
Type	*Attribute*

Figure 9. UML Data model representation

Where the class name represents the Data Model itself, the type represents the Data Type and the attribute represents the Data source and Data item.

In the following figures, will define the User (Figure 10. User Data model), Activity (Figure *11*. Activity data model), Chat and message (Figure *12*. Chat and Message data model), Question (Figure *13*. User question model) and Video (Figure 14. Video model) Data models in UML language.

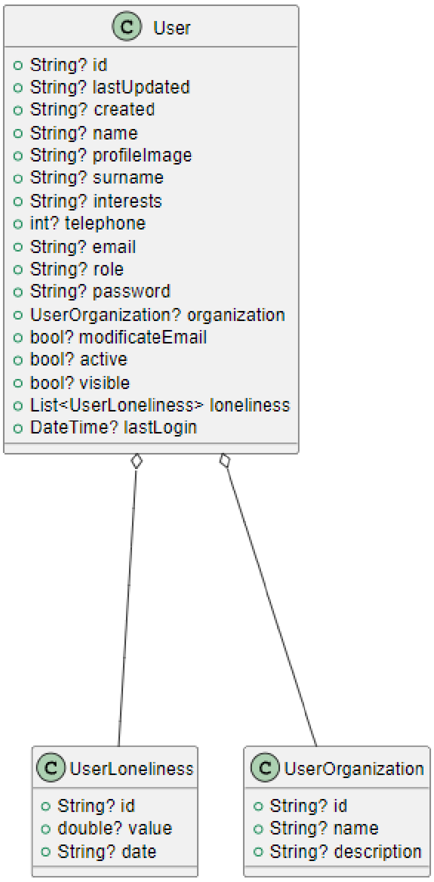
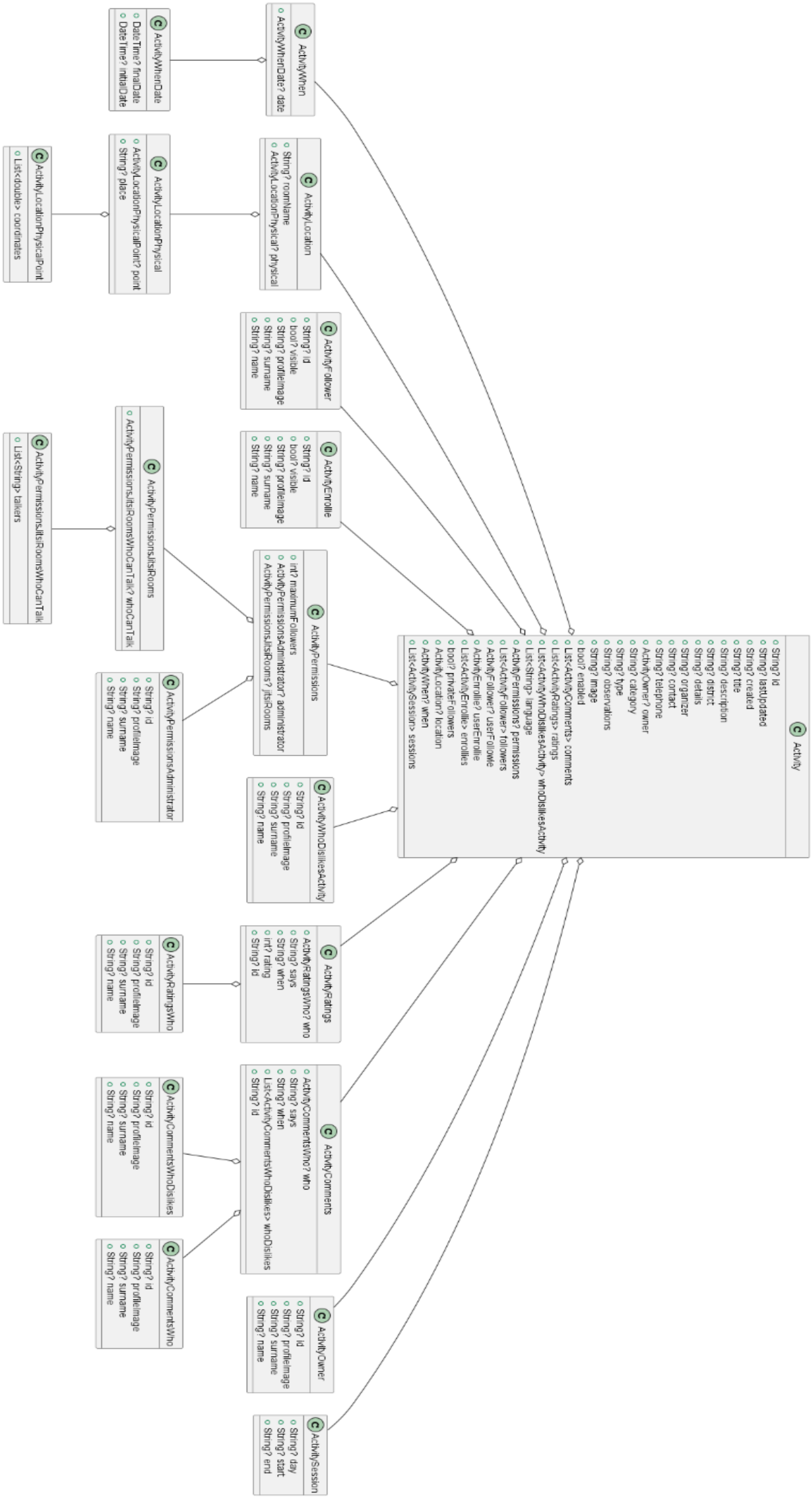


Figure 10. User Data model

Figure 11. Activity data model



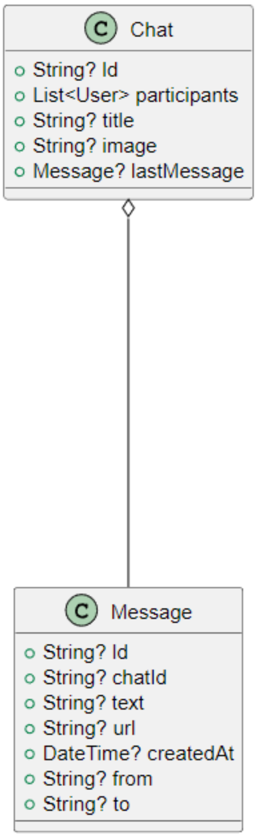


Figure 12. Chat and Message data model

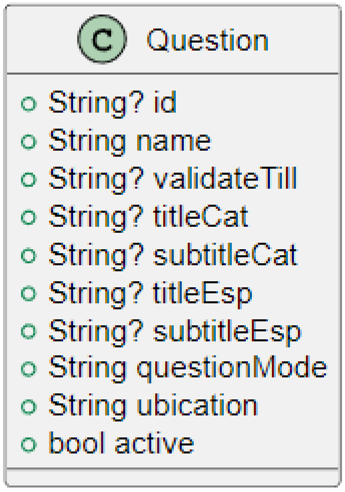


Figure 13. User question model

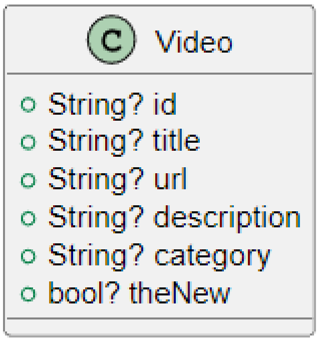


Figure 14. Video model

We have defined data models, which has data fields best fit and adapted to the use cases *1.3.1 Use cases* and goals (Project Requirement and Constraints).

We can finally move to next step, which is to choose what tools we need to actually use to develop our application with and which are the tools which accommodate to the requirements defined previously.

CHAPTER 2. Framework selection

Now that we have defined the requirements, constraints, development methodology, user process and actions, data models for the application. We can finally move to the next step which will be to select proper tools to develop our application with and which best adapts to the needs proposed earlier.

The framework selection will be defined in two steps. As we will also separate the data access layer from the presentation layer which will be visible to actors or users in our system.

This separate data access layer is also known as backend whose core function is to find and deliver data to the presentation layer. The presentation layer is also termed as the frontend. As this layer is the front visible part of the application to users.

2.1 Backend

There are many available options on the market currently to manage and deliver data to frontend. But as one of the requirements of the product. An inhouse tool built known as route injector has to be used. (see reference [22])

This requirement was established in place to lessen the burden from data layer specification and just focus on the frontend layer development as most of the features requested will result in implementation to frontend and the development cycle would have been much slower otherwise.

2.1.1 Route Injector

The main purpose of this framework is, starting from a model definition, generate automatically both a CRUD REST API and a Backoffice to work with the specified models. (see reference [11])

The main difference of this framework with other available is the degree of customization that can be applied to the API and the Backoffice.

This framework is an opensource framework with the source code completely available online.

The main features of this framework are as follows

- Focus on prototyping
- Automatic generation for framework entities
- Administration panel
- API Documentation
- Extend functionality by plugins

This feature set and easily adaptation to data models allowed even faster pace of development cycles.

2.2 Frontend

For the frontend or presentation layer, there also exists multitude of frameworks (see reference [23]), each with its own advantage and disadvantage. But it all depends on the demands of the project, constraints and design adaptability.

As such, we will discuss between few selected options. This were selected from plethora of options based on the following requirements

- Ability to generate platform specific code
- Single codebase which means write once run anywhere
- Fast
- Stable
- Consistent user interface throughout devices or platforms with constraint in our case are only for web, mobile android and mobile iOS.

2.2.1 Native

Native is not actually a framework, but the platform itself. In our case the mobile and web. In our case, the constraints enforced to use specific platforms are

- Web: Which would mean the use of HTML + JavaScript or some web frameworks like Angular or React.
- Mobile Android: Which would mean the use Google android Kotlin
- Mobile iOS: Use of Swift with Apple Ecosystem.

And any app developed for any one of the platforms won’t work on the other platform. Without any other option than rewriting the whole code and writing it into the other platform language.

Even though there are downsides for the native solutions. The main advantage of the native solution is the fact. They provide exception user experience and show high performance. As they are built with tools specific to each platform.

As such the feature of native solution are

- Superior runtime optimization and low battery usage
- Faster performance
- Access to platform specific up to date features
- High Quality UI/UX specific to platform
- Better store positioning
- Full access to device feature set

But the disadvantages of native solutions are also many

- No multiplatform support
- Higher development cost
- No reusable code
- More time for feature development

2.2.2 Hybrid platform framework

Hybrid platform framework are actually a blend of both native and web solution. This kind of solutions requires the code to be written in web languages and when compiling for specific platform. They transpile the code into a plugin and inject it into native code of the respective platform which allows the application to access native features of the platforms while maintaining cross platform support.

As such the features of the Hybrid App Development are

- Portability Support
- Can manage multiple platforms
- Cost effective
- Rapid development time
- Easier to develop the code

But the disadvantages of the hybrid app development are also as follows

- Sluggish performance
- Hard to develop complex solutions
- Limited platform specific feature access
- One codebase creates issues with device specific feature
- Complex task to achieve good UI/UX for a user.

Currently available frameworks with hybrid solutions are React Ionic and Cordova.

Framework	React Ionic	Cordova
Language	HTML5	HTML5
Rendering	Webview	Webview
Popularity	Medium	Medium
Cost	Priced	Free
Platforms	Android/iOS, Web, Windows, MacOS	Android/iOS, Windows, iOS
Performance	Sluggish	Sluggish

Figure 15. Hybrid Framework (see reference [14] and [15])

2.2.3 Cross platform framework

Cross-platform apps are often confused with hybrid ones. As they seem similar offer same features as the hybrid solution. The ability to develop for multitude of platforms with singular code base.

But the main differentiation point for cross platform frame is the ability to compile the code written in framework language to native application code of the platform.

This gives many advantages, but most specifically it speeds up the application and decreases the energy consumption vastly.

It also increases the feature set available to be used in the application vastly larger than the hybrid solution. As the app is also compiled to native language of the platform. The UI/UX is also near native and straightforward to the platform specific.

As such the features of the Cross App Development framework are

- Developer friendly
- Reusable code
- Plugin Support
- Cost effective
- Fast compilation
- Simple testing

Of course, there are still limitations and disadvantages

- Slower performance compared to native
- Access to OS functionality is reduced but not as much as hybrid
- Interaction with another native app is greatly hindered.

Currently there exists many cross platforms frameworks

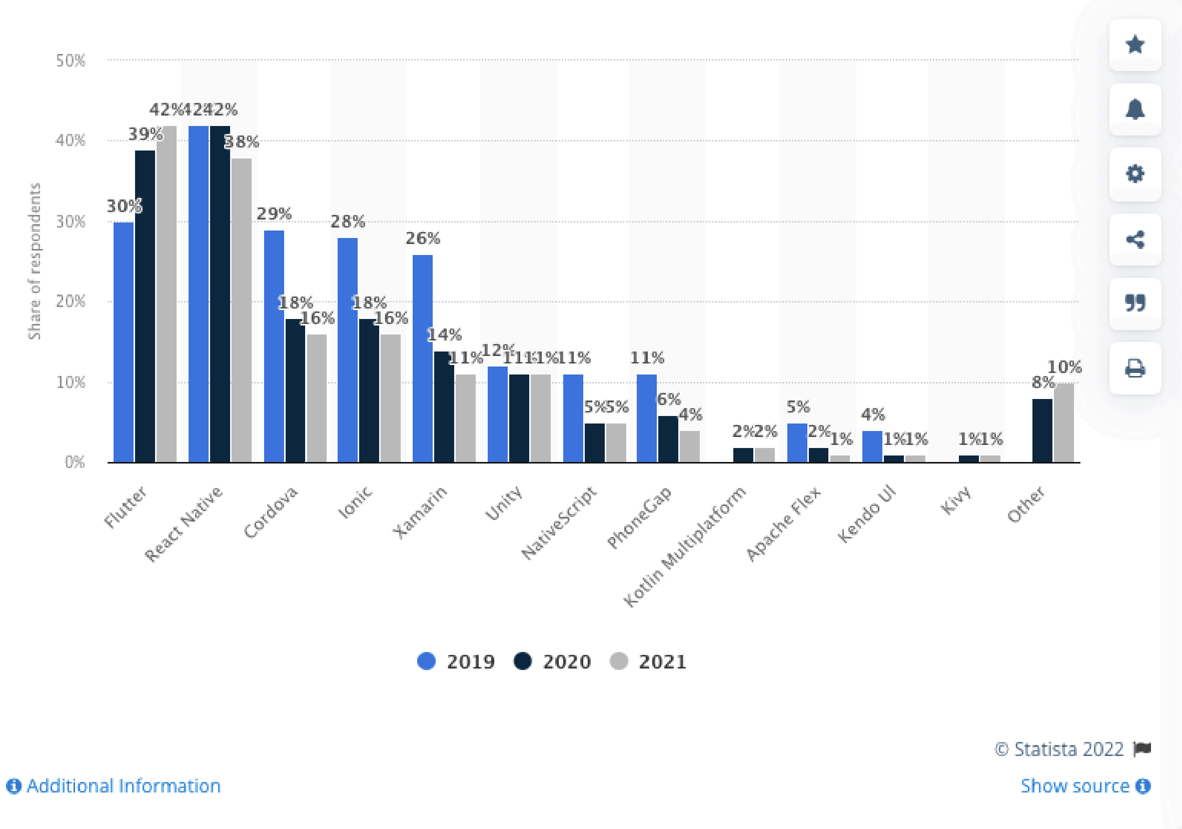


Figure 16. Cross platform framework market share (see reference [13])

But only Flutter, Xamarin and React Native are true cross platform library with a market share over 15% with active community as we can see in the Figure 16. Cross platform framework market share (see reference [13]).

Out of which, only flutter has been showing growth and at that increasing at a rapid pace, while all others have been declining over the years. As visible in the Figure *15.*

Cross platform framework market share.

