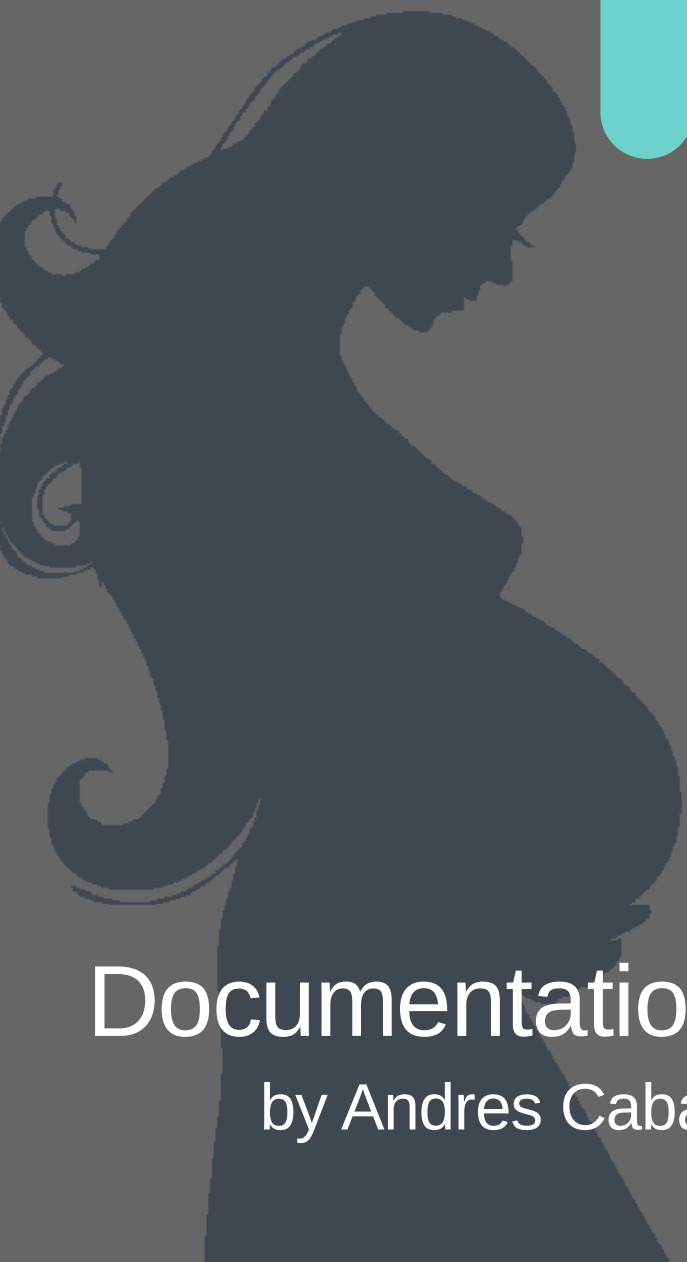
Information system sofware for H-integral



Documentation and architecture

by Andres Caballero and Alexis Erne

Design and implementation

Andres Caballero
Alexis Erne

SOMMAIRE

1. Introduction

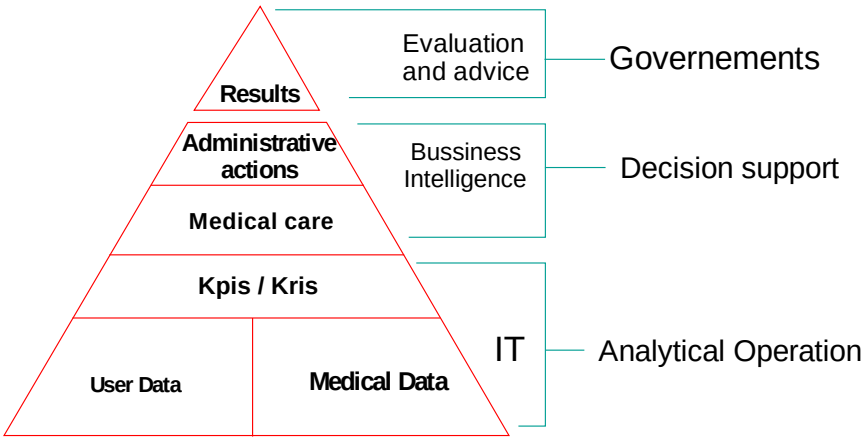
"H-Integral" is a commercial project that belongs to SUHMA POR TU SALUD, SAS de CV, this proposes to use new technologies of communication and information to improve the quality of medical care, as it is specified in the agreement, they asked us to make a software that can support the minimum requirements to run their concept of software (functional prototype). We will call this system "H-Integral software".