This is mainly due to two reasons, first flutter changes the way application are developed for cross platform, with a unified toolset to debug and optimize. Second the language itself, which is Dart. A new language with almost all of the feature any language could ask such as

- Auto garbage collection
- Declarative and functional
- Easy syntax
- Easy to learn
- Concurrency
- Object oriented
- Type safety
- Platform independent

If we now compare the cross platform. More specifically, just React Native and Flutter. Between this cross-platform frameworks Flutter has a simpler approach to development with its simple language, development tools and close to native performance.

This is due to the face, While React Native uses JavaScript for its build and develop which requires a compiler to translate the code line by line during run time as the language is dynamic. Flutter is both Ahead of time(AOT) and Just in time(JIT) compile language depending on the phase of development. While we are debugging the flutter uses Just in time compile to be able to catch any bugs and debug the application line by line. Whereas, to make the application production ready for users, the flutter application compiles the code into native code as to be executed without waiting needing to translate code during run time.

As such currently the best cross platform framework is Flutter and also few of the reasons why it's growing at such an absurdly immense rate.

2.2.4 Result

Now that we understand the nuances of different framework. First, we have discarded the native development for each platform as it would be impossible to develop any feature on time and cost effectively.

As such, just comparing between Hybrid Framework and Cross platform framework. The only framework type that fit's the requirement and constraints are cross platform framework. As with hybrid framework, there is no method to develop for web in singular code base without looking for plugins and extra development time on top of the feature development cycle.

Thus, our final choice of framework for the presentation layer or the frontend will be Flutter, for all of our platforms which include Web, iOS and Android.

CHAPTER 3. Development

There is not going to be a solution comparison to select the best backend which meets the requirements as we are going to use an inhouse built opensource backend solution called Route Injector. (see chapter 2.1 Backend)

3.1 Route Injector - Backend

The development cycle for the backend is short due to the fact that, the route injector generates all of the delivery and data management code itself and we just need to define the models in JavaScript language as a Route Injector typed model. (see reference [16])

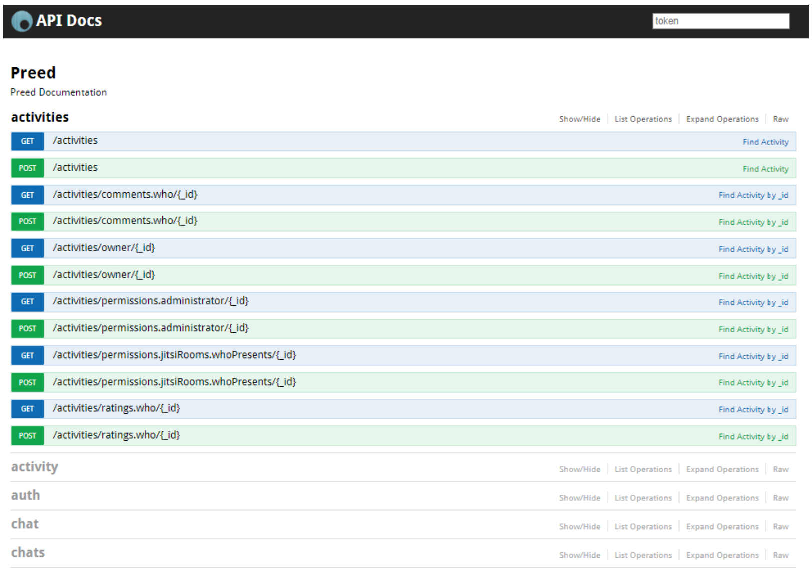
The generation of the backend code for the first time follows a 4-step process

Step1. Create an empty app
Step2. Generate a route injector project
Step3. Add Admin user in charge of managing data layer.
Step4. Add new models in the model's folder of the generated route injector project with the data models as specified in the 1.3.2 Platform models.

As such a complete project is generated with following data information layer

- API endpoints with auto API documentation
- Backoffice: To manage, oversee and supervise the data of user.

The Available API documentation was be generated in Opensource API documentation language. we will let the reader to access the code base of the application available here in the ANNEX 3. Backend or Data Layer Codebase repository.



Preed
Preed Documentation

activities

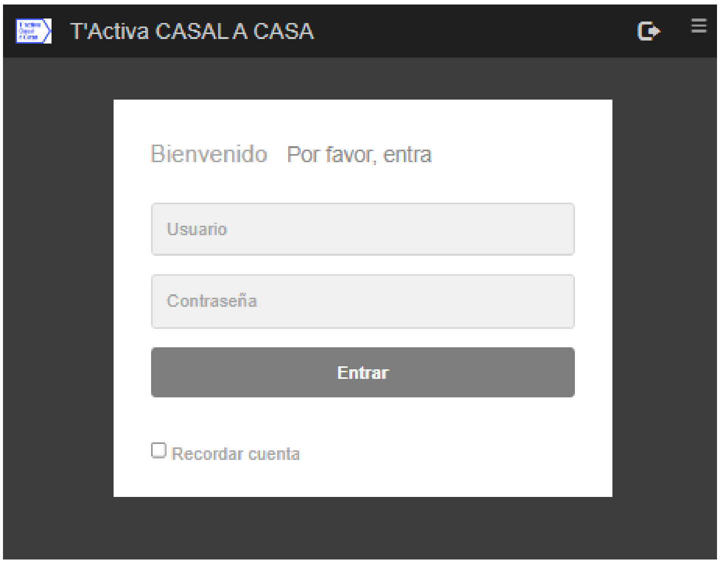
GET	/activities	Find Activity
POST	/activities	Find Activity
GET	/activities/comments.who/{_id}	Find Activity by _id
POST	/activities/comments.who/{_id}	Find Activity by _id
GET	/activities/owner/{_id}	Find Activity by _id
POST	/activities/owner/{_id}	Find Activity by _id
GET	/activities/permissions.administrator/{_id}	Find Activity by _id
POST	/activities/permissions.administrator/{_id}	Find Activity by _id
GET	/activities/permissions.jitsiRooms.whoPresents/{_id}	Find Activity by _id
POST	/activities/permissions.jitsiRooms.whoPresents/{_id}	Find Activity by _id
GET	/activities/ratings.who/{_id}	Find Activity by _id
POST	/activities/ratings.who/{_id}	Find Activity by _id

activity
auth
chat
chats

Figure 17. Backend API Documentation

Where each of the documentation is an API endpoint to perform an operation on the Data layer. Which can be accessed by any frontend framework and not excluding textual frontend.

The Backoffice is also generated with an authentication and data management layer as a browser page.



T'Activa CASAL A CASA

Bienvenido Por favor, entra

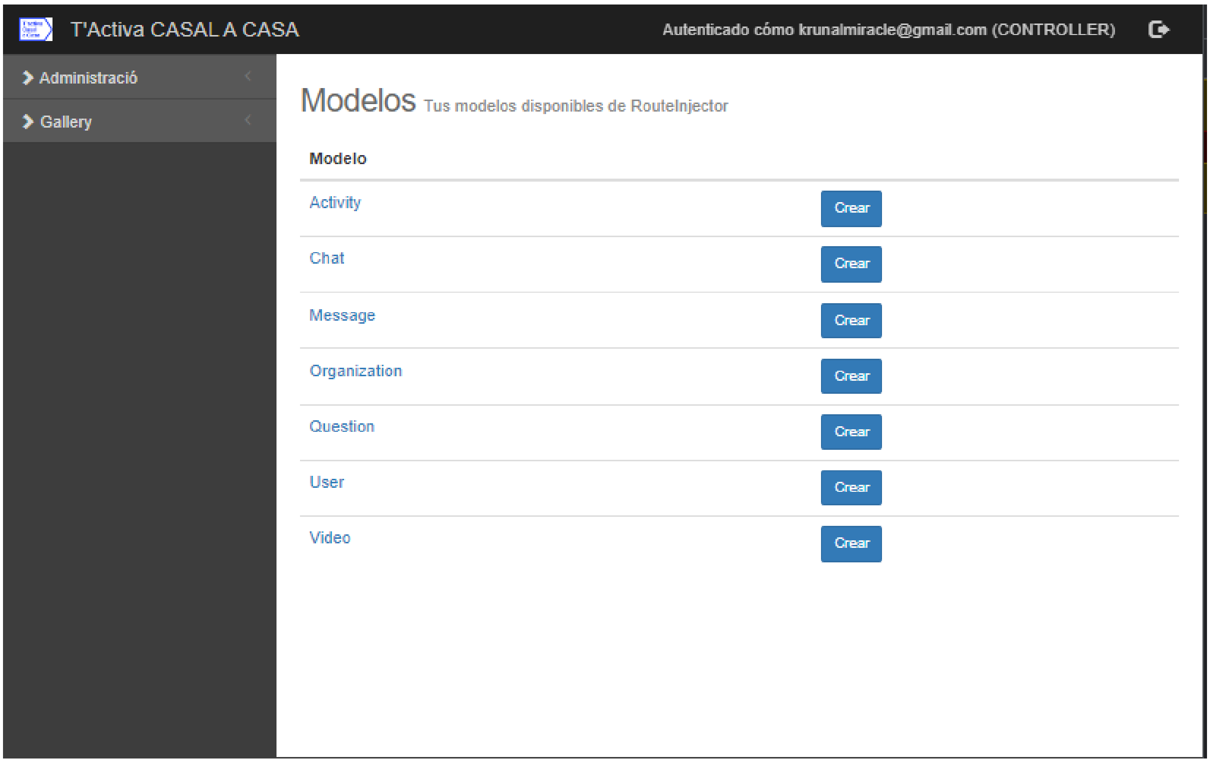
Usuario

Contraseña

Entrar

□ Recordar cuenta

Figure 18. Backend Backoffice



T'Activa CASAL A CASA

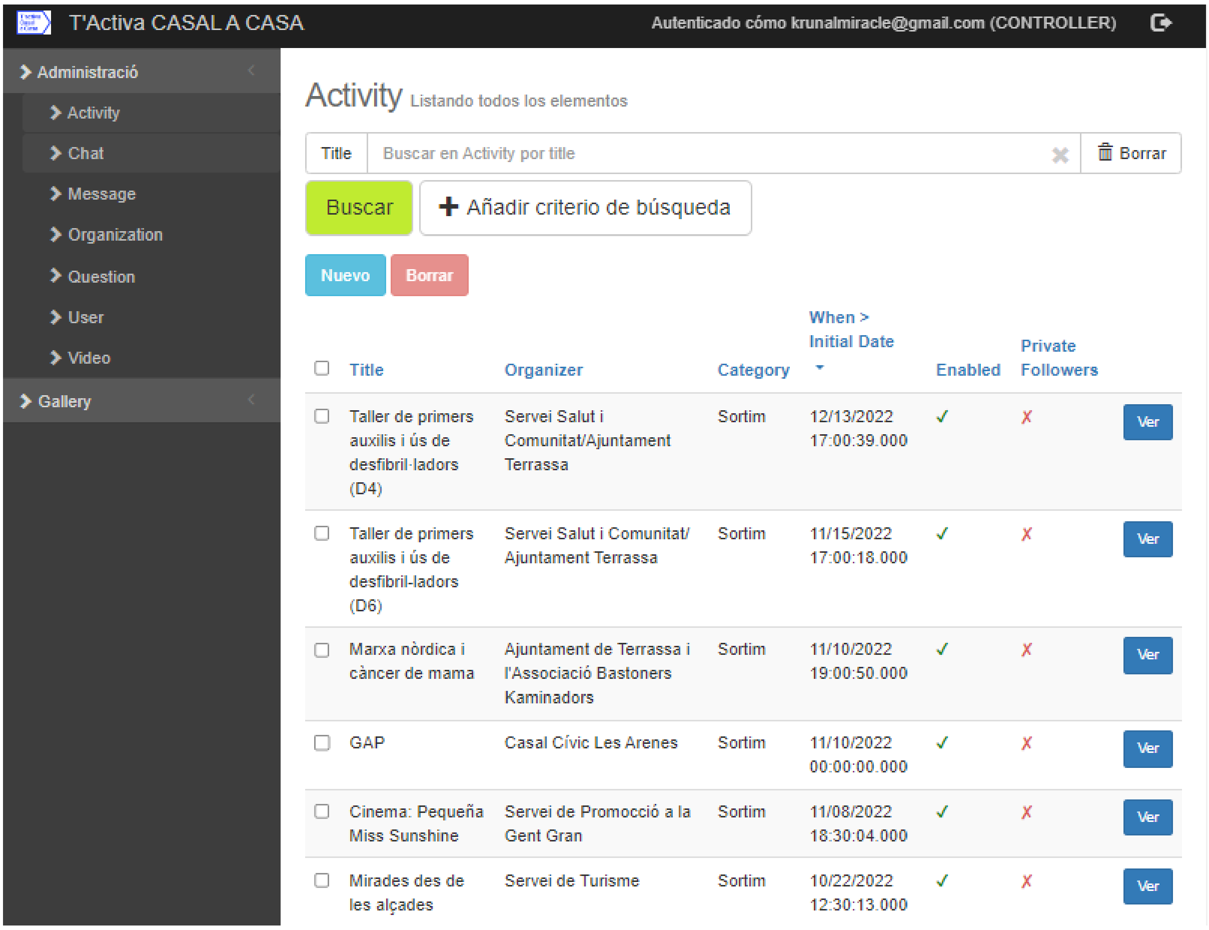
Autenticado cómo krunalmiracle@gmail.com (CONTROLLER)

Modelos Tus modelos disponibles de RouteInjector

Modelo	
Activity	Crear
Chat	Crear
Message	Crear
Organization	Crear
Question	Crear
User	Crear
Video	Crear

Figure 19. Backend Backoffice Data management

Where each element in the admin page is a model defined in the data layer. And selection of any data model. Gets us to the available list data models currently available for consumption.



T'Activa CASAL A CASA

Autenticado cómo krunalmiracle@gmail.com (CONTROLLER)

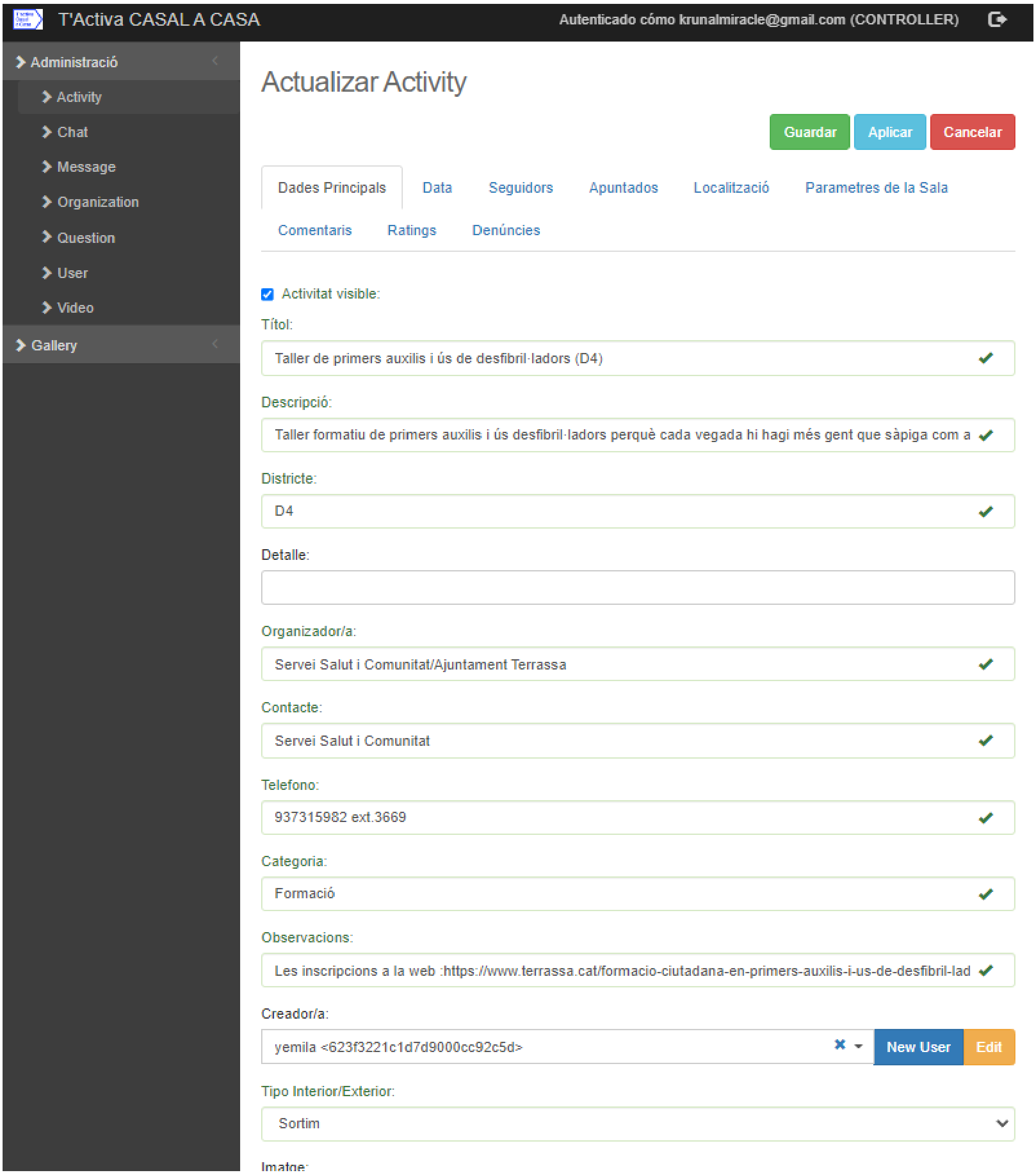
Activity Listando todos los elementos

Title | Buscar en Activity por title | Borrar

□	Title	Organizer	Category	When > Initial Date	Enabled	Private Followers	
□	Taller de primers auxilis i ús de desfibril·ladors (D4)	Servei Salut i Comunitat/Ajuntament Terrassa	Sortim	12/13/2022 17:00:39.000	✓	✗	Ver
□	Taller de primers auxilis i ús de desfibril-ladors (D6)	Servei Salut i Comunitat/ Ajuntament Terrassa	Sortim	11/15/2022 17:00:18.000	✓	✗	Ver
□	Marxa nòrdica i càncer de mama	Ajuntament de Terrassa i l'Associació Bastoners Kaminadors	Sortim	11/10/2022 19:00:50.000	✓	✗	Ver
□	GAP	Casal Cívic Les Arenes	Sortim	11/10/2022 00:00:00.000	✓	✗	Ver
□	Cinema: Pequeña Miss Sunshine	Servei de Promocció a la Gent Gran	Sortim	11/08/2022 18:30:04.000	✓	✗	Ver
□	Mirades des de les alçades	Servei de Turisme	Sortim	10/22/2022 12:30:13.000	✓	✗	Ver

Figure 20. Backend Backoffice CRUD compatible models

These models are all CRUD (Create, read, update and delete) compatible. Each data model entity can be further expanded to see the detail and perform the CRUD operation on a singular entity.



Actualizar Activity

☑ Activitat visible:

Títol: Taller de primers auxilis i ús de desfibril·ladors (D4)

Descripció: Taller formatiu de primers auxilis i ús desfibril·ladors perquè cada vegada hi hagi més gent que sàpiga com a

Districte: D4

Detalle:

Organizador/a: Servei Salut i Comunitat/Ajuntament Terrassa

Contacte: Servei Salut i Comunitat

Telefono: 937315982 ext.3669

Categoria: Formació

Observacions: Les inscripcions a la web :https://www.terrassa.cat/formacio-ciutadana-en-primers-auxilis-i-us-de-desfibril-lad

Creador/a: yemila <623f3221c1d7d9000cc92c5d>

Tipo Interior/Exterior: Sortim

Imatge:

Figure 21. Backend Backoffice Singular data model entity

3.2 Flutter - Frontend

The frontend will be developed using a Cross platform framework called flutter. As we have already selected with the previous comparison of framework for multiplatform compatibility and singular codebase (see chapter 2.2.4 Result). Because the role of frontend development is just a presentation layer, most of the data will just be received and presented in an accessible way.

Considering this fact, we will let the reader to access the code base of the application available here in the ANNEX 2. Frontend or Presentation Layer Codebase repository.

Now in this section, we will only show the results of the development phase of the frontend and different views of which the presentation layer is composed of. In the first part we will be taking a look at the Web frontend accessible in the desktop for Windows, MacOS and Linux.

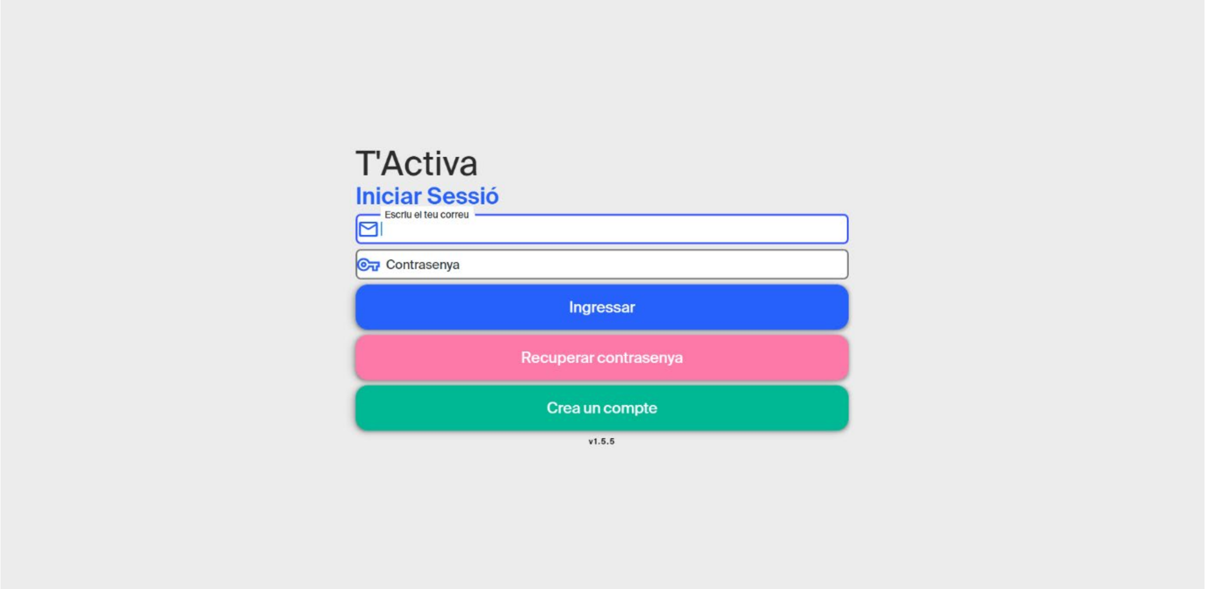
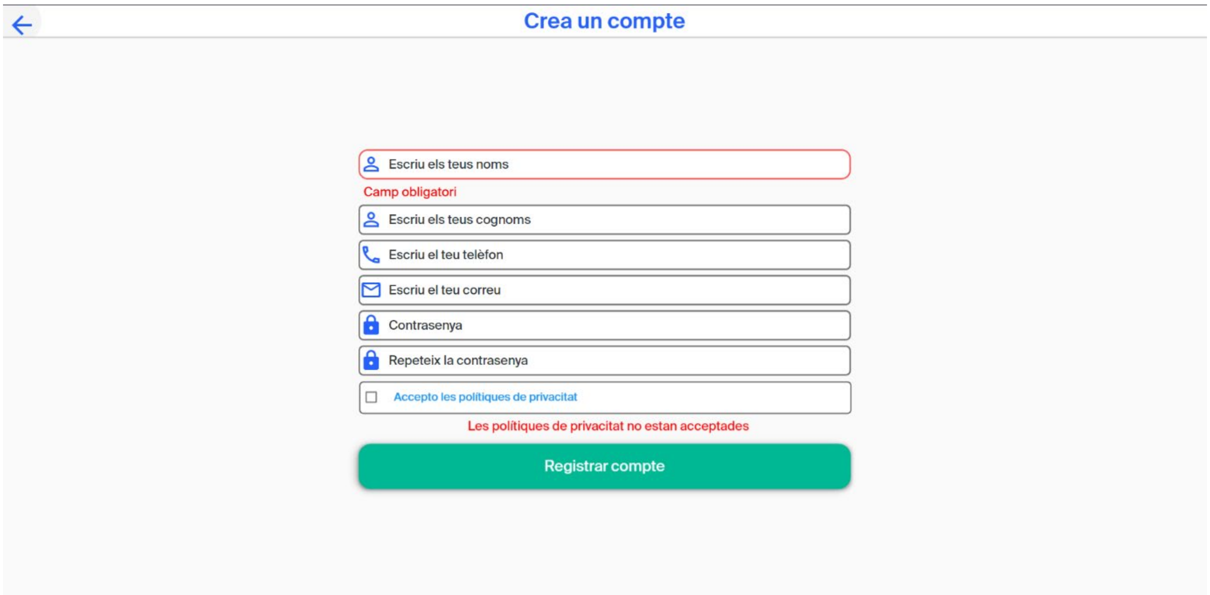


Figure 22. Frontend Web Authentication



Crea un compte

Escriu els teus noms

Camp obligatori

Escriu els teus cognoms

Escriu el teu telèfon

Escriu el teu correu

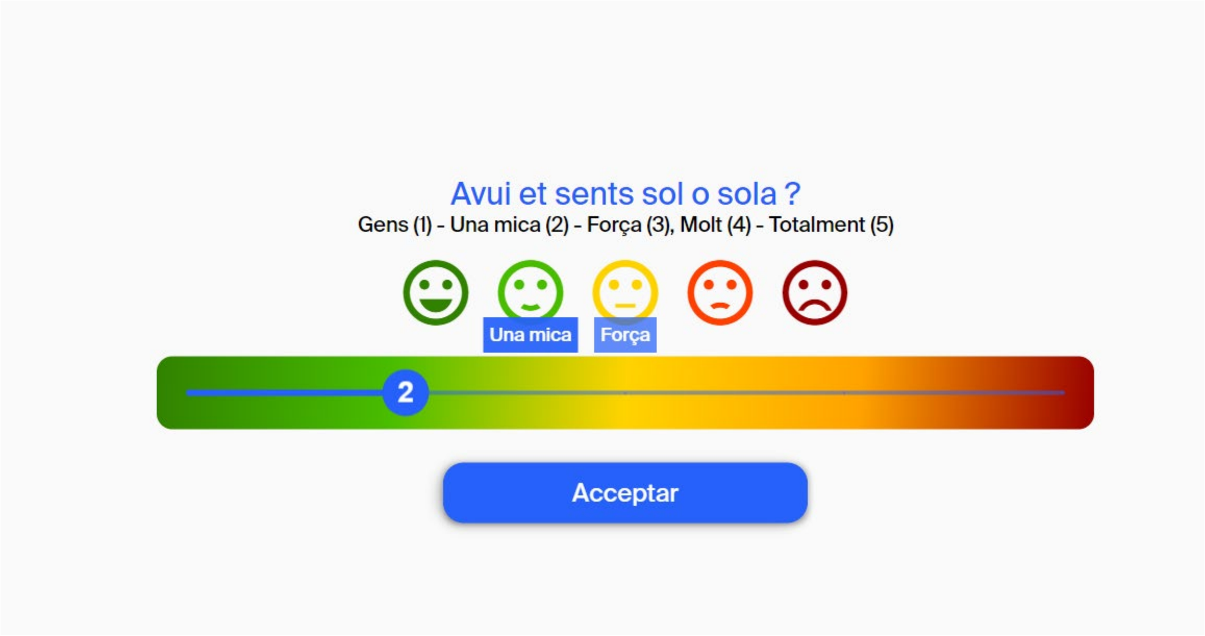
Contrasenya

Repeteix la contrasenya

□ Accepto les polítiques de privacitat

Les polítiques de privacitat no estan acceptades

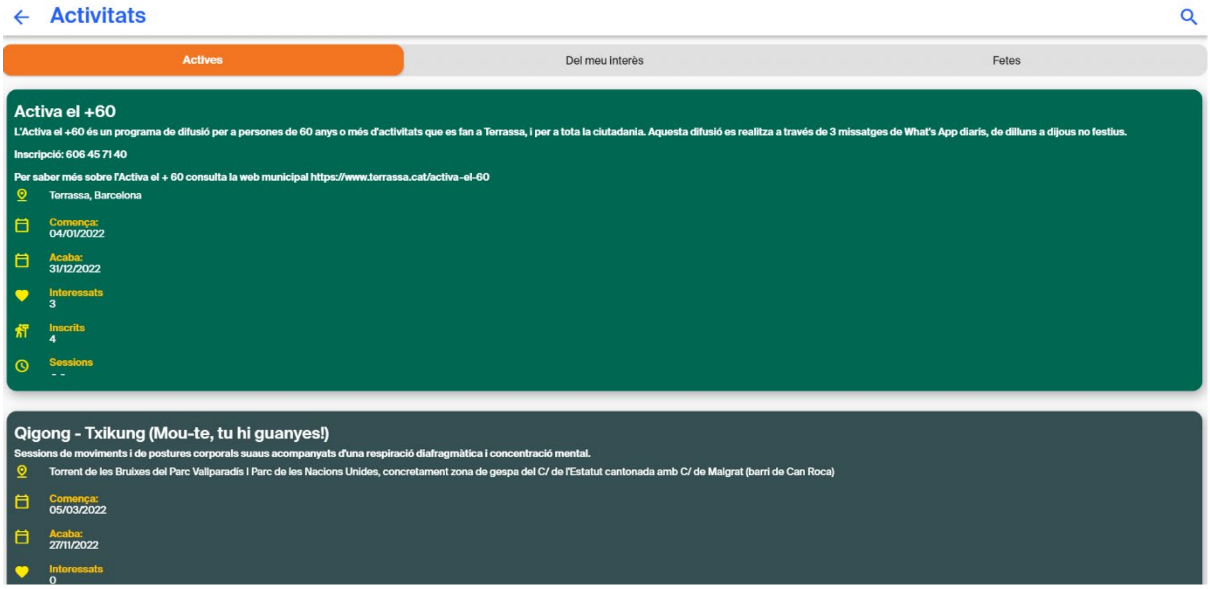
Figure 23. Frontend Web Registration



Avui et sents sol o sola ?

Gens (1) - Una mica (2) - Força (3), Molt (4) - Totalment (5)

Figure 24. Frontend Web Health Question



Activa el +60

L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activitats que es fan a Terrassa, i per a tota la ciutadania. Aquesta difusió es realitza a través de 3 missatges de What's App diaris, de dilluns a dijous no festius.

Inscripció: 606 45 71 40

Per saber més sobre l'Activa el + 60 consulta la web municipal https://www.terrassa.cat/activa-el-60

Terrassa, Barcelona

Comença:
04/01/2022

Acaba:
31/12/2022

Interessats
3

Inscrits
4

Sessions
- -

Qigong - Txikung (Mou-te, tu hi guanyes!)

Sessions de moviments i de postures corporals suaus acompanyats d'una respiració diafragmàtica i concentració mental.