"H-Integral software" is a technological system that allows a self-assessment of health during pregnancy, delivery and postpartum in a comprehensive way.

It also allows the evaluation of the quality of care that patients receive in health clinics or hospitals. This is done to improve the quality of care received by mothers who are breastfeeding or who are pregnant.

To work properly, H-Integral uses an information system based on web technologies and "new SQL" databases. In addition, it also uses several internal data structures that allow a better treatment of multidimensional data.

To capture the data, H-integral uses a set of surveys that are carried out throughout the gestation process and during the first months of breastfeeding. these surveys are mainly aimed at patients and the medical staff that care for the patient. it also provides some useful microtransactions to simplify some administrative tasks, such as medical registration.



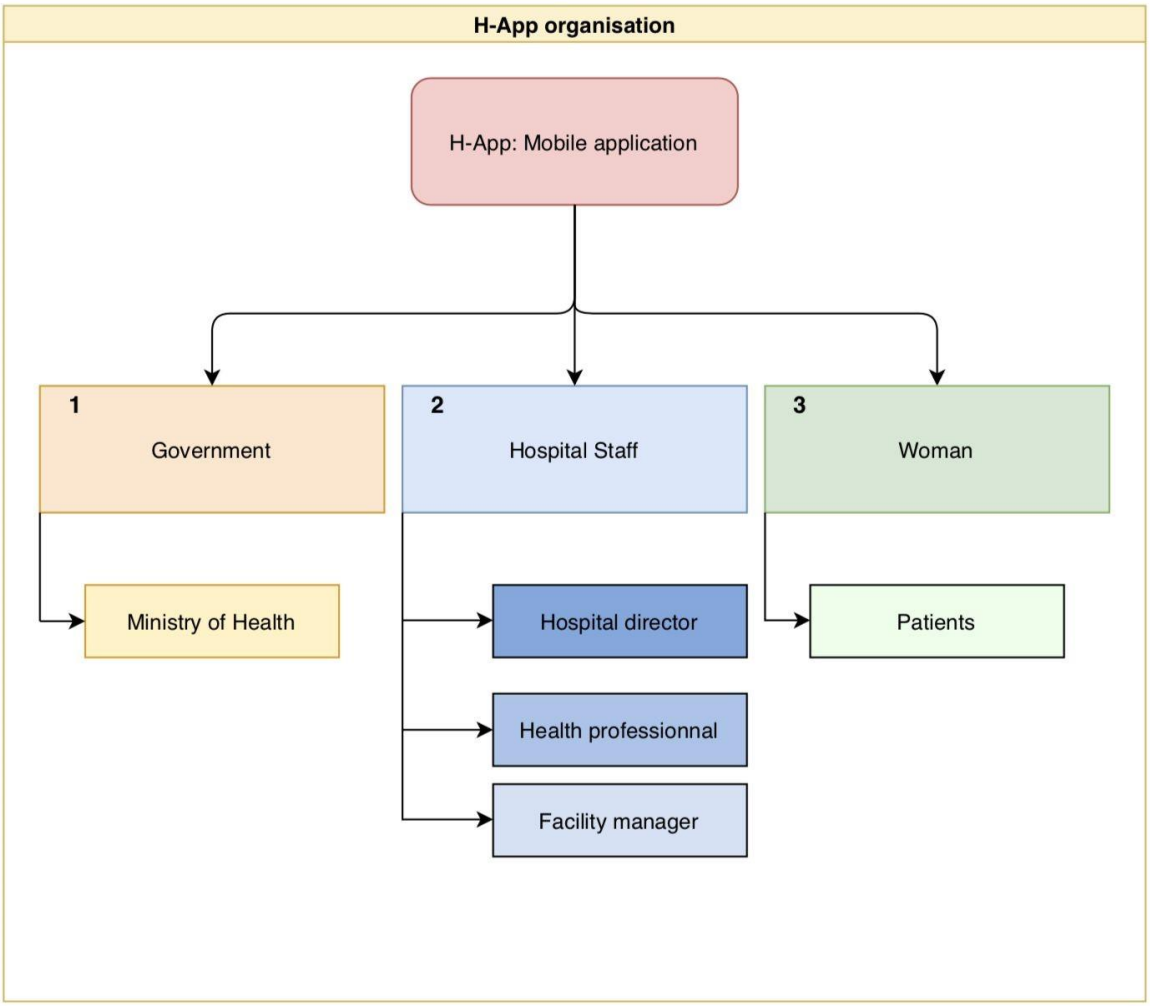
The information system used by H-Integral is divided into three parts, "***analytical operations"***, "***evaluation, and decision support for hospitals"*** and finally "***advice to governments and international organisations"***.