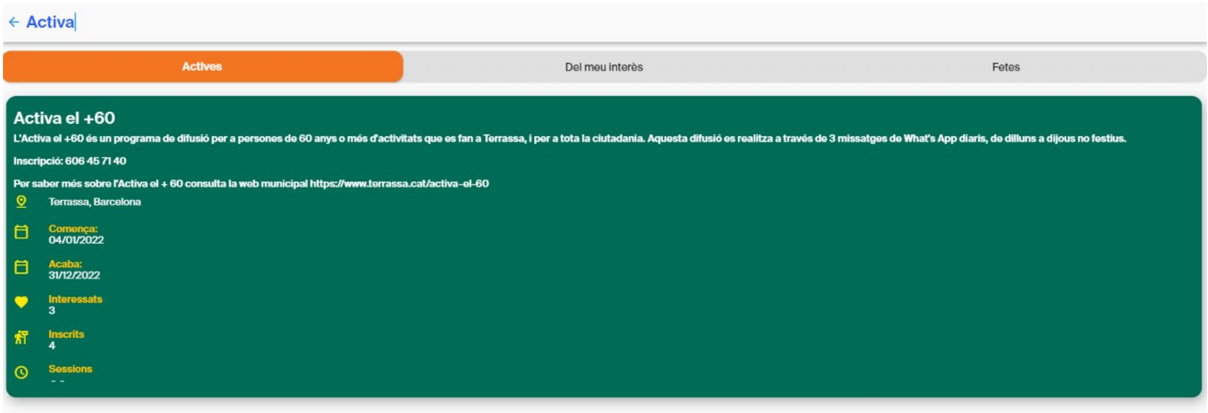
Torrent de les Bruixes del Parc Vallparadís I Parc de les Nacions Unides, concretament zona de gespa del C/ de l'Estatut cantonada amb C/ de Malgrat (barri de Can Roca)

Comença:
05/03/2022

Acaba:
27/11/2022

Interessats
0

Figure 25. Frontend Web Activities View



Activa el +60

L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activitats que es fan a Terrassa, i per a tota la ciutadania. Aquesta difusió es realitza a través de 3 missatges de What's App diaris, de dilluns a dijous no festius.

Inscripció: 606 45 71 40

Per saber més sobre l'Activa el + 60 consulta la web municipal https://www.terrassa.cat/activa-el-60

Terrassa, Barcelona

Comença:
04/01/2022

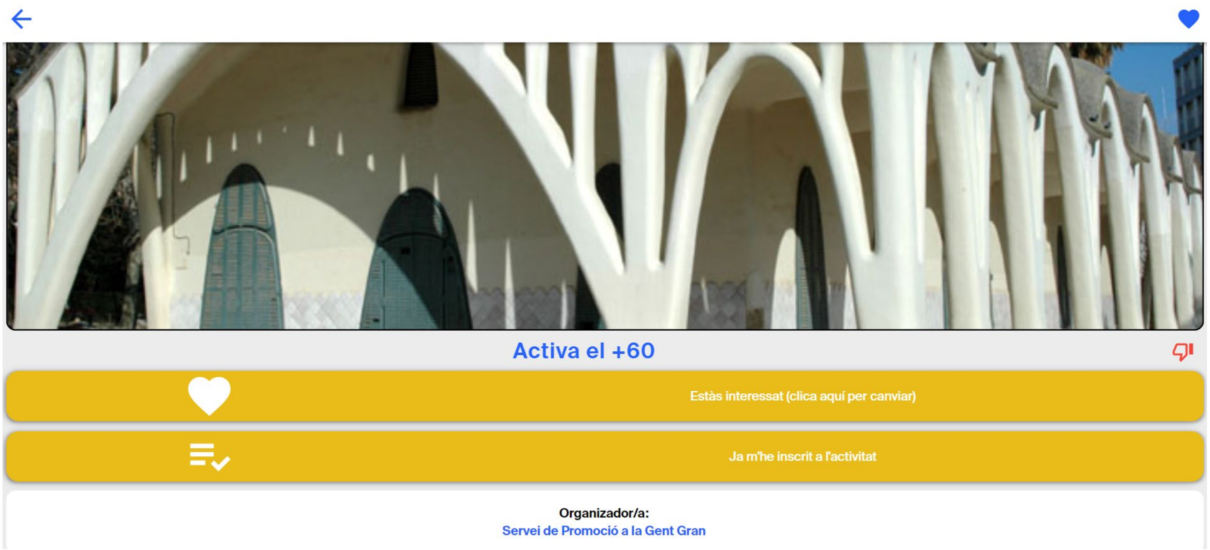
Acaba:
31/12/2022

Interessats
3

Inscrits
4

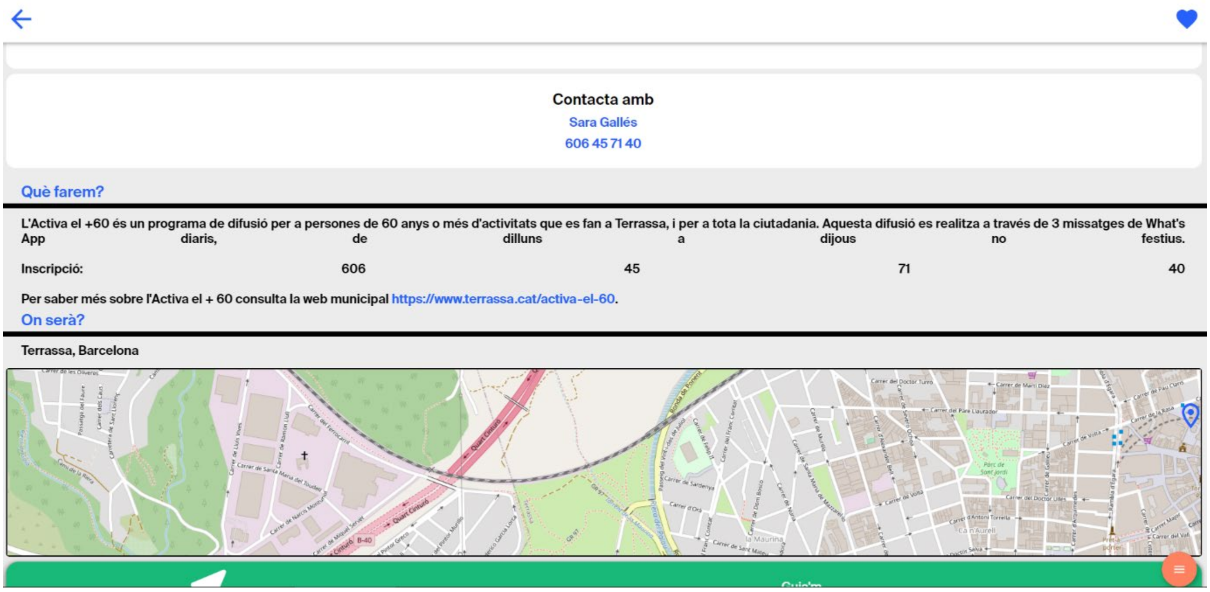
Sessions
- -

Figure 26. Frontend Web Activities Search


Activa el +60
Estàs interessat (clica aquí per canviar)
Ja m'he inscrit a l'activitat

Organizador/a:
Servei de Promoció a la Gent Gran

Figure 27. Frontend Web Activity in detail



Contacta amb
Sara Gallés
606 45 71 40

Què farem?

L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activitats que es fan a Terrassa, i per a tota la ciutadania. Aquesta difusió es realitza a través de 3 missatges de What's App diaris, de dilluns a dijous no festius.

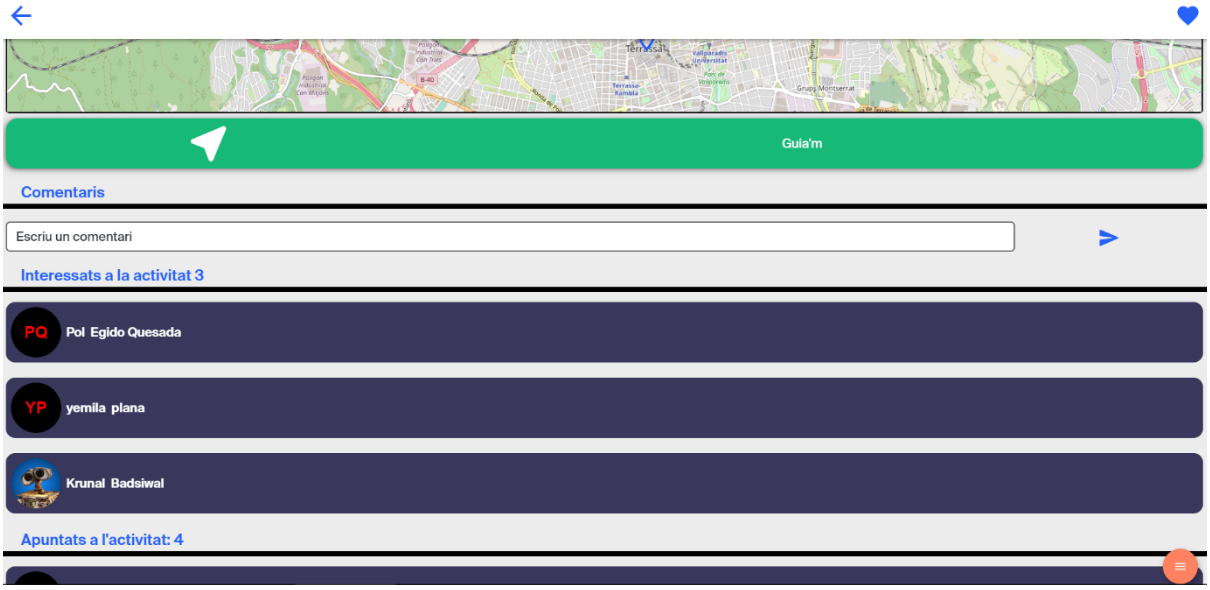
Inscripció: 606 45 71 40

Per saber més sobre l'Activa el + 60 consulta la web municipal https://www.terrassa.cat/activa-el-60.

On serà?

Terrassa, Barcelona

Figure 28. Frontend Web Activity view description, contact...



Comentaris

Escriu un comentari

Interessats a la activitat 3

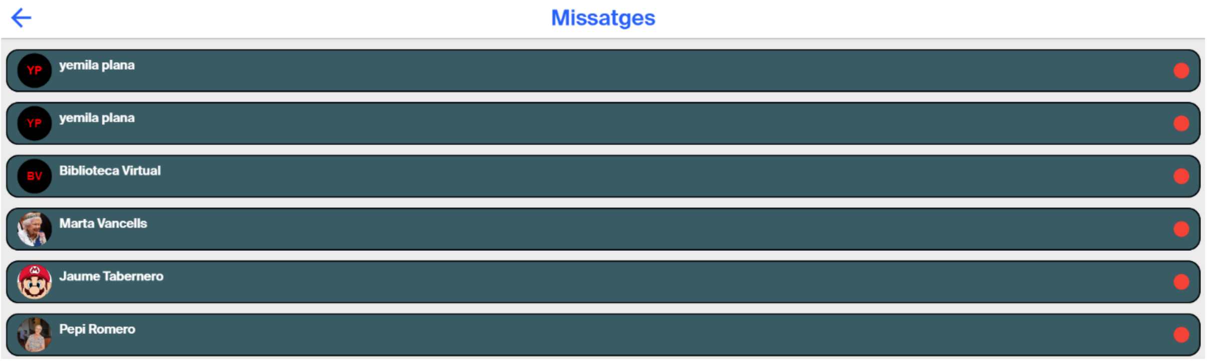
PQ Pol Egido Quesada

YP yemila plana

Krunal Badsiwal

Apuntats a l'activitat: 4

Figure 29. Frontend Web Activity View Followers and Enrollies



Missatges

YP yemila plana

YP yemila plana

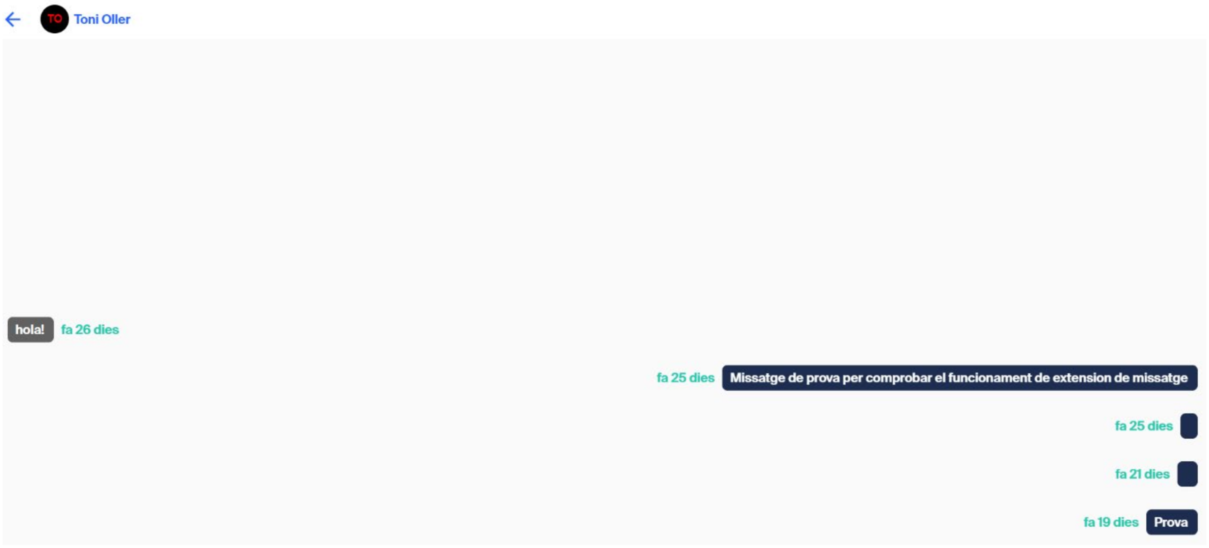
BV Biblioteca Virtual

Marta Vancells

Jaume Tabernero

Pepi Romero

Figure 30. Frontend Web Chat conversation's view



Toni Oller

hola! fa 26 dies

fa 25 dies Missatge de prova per comprobar el funcionament de extension de missatge

fa 25 dies

fa 21 dies

fa 19 dies Prova

Figure 31. Frontend Web Individual private conversation

Once the web was generated using flutter, the same code can be used without any modification for the mobile devices. Except the permission list.