Analytical operations: this is the primary interest of the patient, using his or her data to measure the quality of care or to activate alerts in cases of risk.

Decision support: uses the Kpis/Kris generated from the patient data, to generate a report of global results that allows better decision making in health institutions, which contributes to an improvement in the quality of health.

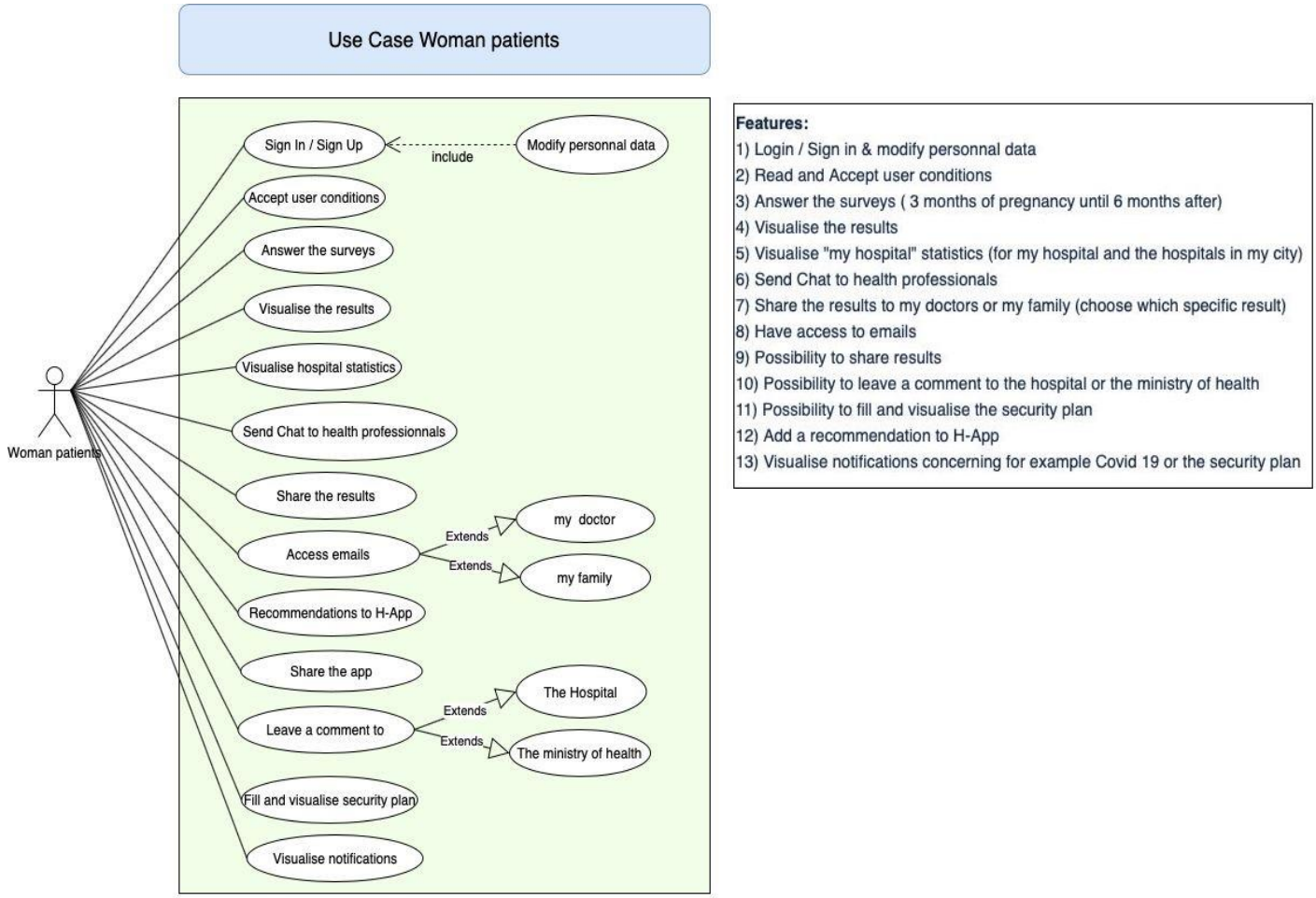
2. Analysis of user's requirements

The system should initially work for 3 types of users. (patients, Hospital staff, Governments)



For the initial version (functional prototype) only the operation of the patient user and the administrator is required.

Use Case woman patients: This use case represents every functionality that will have a woman patient.



HIGHLIGHTS:

1.Register

a. The system must allow a user to register using some method of identity validation.
b. The user must accept terms and conditions to participate (annex 10).
c. The user must enter at least his personal data and an e-mail address to register.

2. Take the survey.

a. The system must notify the time it has to respond.
b. The system must notify when the deadline for response is near.
c. The system must allow for the selection of a survey to be answered.
d. The system must allow for the answering of survey questions.

3. Consult indicators

a. The system must allow consultation of global indicators (Annex 6).
b. The system must allow consultation of personal results (Annex 11).

3. Architecture

Technology tree:

"H-Integral software" is completely developed in javascript, uses NodeJs on the server side and ReactJs on the client side. the union between the two is carried out harmoniously through NextJs, the data is saved in three different ways, SQL, NoSQL and "JSON native DataStructure", this according to its use and need. The client-side data processing is done using redux.

The requests to the server are made through two different methods, the first is webSokets for all transactions that do not include sensitive data and the second is through a REST: TCP / IP API, this allows the secure exchange between transactions.

Security:

the system implements 7 main security measures and 8 data protection strategies.

- SSL connection protocol
- Linear encryption in Redux database.
- Use of identification keys public/private in redux.
- Use of unique identification keys stored in cookies for session context.
- Use of the data context between the server and the client.
- Implementation of a system of unique sessions per window and browser.
- Hybrid encryption prototype combining the methods of “aes” and “rsa” encryption.

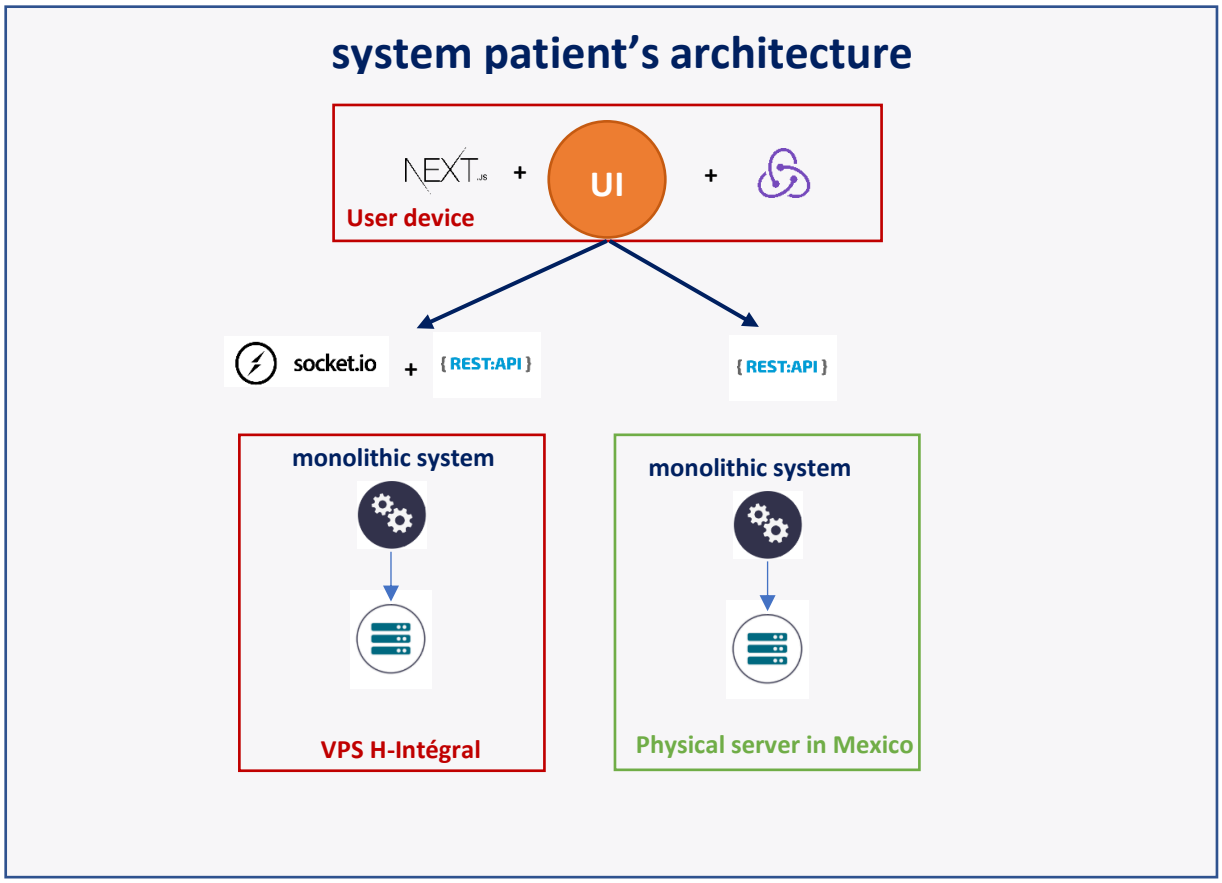
Data flow:

Initially the system was designed to exist in a monolithic system for Rest: API, in a traditional way. which implied a server, a database and a client ready to consume the API.

Even so, this solution is supported for a functional prototype where scalability is not expected, this because we wanted to finish as soon as possible and in the most economical way, however recently this part was modified, since we do not expect a functional prototype, but an operational software.

For this reason, two options are analyzed, the first one is a monolithic system for the woman client, which implies refactoring the client application, and a micro-services architecture for the management of the rest of the application.