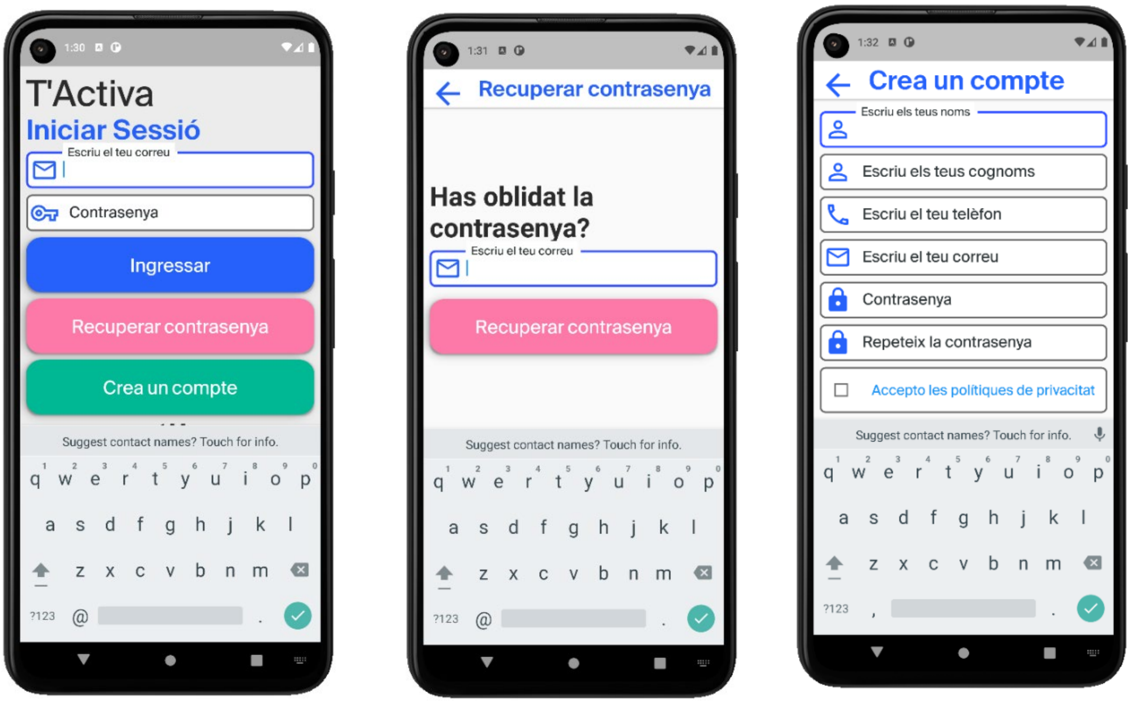
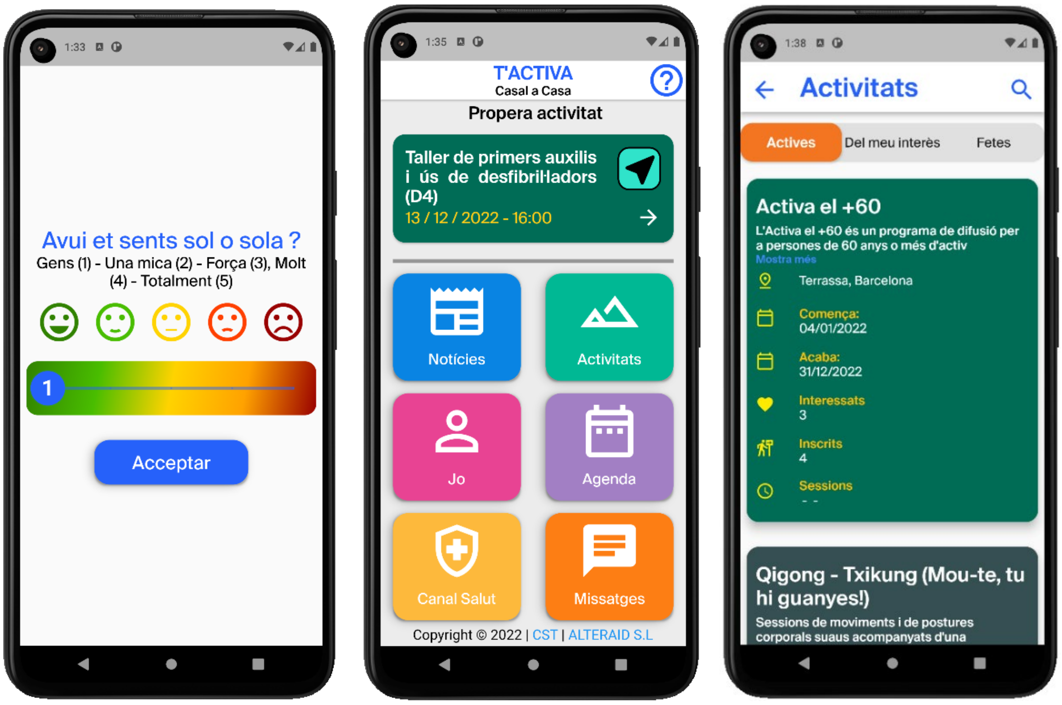


Figure 32. Frontend mobile Authentication, recover credential and registration



Avui et sents sol o sola ?

Gens (1) - Una mica (2) - Força (3), Molt (4) - Totalment (5)

Propera activitat

Taller de primers auxilis i ús de desfibril·ladors (D4)

13 / 12 / 2022 - 16:00

Copyright © 2022 | CST | ALTERAID S.L

Activa el +60

L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activ

Terrassa, Barcelona

Comença: 04/01/2022

Acaba: 31/12/2022

Interessats 3

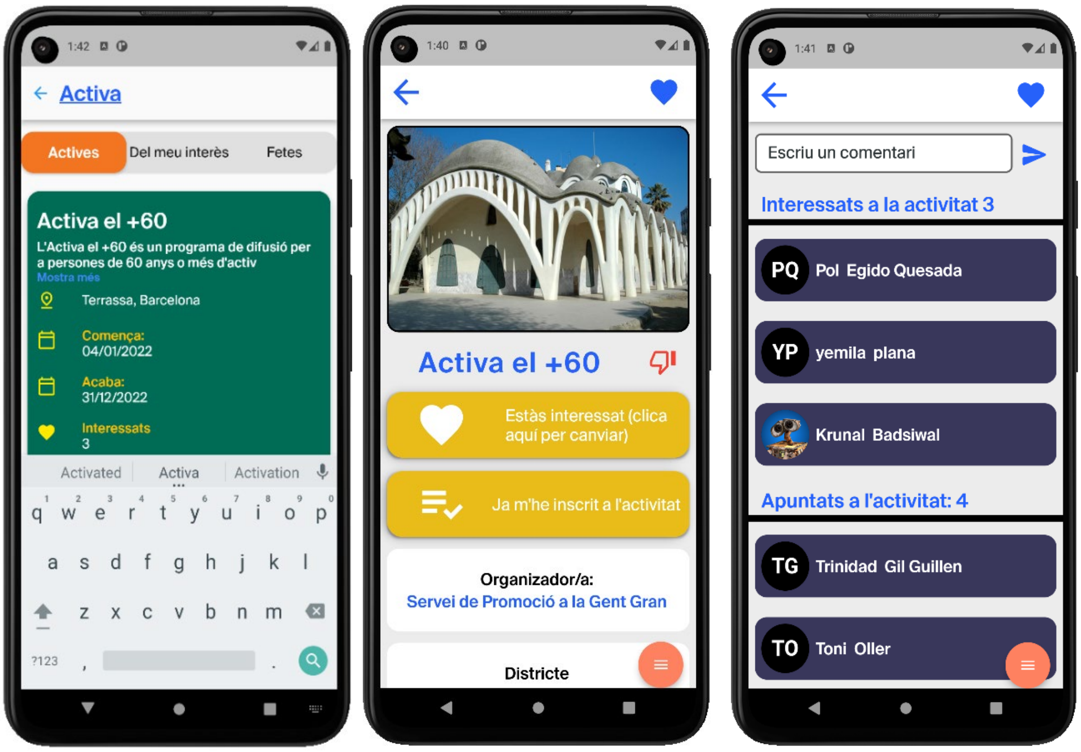
Inscrits 4

Sessions - -

Qigong - Txikung (Mou-te, tu hi guanyes!)

Sessions de moviments i de postures corporals suaus acompanyats d'una

Figure 33. Frontend mobile Health Question, Home, Activities View



Activa el +60

L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activ

Terrassa, Barcelona

Comença: 04/01/2022

Acaba: 31/12/2022

Interessats 3

Activa el +60

Estàs interessat (clica aquí per canviar)

Ja m'he inscrit a l'activitat

Organizador/a: Servei de Promoció a la Gent Gran

Districte

Escriu un comentari

Interessats a la activitat 3

PQ Pol Egido Quesada

YP yemila plana

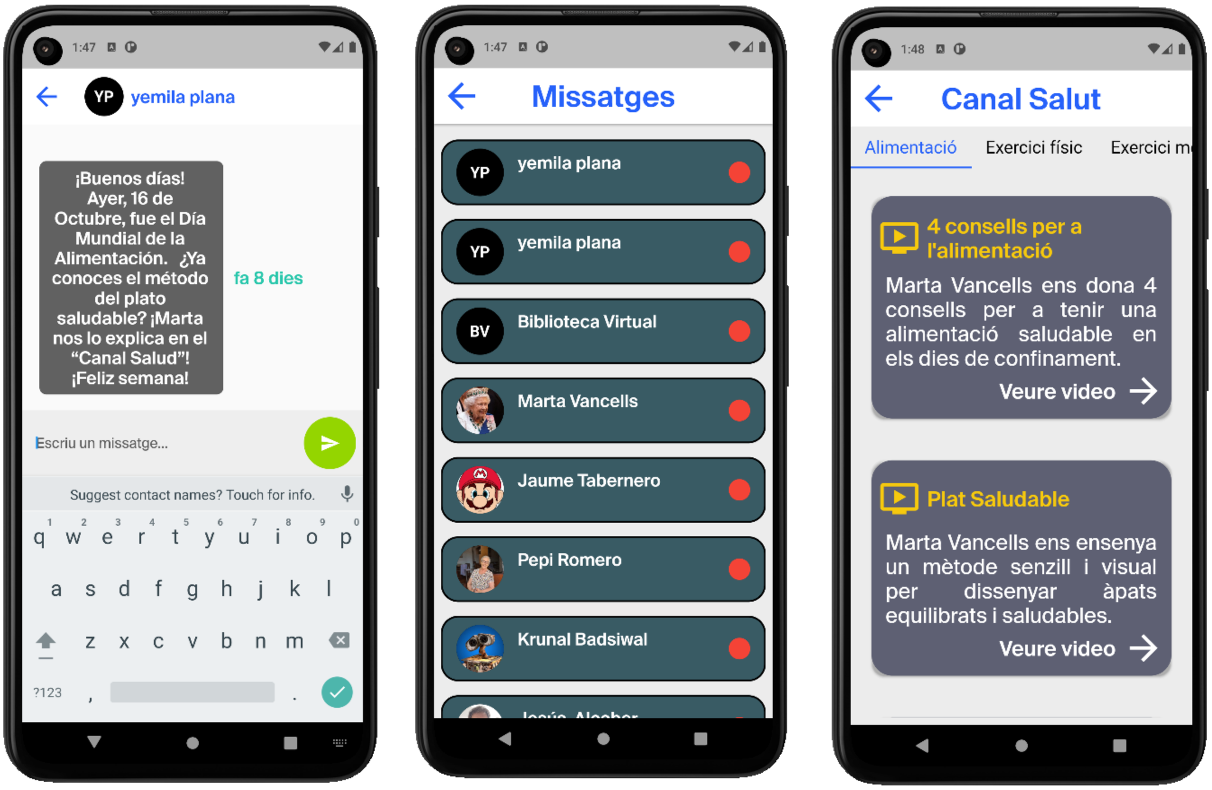
Krunal Badsiwal

Apuntats a l'activitat: 4

TG Trinidad Gil Guillen

TO Toni Oller

Figure 34. Frontend mobile Activity Search and Activity View



¡Buenos días! Ayer, 16 de Octubre, fue el Día Mundial de la Alimentación. ¿Ya conoces el método del plato saludable? ¡Marta nos lo explica en el "Canal Salud"! ¡Feliz semana!

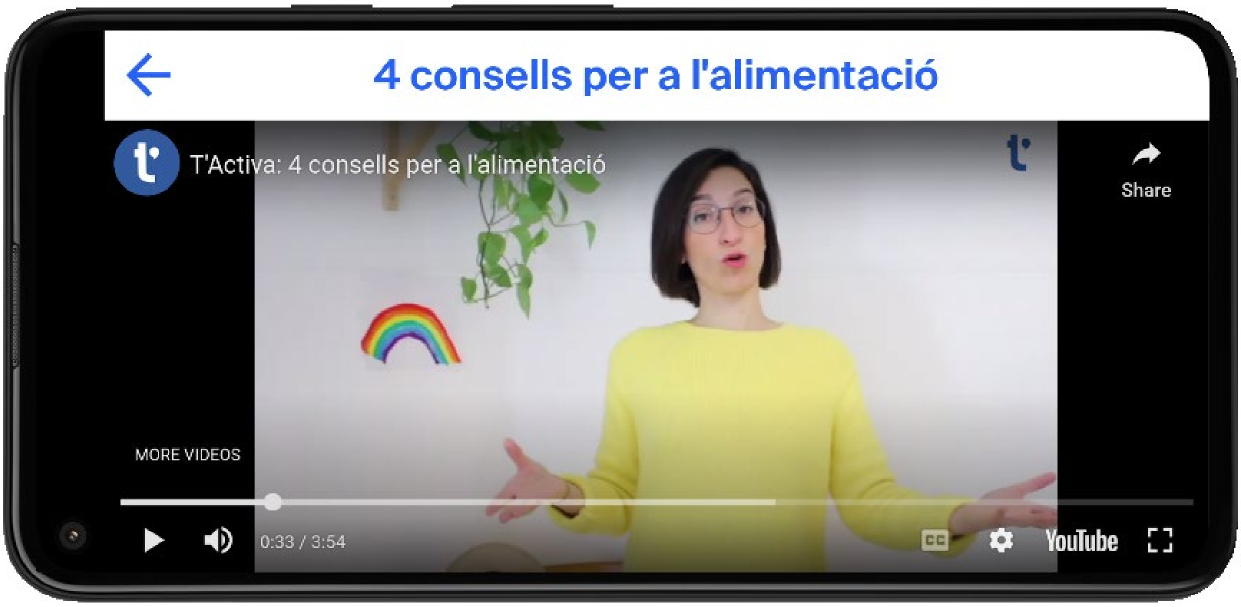
4 consells per a l'alimentació

Marta Vancells ens dona 4 consells per a tenir una alimentació saludable en els dies de confinament.

Plat Saludable

Marta Vancells ens ensenya un mètode senzill i visual per dissenyar àpats equilibrats i saludables.

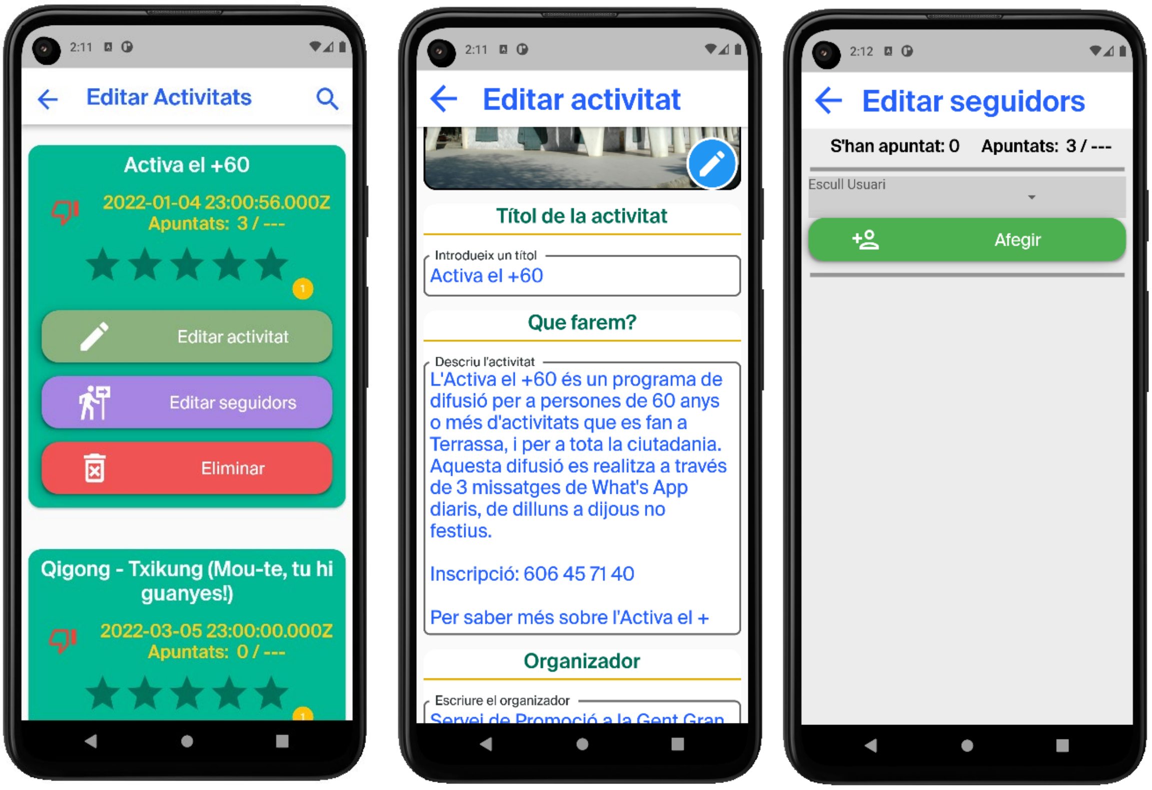
Figure 35. Frontend mobile Chat, Conversation and Health Channel



4 consells per a l'alimentació
T'Activa: 4 consells per a l'alimentació
Share
MORE VIDEOS
0:33 / 3:54
YouTube

Figure 36. Frontend mobile Health Videos

Until now, we have only implemented the views on the user side. But we had also defined use cases for controller. These views are implemented as follows



Editar Activitats

Activa el +60

2022-01-04 23:00:56.000Z

Apuntats: 3 / ---

Editar activitat

Editar seguidors

Eliminar

Qigong - Txikung (Mou-te, tu hi guanyes!)