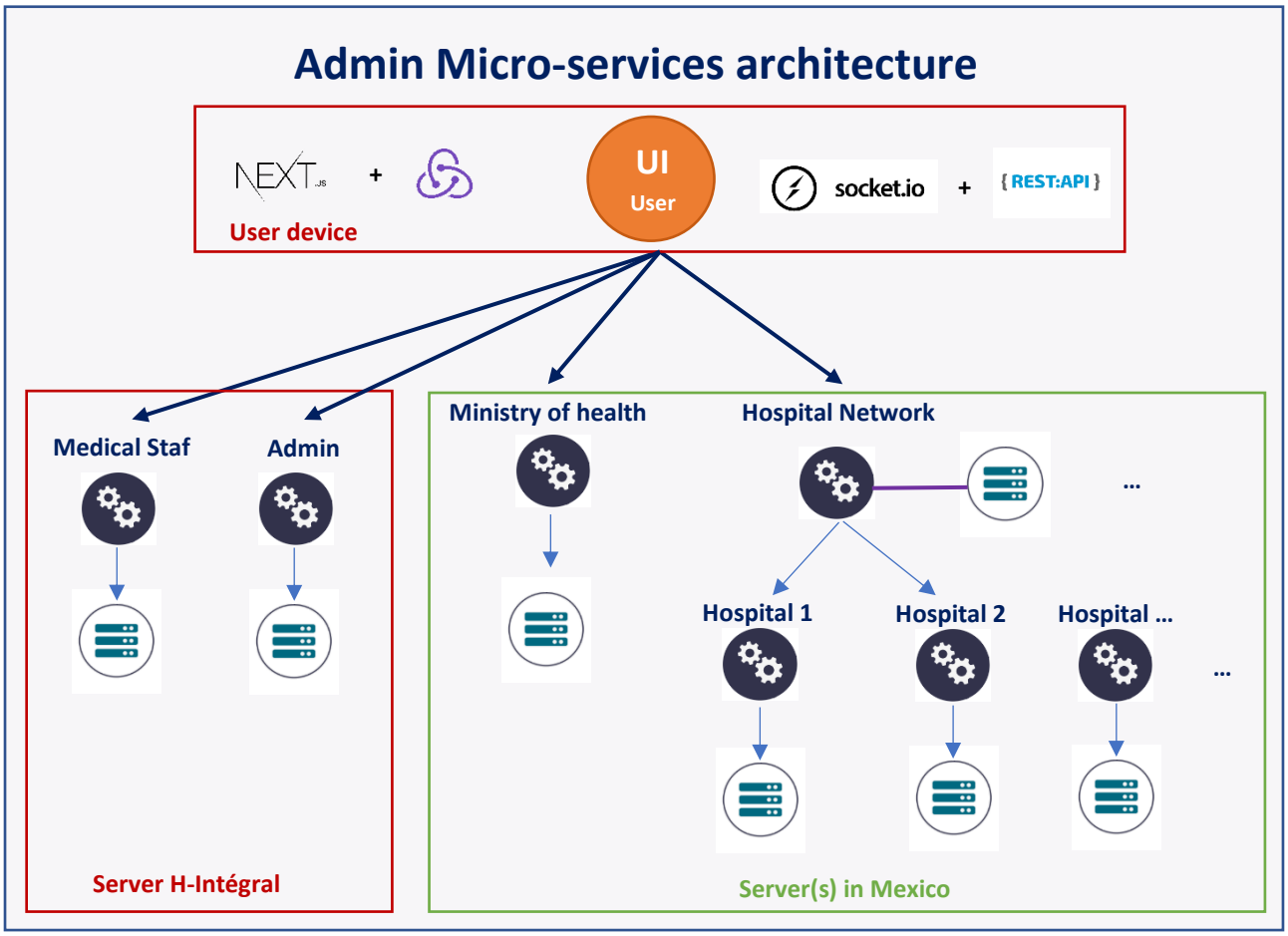
Thus, 2 different data flow architectures appear here.



The current architecture uses two independent servers, this with the objective of separating the users' data that will be stored in H-integral's private server and the sensitive medical data that belong to the health secretariat and must be in Mexico.

the user interface does part of the calculation work, but another part is done on the server, the user connects simultaneously to both servers to add or consult his information, while the application is open the user saves the local data of his session on his device using Redux, but these are deleted when he closes the browser.

the current system has great strengths due to its simplicity and ease of implementation, it is called monolithic, which means that it exists in a single non-reproducible state. however, if one part fails, the whole system is considered to have failed.



"The H-Integral software features microservices architectures, so we believe it is possible to integrate separate scalability based on its primary components.

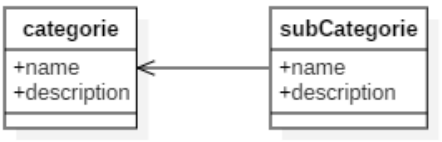
According to the minimum performance specifications, 2GB of ram, 20GB of disk and at least 2 cores of 1.8GHZ, this solution supports 1,500,000 users per service instance of each server, so the availability could be doubled by adding one core and 500MB of ram respectively, so the server can evolve using a cloud-based scaling.

Most of the computation will be done on the user side, which relaxes the server and allows it to better balance its routine tasks, the main advantage being to separate the overhead produced by triggers.

Since each microservice exists in a separate instance, if one part fails, the rest of the system will continue to function.

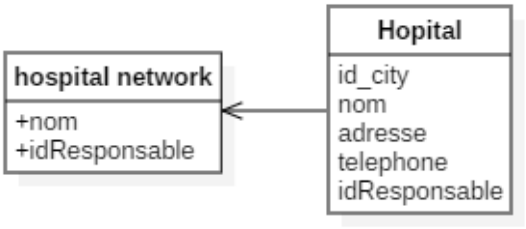
Since each microservice is an independent instance of the system, it can be developed autonomously by each developer without its progress or errors affecting the others, which allows for better evaluation of progress, but will make it difficult to share code or help with some tasks.

4. Data structure



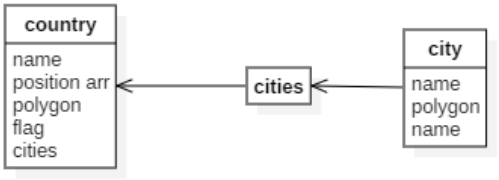
- The categories allow to classify the questions, these are composed of several subCategories, a name, and a description.

- The subCategories allow to classify the questions, these are composed of a name and a description.



- A hospital network is defined as a set of hospitals that share information and are represented by the same "responsible" user.

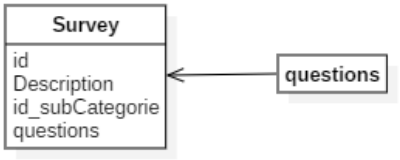
- A hospital represents any health entity where they can attend a user of the type "woman".



- A country represents a larger government entity, is made up of cities and has data related to their identification.

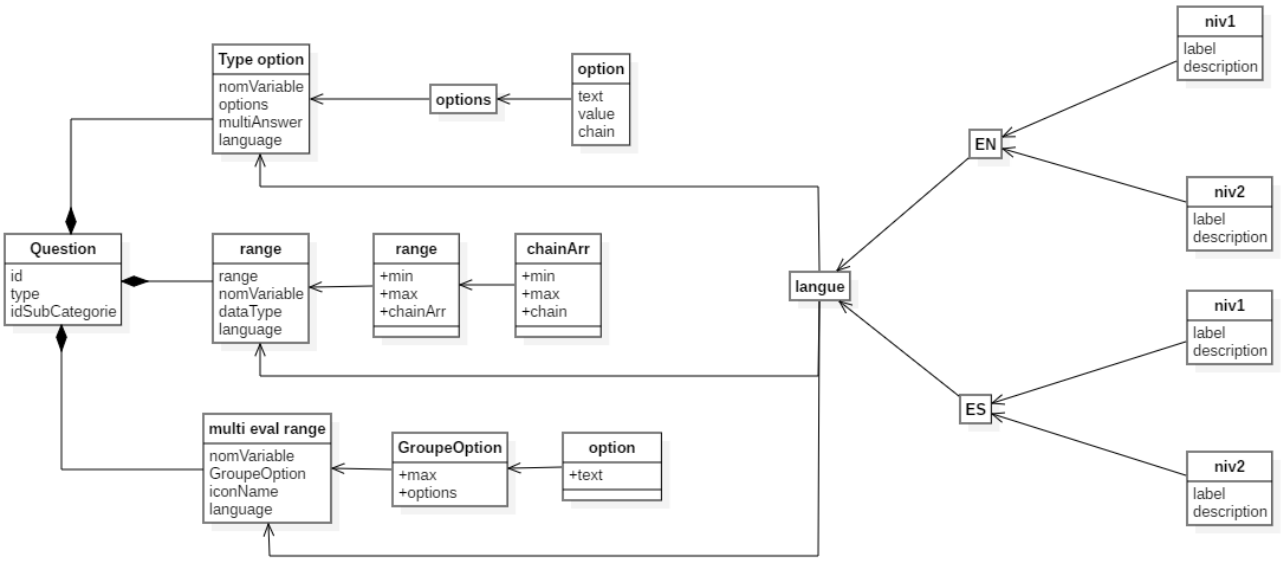
- Cities is an intermediary data structure that indexes the cities that belong to a country.