2022-03-05 23:00:00.000Z

Apuntats: 0 / ---

Editar activitat

Títol de la activitat

Introdueix un títol: Activa el +60

Que farem?

Descriu l'activitat: L'Activa el +60 és un programa de difusió per a persones de 60 anys o més d'activitats que es fan a Terrassa, i per a tota la ciutadania. Aquesta difusió es realitza a través de 3 missatges de What's App diaris, de dilluns a dijous no festius.

Inscripció: 606 45 71 40

Per saber més sobre l'Activa el +

Organizador

Escriure el organizador: Servei de Promoció a la Gent Gran

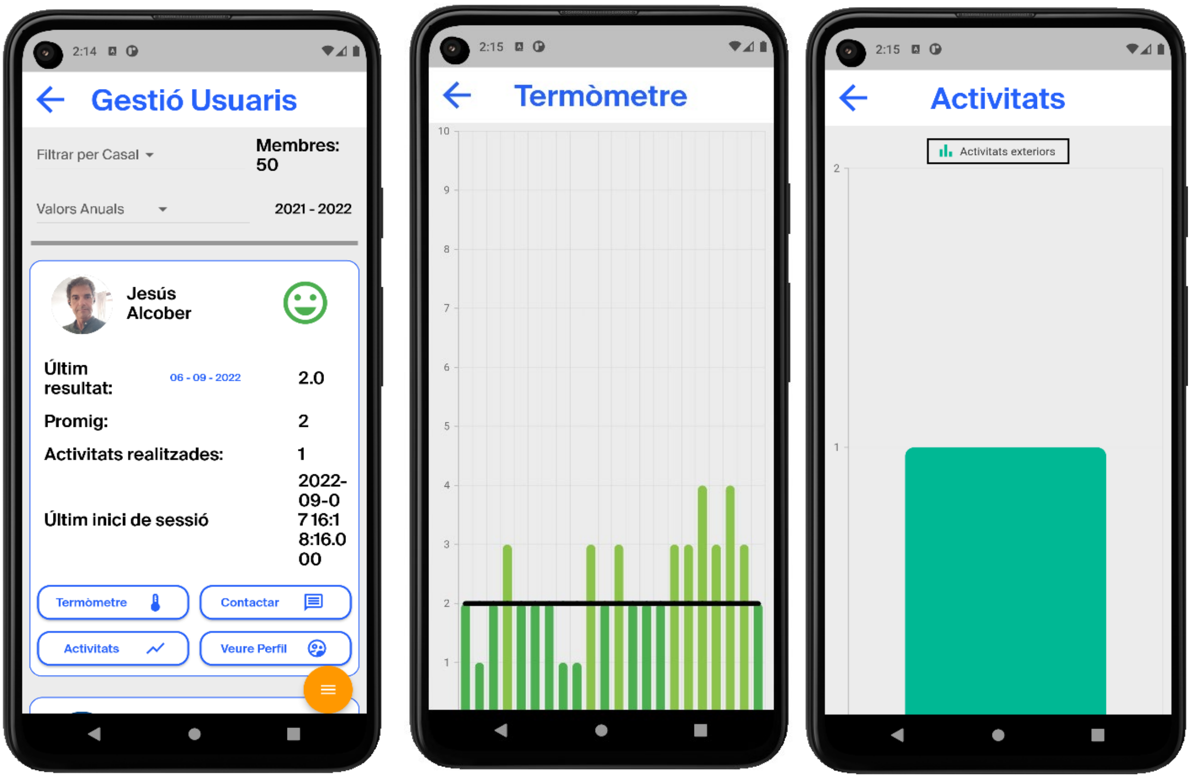
Editar seguidors

S'han apuntat: 0 Apuntats: 3 / ---

Escull Usuari

Afegir

Figure 38. Frontend mobile Controller Edit Activity



Gestió Usuaris

Filtrar per Casal

Membres: 50

Valors Anuals 2021 - 2022

Jesús Alcober

Últim resultat: 06 - 09 - 2022 2.0

Promig: 2

Activitats realitzades: 1

Últim inici de sessió 2022-09-07 16:18:16.000

Termòmetre

Contactar

Activitats

Veure Perfil

Figure 37. Frontend mobile Controller User Historical Data

We have successfully implemented all of the proposed use cases, proposed in the project proposal. For all of the planned scenarios and actors involved in the interaction within planned use cases.

CHAPTER 4. Future Appliances to Aerospace Industry

Technology is ushering in every part of life and connecting everything online to the Internet. This includes the aviation and aerospace, where recently many of the systems have been started to move towards a connected system. Since past few decades small continuous changes has moved aviation and aerospace from analogous systems to digital system, from hydraulics, avionics, etc.

This does not exclude the ground infrastructures, and this includes airports and humans. Past few years, everybody used to carry printed tickets in hand and had to look up the displays at the airport to check for flights. But nowadays everybody carries a portable handheld device which makes the access to all of the relevant information easy to access and keeps the user more up to date with news and information in regards to their current flights, etc.

4.1 Current state of universal application for airports

Presently each of the airports and airline has developed an application which is specifically used to serve themselves and does not allow any interconnection with any other.

Airports are already trying to make themselves smart replicating the conception of smart cities and trying to implement advanced technologies and improved methods of data collection to extract meaningful real-time insights.

As this will help improve operations and overall passenger experience. Which will improve the efficiency of the airport and the airlines itself and be less dependent on the airport notification system such as screen and the passengers can be more relaxed within the airport and maybe as such also increase sales and enjoy more.

IoT is making its way into the airport in forms of beacon and gateways which can become triggerable zones, improving passenger experience as allowing the climate control in the cabin to be automatically controlled using application, management of luggage via IoT beacons which tracks automatically all baggage’s.

All of this is currently possible only having installed specific application through the provided app support of the specific airline or airport.

4.2 Future applicability of the application

What this project proposes is a unified platform where the user is able to access the airport information, efficient travel routes from gate to gate, healthy and fun activities at the airport, IoT control of allowed device, interconnection and real-time data of the current state of the luggage, important notification and urgent news directly provided to the user mobile device in an accessible and universal approach.

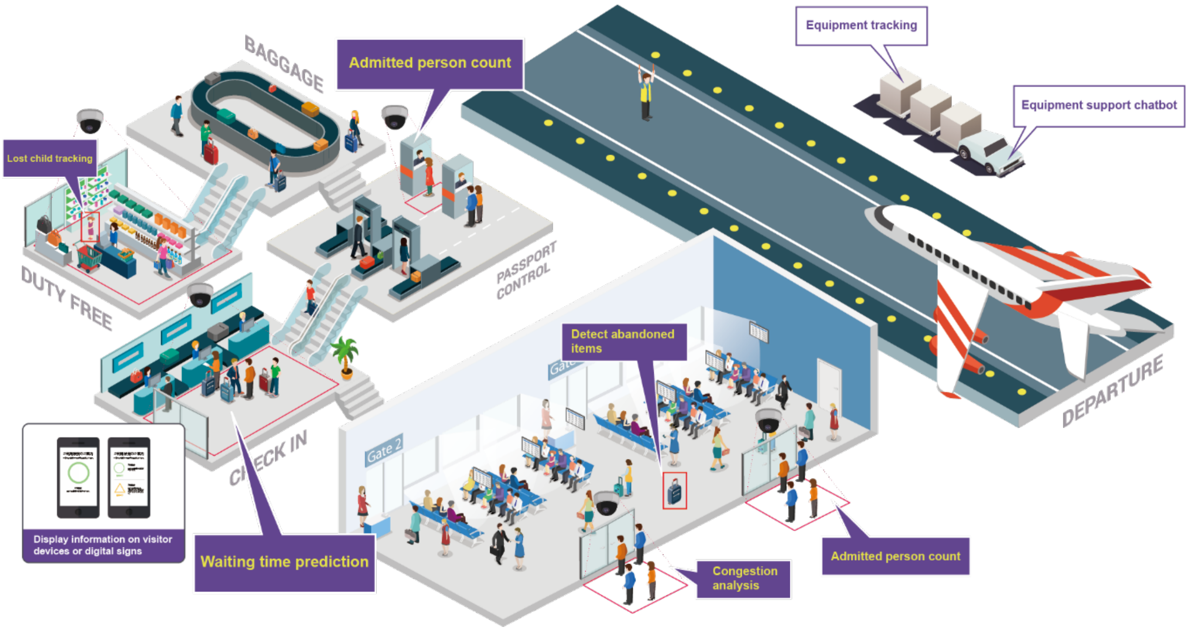


Figure 39. Future at airport with IoT (see reference [19])

The airport and other stores are also able to generate revenue from a smart airport with the integration of ads and publicity.

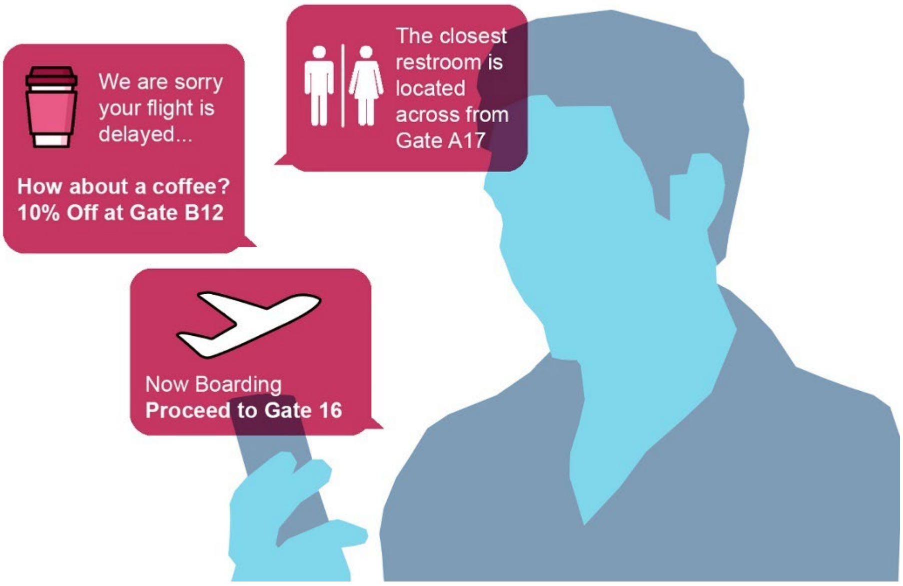

We are sorry your flight is delayed...
How about a coffee? 10% Off at Gate B12
The closest restroom is located across from Gate A17
Now Boarding Proceed to Gate 16

Figure 40. Smart Airport Ads and Publicity (see reference [21])

4.2.1 Feasible approach for the development

The current developed application only has few applications to the use cases at the airport. Such as ability to allow airports and shops at the airport itself to create activities which could be anything from trivial Yoga or short walking or running health-based activities to marketing stunt and publicity for the shop or brand themselves to have make customer more attached to the brand.

Most of the objectives and possible features are already possible.

4.2.1.a User Location determination

There are multiple ways to triangulate and obtain almost precise location within airport with just the use of existing technologies such as Wi-Fi of the airport or airplane.

This is possible with a possible usage of pings and device response time and multiple Wi-Fi stations. Which is currently available at all of the airports. Just an application-level software could track any device connected to the Airport Wi-Fi network with Trilateration.

Trilateration requires at least 3 Wi-Fi station with same AP to be accessible and visible on the handheld.

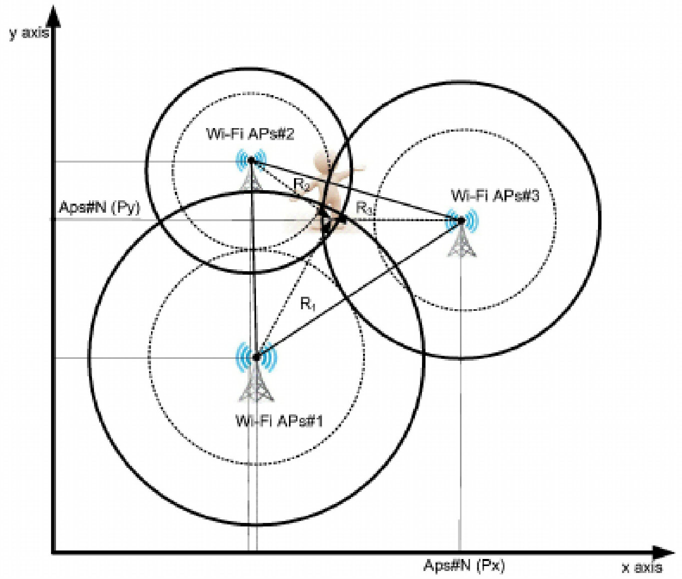


Figure 41. Trilateration Wi-Fi Technique for relative location (see reference [17])

The first methodology to obtain relative positioning of any person could be just with the usage of signal strength and a Hybrid model which would use Machine learning with the environment signal path loss and generate a relative location and movement path of the handheld device.

The second methodology to obtain the relative positioning of any person with a handheld device which could use protocol IEEE 802.11 and connect to the Access point would just require the use of ping and a historic modeled data of the location in the airport using real device. This is also known as Time-of-Flight (ToF).

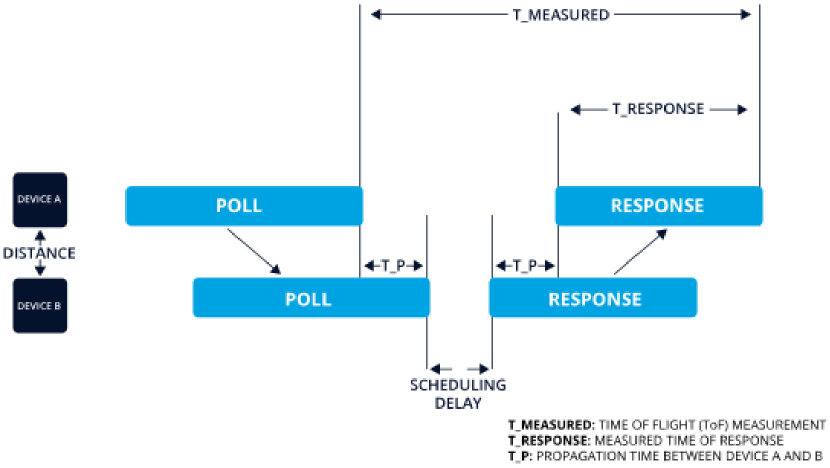

T_MEASURED
T_RESPONSE
DEVICE A
DISTANCE
DEVICE B
POLL
RESPONSE
T_P
POLL
RESPONSE
T_P
SCHEDULING DELAY
T_MEASURED: TIME OF FLIGHT (ToF) MEASUREMENT
T_RESPONSE: MEASURED TIME OF RESPONSE
T_P: PROPAGATION TIME BETWEEN DEVICE A AND B

Figure 42. Trilateration using ToF (see reference [18])

And just with the help of a data layer and a presentation layer which could make use of this relative position within airport would add so many advantages to follow and add value to user and the airport. Of course, the amount of computer power would be immense but with the aid of Artificial Intelligence, we could simplify the calculation lower the cost of implementation.

4.2.1.b RFID Integration with shared data layer

For airport to be able to offer customer/passengers any benefit to their baggage and other amenities throughout the airport easily and comfortably could be achieved with Radio Frequency Identification (RFID) for baggage tracking and which could be directly inserted in the data layer and traceable by both passengers and airlines from anywhere at any time without the requirement of any worries, as the data layer will be unified and have the same real-time updates.

There is a proposed solution defined as research paper from the School of Computer Science and Engineering, Beijing University of Aeronautics and Astronautics. Which proposes a simple topological solution for airport augmentation and adding this cost-effective method with a universal shared data layer between airlines and airports could make the whole operation effective and beneficial for airlines and airports. To track the baggage and find the faulting issue in the processing chain.

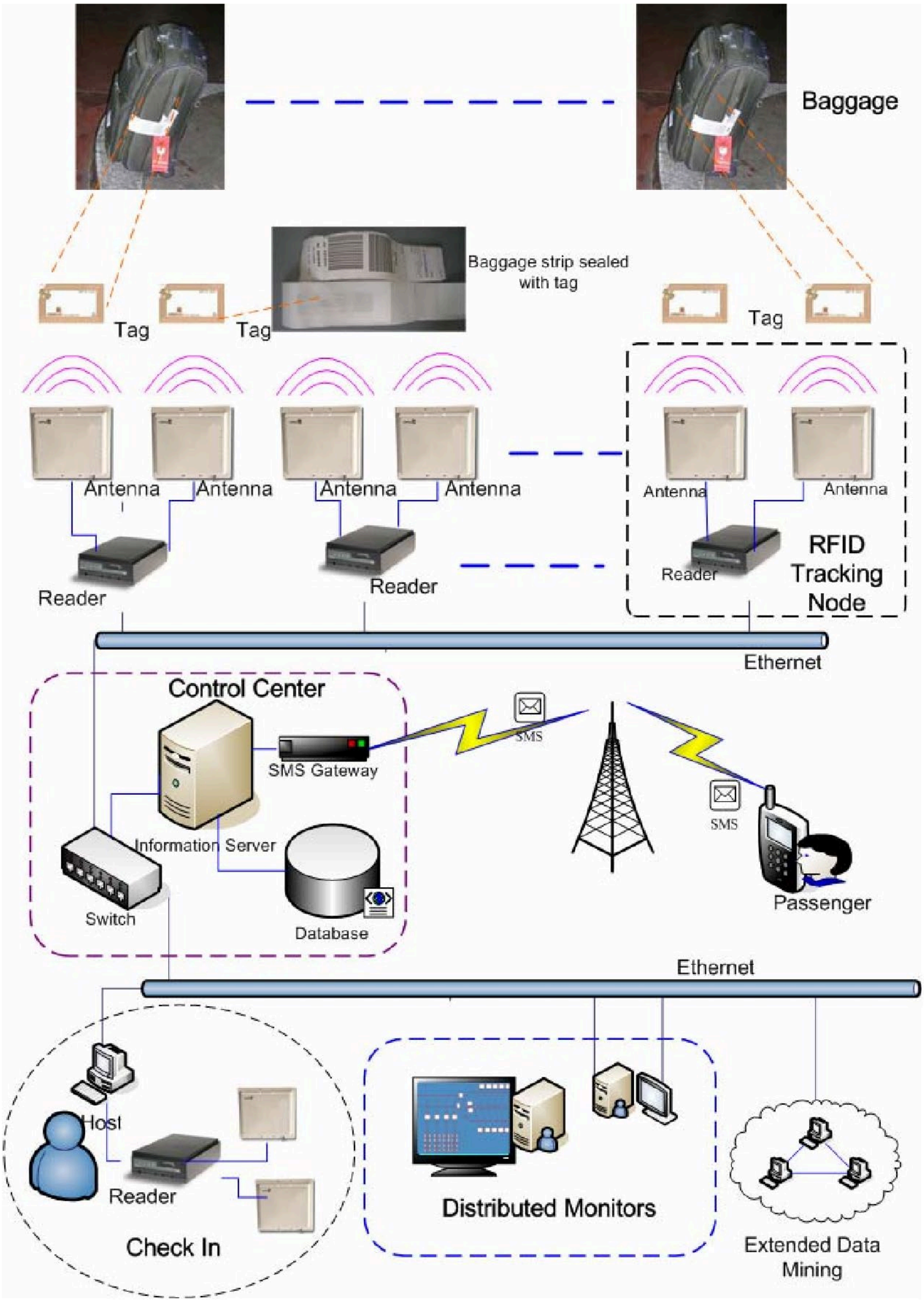


Figure 43. Baggage tracking throughout airport and airline topology (see reference [20])

CHAPTER 5. CONCLUSIONS

This project proposed a vast set of abstract requirements and goals for completion. All aiming towards improving people life and to be precise, the life of elderlies who had to go through covid without much help from outside world and put themselves in their shoes. Consorci Sanitari de Terrassa were the first to see this problem and propose a solution.

And our goal as humanity is to always try to make a better place for all of the people living in society. If one part of the society experiencing change and is moving toward a new future. It should also help other parts of the society to move toward the same future, even though it may require extra effort. In this project we proposed a solution to help the partially accessible people in our society to be adaptable to the new situation of remote meeting and remote activities and communication in a easier learning environment. Which is just what the Internet as a whole is all about. Even the goal of technology and engineering is to solve problems.

There are many different ways to address the problem and this happens to be just a single solution out of many. We could have even built a platform to teach and guide people to understand more about the internet and make them able to use the normal Internet. But we happened to prefer this solution more over the other, as this could also help us detect the change in their health and phycology on a daily basis and help in real life through other methods.

We even proposed an extension of the solution we built towards a better future in aerospace and aviation industry which is well within reach of making reality in few next years with very limited inversion or resource. And most of the changes required are all software and managerial on the global chain and not technology restrictive. As such we would like to see a future where a global interconnected aviation and aerospace industry future. And from the looks of it, it might not be much closer than I had Intentionally thought.

This project has given me an Insight into building complex project and working with customer, partially accessible people and businesses. And able to understand the business language and processes.

Also opened my eyes in the aviation industry and the rapid change that is ongoing internally in all parts of the aviation industry to optimize and make the process further and further efficient. This is first part to increase revenue and decrease losses but this increase also is beneficial to passengers, customer and all people working in these industries.

As such and such, I would like to be part of the future of aviation industry and help optimize and make the process more efficient in any way I can, If the opportunity arises.

REFERENCES

[1] StatCounter, "GlobalStats," GlobalStats, 09 2022. [Online]. Available: https://gs.statcounter.com/os-market-share/mobile/worldwide.

[2] StatCounter, "GlobalStats," GlobalStats, 09 2022. [Online]. Available: https://gs.statcounter.com/os-market-share.

[3] K. Sharma, "Tatvasoft," 25 December 2020. [Online]. Available: https://www.tatvasoft.com/blog/top-12-software-development-methodologies-and-its-advantages-disadvantages/.

[4] Alteriad, "Schema Model - Route Injector," Route Injector, [Online]. Available: https://routeinjector.github.io/docs-js/models/schema.html.

[5] Google, "Playstore Developer Terms and Services," [Online]. Available: https://support.google.com/googleplay/android-developer/answer/12766072?hl=en.

[6] Apple, "Appstore Terms and Services," [Online]. Available: https://developer.apple.com/app-store/review/guidelines/.

[7] J. F. Dooley, "Software Development, Design and Coding With Patterns, Debugging, Unit Testing, and Refactoring," in *Software Development, Design and Coding With Patterns, Debugging, Unit Testing, and Refactoring*, 2017.

[8] A.-N. Qadir, "Crowdbotics," 06 2021. [Online]. Available: https://www.crowdbotics.com/blog/best-practices-for-building-a-modularized-app.

[9] Editorial Team, "Synopsys," 28 March 2017. [Online]. Available: https://www.synopsys.com/blogs/software-security/top-4-software-development-methodologies/.

[10] Wikipedia, "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Use_case.

[11] Route Injector Step, “Github,” [Online]. Available https://routeinjector.github.io/docs-js/tutorial/step1.html

[12] Data model and Components, “IBM,” [Online]. Available https://www.ibm.com/docs/en/tivoli-netcoolimpact/6.1.1.4?topic=interface-managing-data-models

[13] Statista frameworks Consumer report, "Statista," [Online]. Available: https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/

[14] Ionic framework, "Ionic React," [Online]. Available: https://ionicframework.com/docs/react

[15] Cordova framework, "Cordova," [Online]. Available: https://cordova.apache.org/

[16] Backend data layer docs, "RouteInjector, Alteraid" [Online]. Available: https://routeinjector.github.io/docs-js/models/schema.html

[17] Torteeka, Peerapong & Chundi, Xiu & Dongkai, Yang. (2015). Hybrid technique for indoor positioning system based on Wi-Fi received signal strength indication. 48-57. 10.1109/IPIN.2014.7275467.

[18] Inpixion technology explaination, "Inpixion," [Online]. Available: https://www.inpixon.com/technology/standards/wifi

[19] OPTiM OS, "OPTiM Cloud IoT OS," [Online]. Available: https://en.optim.cloud/industries/airport/

[20] Zhang, Ting et al. “Traceable Air Baggage Handling System Based on RFID Tags in the Airport.” J. Theor. Appl. Electron. Commer. Res. 3 (2008): 106-115.

[21] Internet of Things in an Airport, “Gensler,” [Online]. Available: https://www.gensler.com/blog/iot-internet-of-things-in-the-airport

[22] Backend Solutions, “StatisticsAndData,” [Online]. Available: https://statisticsanddata.org/data/most-popular-backend-frameworks-2012-2022/

[23] Frontend Solutions, “Monocubed,” [Online]. Available: https://www.monocubed.com/blog/best-front-end-frameworks/

ANNEX

ANNEX 1. Use Cases, user stories

USE CASE 1: Login into Application
Primary Actor: User
Scope: Authentication to system
Level: System goal
Stakeholders and Interests:
User - wants to view his/her private space with activities, chats, news, etc.
System - wants to protect the data.
Precondition: User already has been registered/issued a set of credentials which includes username and password.
Minimal guarantee: sufficient information that only the specific user data is available to the actor.
Success guarantee: User can enter the homepage of the application and continue with other specific use cases.
Main success scenario:
1. Actor inputs the credentials.
2. The system authenticates and confirms the identity of the actor.
3. The system generates custom data to be sent over the internet to the actor device.
4. Actor is now able to see his/her homepage with his/her own information regarding his profile.

Extensions:
2a. Actor wants to login with incorrect credentials:
    2a1. System checks the credentials and fails the authentication.
3a. Credentials changed, which may be username and/or password
    3a1. System fails on authentication with old credentials

USE CASE 2: Registration into Application
Primary Actor: User
Scope: New identity to system
Level: System goal
Stakeholders and Interests:
User - wants to create a his/her private space with activities, chats, news, etc.
System - wants to protect the data.
Precondition: User does not have a preexisting credentials with existing email or phone Number.
Minimal guarantee: User created is unique, and does not already exist
Success guarantee: User profile is created and is sent to his/her profile confirmation screen.
Main success scenario:

1. Actor inputs the user data.
    a. Full Name
    b. Password
    c. Email
    d. Phone Number
    e. Accepts the terms and conditions

2. The system created a new user with the inputted data.
3. The system generates a code and sends the code to the user email provided in the input field.
4. Actor is partially registered to platform and need to complete the email confirmation step to be fully registered on the platform.

Extensions:

2a. Actor tries to login with new partially registered credentials:

2a1. System checks the credentials and send to confirmation email use case.

3a. Actor tries to maliciously enter the homepage and other screen

3a1. System fails on generate data and returns the user to confirmation email use case.

USE CASE 3: Confirmation of the partially registered user

Primary Actor: User

Scope: Confirmation of the identity by email

Level: System goal

Stakeholders and Interests:

User - wants to confirm his/her identity on the platform to be able to access his/her private space with activities, chats, news, etc.

System - wants to protect the data.

Precondition: User has a confirmation code with partially registered credentials.

Minimal guarantee: User confirmation code is not expired and email address is validated

Success guarantee: User is fully registered to the platform and is able to access his/her private space without any restrictions.

Main success scenario:

1. Actor inputs the confirmation code.
    a. Confirmation code

2. The system validates the user data with the confirmation code.
3. The system authenticates the user with full privileges.
4. Actor is fully registered to platform and is now able to access his/her private with activities, chats, news, etc.

Extensions:

2a. Actor tries to fully register with invalid confirmation code:

2a1. System checks the confirmation code and invalidates the try.

3a. Actor tries to fully register with expired confirmation code:

3a1. System checks the confirmation code and invalidates the try.

3a2. System generates a new confirmation code and sends it to the email provided when he/her partially registered to the platform.

USE CASE 4: User is able to navigate to activities

Primary Actor: User
Scope: View activities
Level: System goal
Stakeholders and Interests:
User – wants to view available activities.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views all available activities.
Main success scenario:

1. Actor performs action view activities
2. The system validates the request and sends the user with activities.
3. Actor is able to view all of the available activities

Extensions:

USE CASE 5: User is able to navigate to health information
Primary Actor: User
Scope: View health information
Level: System goal
Stakeholders and Interests:
User – wants to view available health information.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views all available health information.
Main success scenario:

1. Actor performs action view health information
2. The system validates the request and sends the user with health information.
3. Actor is able to view all of the available health information, which includes
    a. Textual Information
    b. Tweets
    c. Videos

Extensions:

USE CASE 6: User is able to navigate to messages
Primary Actor: User
Scope: View messages
Level: System goal
Stakeholders and Interests:
User – wants to view available messages.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views all available messages.
Main success scenario:

1. Actor performs action view messages
2. The system validates the request and sends the user with messages.
3. Actor is able to view all of the available messages from other users.

Extensions:

USE CASE 7: User is able to send message to another user
Primary Actor: User
Scope: Send message
Level: System goal
Stakeholders and Interests:
User – wants to view available activities.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views all existing ongoing conversations.
Main success scenario:

1. Actor performs action view conversation
2. The system validates the request and sends the user with the conversation including the messages.
3. Actor is able to view all of the past messages.
4. Actor is now performing an action of sending a text message to the current opened conversation
5. The system validates the conversation and the message and adds the message for another user.
6. The actor now sees the message as being sent with a confirmation notification of message being sent successfully

Extensions:
2a. The system fails the message dispatch on any error.

USE CASE 8: User is able to navigate to a single activity in detail
Primary Actor: User
Scope: View single activity in detail
Level: System goal
Stakeholders and Interests:
User – wants to view single activity in detail.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views the single activity in detail.
Main success scenario:

1. Actor performs action view activity from all of the activities
2. The system validates the request and sends the user with singular activity.
3. Actor is able to view all of the detail of the selected activity.
   This includes, current status of the user in relation to the activity as follows
   a. Follow status
   b. Enroll status to the activity
   Also includes other users' status, in relation to current activity
   c. Other user follow status to the activity
   d. Other user enroll status to the activity
4. Users are contactable through the activity detail.

Extensions:

USE CASE 9: User is able to navigate to a single activity in detail and contact a user from the list of users
Primary Actor: User

Scope: View single activity in detail and start a conversation with another user
Level: System goal
Stakeholders and Interests:
User – wants to view single activity in detail and start a conversation with another user
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views the single activity in detail and users available and creates a new conversation.
Main success scenario:

1. Use case 8 is performed with a success result.
2. The actor views the list of users.
3. The actor starts a new conversation with another user.
4. System checks if the conversation exists.
    a. If the conversation is nonexistent previously the system generates a new conversation and navigates the user to conversation.
    b. If the conversation exists previously the system redirects the user to the existing conversation.

Extensions:
2a. If another user is the same as the actor

- The system fails with a result of the actor being the same as the conversation user.

USE CASE 10: User is able to navigate to a single activity in detail and follow an activity
Primary Actor: User
Scope: View single activity in detail and follow/unfollow an activity
Level: System goal
Stakeholders and Interests:
User – wants to view single activity in detail and follow/unfollow that activity
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views the single activity in detail, follow the activity and add it to the calendar. On the contrary if the activity is already being follow, is unfollows the activity.
Main success scenario:

1. Use case 8 is performed with a success result.
2. The actor views the option to follow the activity
3. The actor follows the activity.
4. System checks the activity and validates user action.
    a. If the system detects, that the user already follows the activity, it will unfollow the activity and removes the actor from the user following the activity list.
    b. If the system detects, that the user does not already follow the activity, it will follow the activity and add the actor into user following the activity list in the activity detail.

Extensions:
2a. If the activity is in state different than active

- The system fails with a result of the activity being not active.

USE CASE 11: User is able to navigate to a single activity in detail and enroll/unenroll an activity
Primary Actor: User
Scope: View single activity in detail and enroll/unenroll an activity
Level: System goal
Stakeholders and Interests:
User – wants to view single activity in detail and enroll/unenroll that activity
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views the single activity in detail, enroll the activity. On the contrary if the activity is already being enrolled, is unenrolls the activity.
Main success scenario:

1. Use case 8 is performed with a success result.
2. The actor views the option to follow the activity
3. The actor enrolls the activity.
4. System checks the activity and validates user action.
   a. If the system detects, that the user already follows the activity, it will unenroll the activity and removes the actor from the user enrolled in the activity list.
   b. If the system detects, that the user does not already follow the activity, it will enroll into the activity and add the actor into user enrolled in the activity list in the activity detail.

Extensions:
2a. If the activity is in state different than active

- The system fails with a result of the activity being not active.

USE CASE 12: User is able to search through the activities
Primary Actor: User
Scope: Search activities
Level: System goal
Stakeholders and Interests:
User – wants to search through the available activities.
System - wants to protect the data.
Precondition: User is fully registered and Logged In.
Minimal guarantee: User remains authenticated and protected
Success guarantee: User views all activities results matching the search query.
Main success scenario:

1. Actor performs action view activities.
2. Actor performs search query.
3. The system validates the request and sends the user with activities which match the query.
   This query performed include activity title and description.
4. Actor is able to view all of the available activities with title and description matching the query performed through search.

Extensions:

2a. The list of activities can be empty if the search does not match nothing from the activities list.

USE CASE 13: System is able to distinguish between controllers and users
Primary Actor: System
Scope: Global
Level: System goal
Stakeholders and Interests:
Controller – wants to access and administer other users.
User – wants to access and view his/her own user data.
System - wants to protect the data.
Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs login.
2. The system validates the request and confirms that the user is a controller and allows the controller home screen to include extra actions which are now performable. This are defined in next use cases 14 and 15.

Extensions:
2a. The system returns normal home if the authenticated user is not a controller.

USE CASE 14: Controller wants to view and edit the activities
Primary Actor: Controller
Scope: View activities and edit activities
Level: System goal
Stakeholders and Interests:
Controller – wants to view available activities and edit them.
System - wants to protect the data.
Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs action view activities
2. The system sends activities in a different view.
3. The controller is able to open the activity and edit the activities.
4. The system validates the request and updates the activity.
5. Other users and controllers, see the updated activity.

Extensions:
2a. Any erroneous data change in the activity fails the modification.

USE CASE 15: Controller wants to view and delete an activity
Primary Actor: Controller
Scope: View activities and delete activities
Level: System goal
Stakeholders and Interests:
Controller – wants to view available activities and delete them.
System - wants to protect the data.

Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs action delete activity
2. The system sends activities in a different view.
3. The controller is able to open the activities and delete the activities.
4. The system validates the request and archives the activity and not deleting them.
5. Other users and controllers, see the activity deleted as inexistant through any activity view.

Extensions:

USE CASE 16: Controller wants to view and check normal user data
Primary Actor: Controller
Scope: View users and check history of activities and health results
Level: System goal
Stakeholders and Interests:
Controller – wants to view available activities and delete them.
System - wants to protect the data.
Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs action view users
2. The system navigates the controller to user's view
3. The controller is able to see health numbers and activities followed, enrolled of each user.
4. The controller is also able to start a conversation with a specific user.
5. The controller is able to see a graph of the historical activities enrollment and follow changes over period of times definable.

Extensions:

USE CASE 17: Controller wants to contact all of the users on the platform
Primary Actor: Controller
Scope: Contact all of the users on the platform together with same message
Level: System goal
Stakeholders and Interests:
Controller – wants to send a global message.
System - wants to protect the data.
Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs action view users
2. The system navigates the controller to user's view

3. The controller is able to see a option to send a message to all of the users available in the current view.
4. System navigates the controller to a fragmented view which allows a global textual message to be sent globally to all of the users.
5. The system creates a conversation if it previously did not exist and send the message.
6. The users are able to see a new conversation have been created with the controller privately.

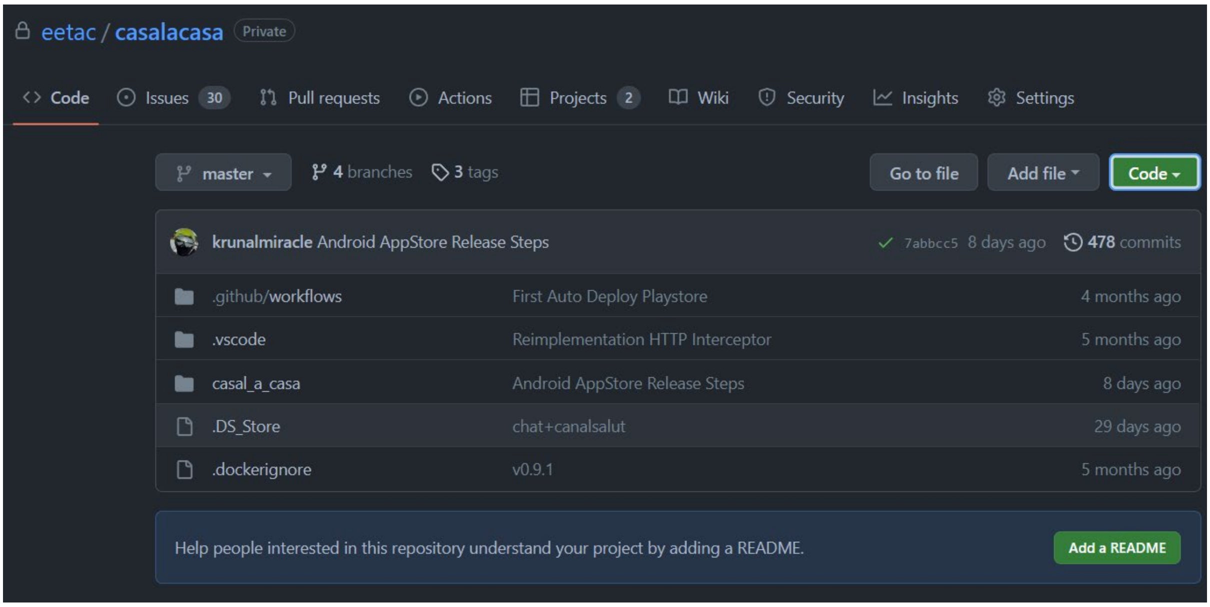
Extensions:

USE CASE 18: Controller wants to filter the users on the users view with a specific type known as organization assigned to each user on registration.
Primary Actor: Controller
Scope: Filter specific users connected to a specific organization.
Level: System goal
Stakeholders and Interests:
Controller – wants to filter specific users connected to a specific organization.
System - wants to protect the data.
Precondition: User is fully registered and Logged In, and is defined as controller
Minimal guarantee: User remains authenticated and protected
Success guarantee: System is able to perform extra use cases which are only available to controller when authenticated as a controller and not just normal user.
Main success scenario:

1. Actor performs action view users
2. The system navigates the controller to user's view
3. The controller is able to see an option to filter users based on organization.
4. System filters the list of users with just the users connected to the specific organization in the filter query.

ANNEX 2. Frontend or Presentation Layer Codebase repository

URL: https://github.com/eetac/casalacasa.git



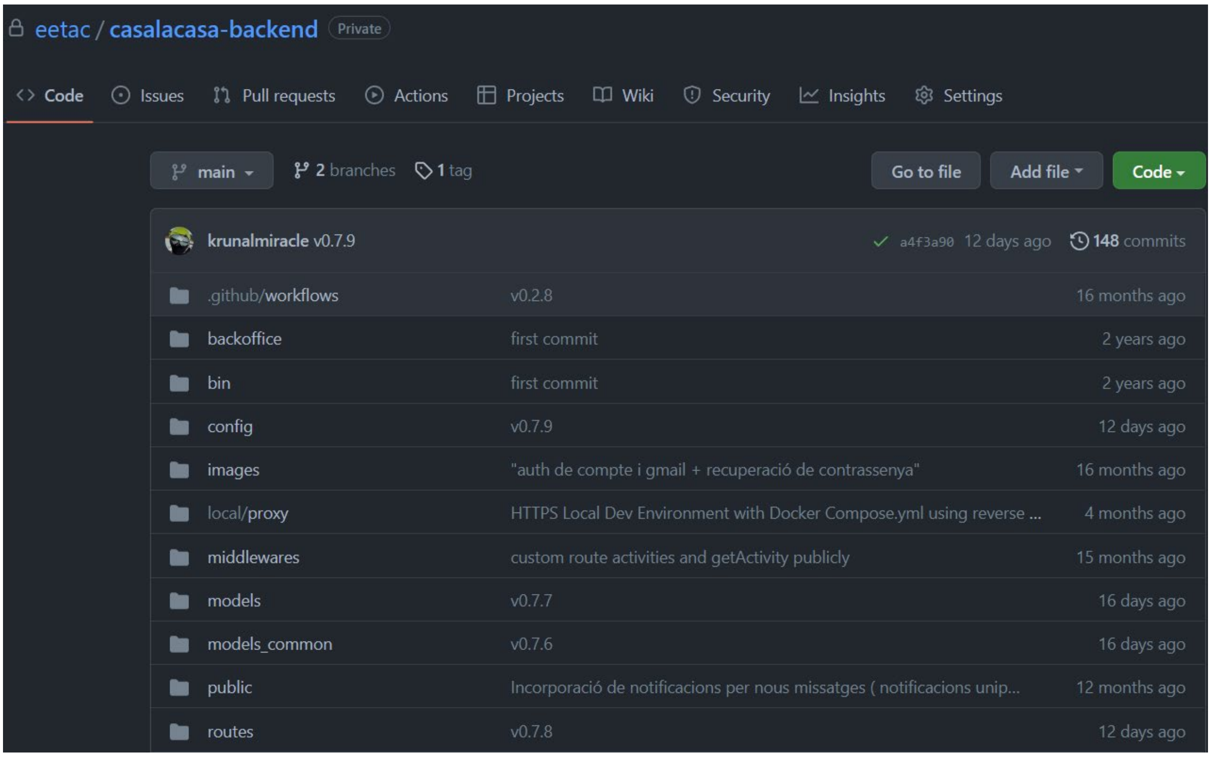
krunalmiracle Android AppStore Release Steps ✓ 7abbcc5 8 days ago 478 commits

.github/workflows	First Auto Deploy Playstore	4 months ago
.vscode	Reimplementation HTTP Interceptor	5 months ago
casal_a_casa	Android AppStore Release Steps	8 days ago
.DS_Store	chat+canalsalut	29 days ago
.dockerignore	v0.9.1	5 months ago

Help people interested in this repository understand your project by adding a README.

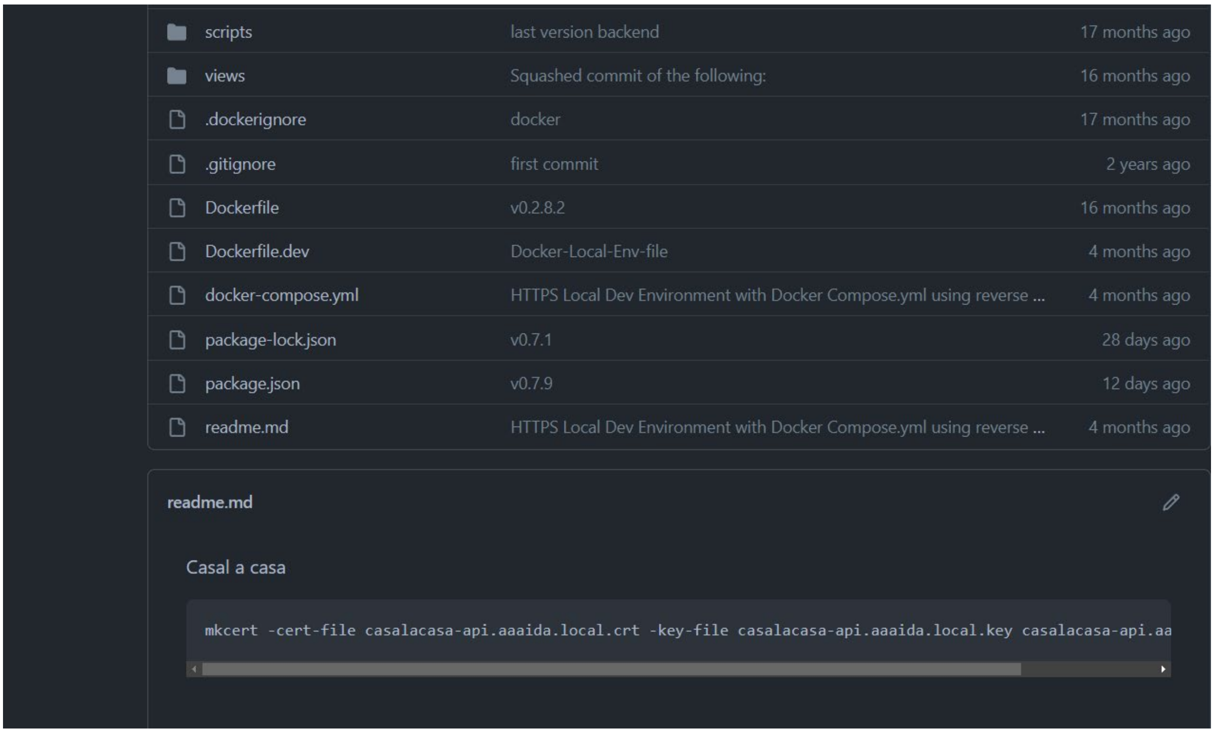
ANNEX 3. Backend or Data Layer Codebase repository

URL: https://github.com/eetac/casalacasa-backend.git



krunalmiracle v0.7.9 ✓ a4f3a90 12 days ago 148 commits

.github/workflows	v0.2.8	16 months ago
backoffice	first commit	2 years ago
bin	first commit	2 years ago
config	v0.7.9	12 days ago
images	"auth de compte i gmail + recuperació de contrassenya"	16 months ago
local/proxy	HTTPS Local Dev Environment with Docker Compose.yml using reverse ...	4 months ago
middlewares	custom route activities and getActivity publicly	15 months ago
models	v0.7.7	16 days ago
models_common	v0.7.6	16 days ago
public	Incorporació de notificacions per nous missatges (notificacions unip...	12 months ago
routes	v0.7.8	12 days ago



scripts	last version backend	17 months ago
views	Squashed commit of the following:	16 months ago
.dockerignore	docker	17 months ago
.gitignore	first commit	2 years ago
Dockerfile	v0.2.8.2	16 months ago
Dockerfile.dev	Docker-Local-Env-file	4 months ago
docker-compose.yml	HTTPS Local Dev Environment with Docker Compose.yml using reverse ...	4 months ago
package-lock.json	v0.7.1	28 days ago
package.json	v0.7.9	12 days ago
readme.md	HTTPS Local Dev Environment with Docker Compose.yml using reverse ...	4 months ago

readme.md

Casal a casa

```
mkcert -cert-file casalacasa-api.aaaida.local.crt -key-file casalacasa-api.aaaida.local.key casalacasa-api.aa
```