- City, represents a territorial entity, can be used for towns, sidewalks, villages, and any other type of designation that a Hospital may contain.



- A survey represents a defined set of questions that must be answered in a period of time and are triggered by a detonator.

- Questions is an intermediate data structure that indexes the identifiers of the questions contained in a questionnaire, in this way it does not store the questions, but rather their indexes.



A question is defined by a type and exists in an indexed form in an independent data structure and is not directly related to a questionnaire. The questions represent the elements of a personal questionnaire.

Each type of question includes a text in several languages and then in several educational levels.

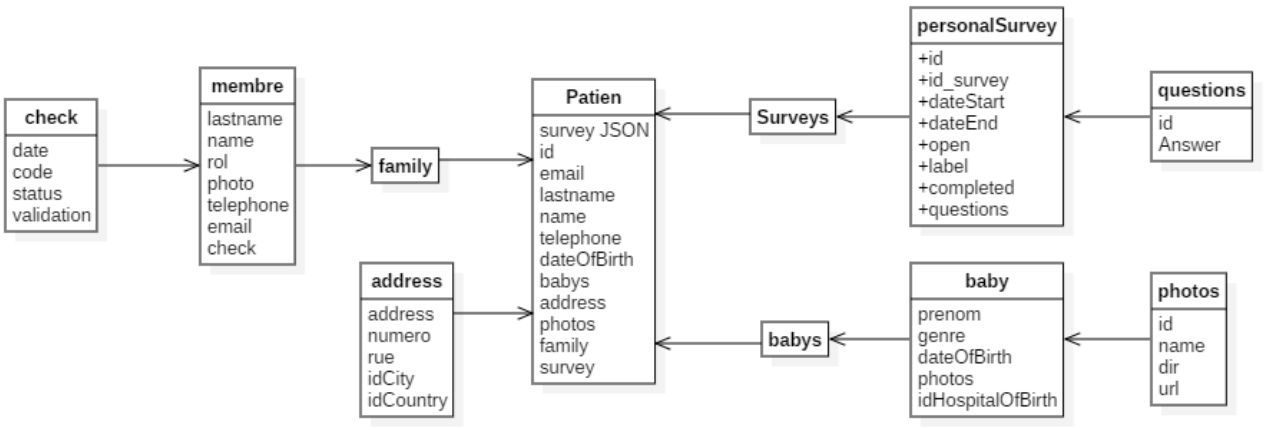
Por el momento existen 3 familias de preguntas,

Type option: puede ser de una unica respuesta o de selecion multiple, en cualquiera de los dos casos, se proponen al usuario un conjunto limitado de respuestas de entre las cuales el debe escojer.

Range: defines the questions within a number interval or a date.

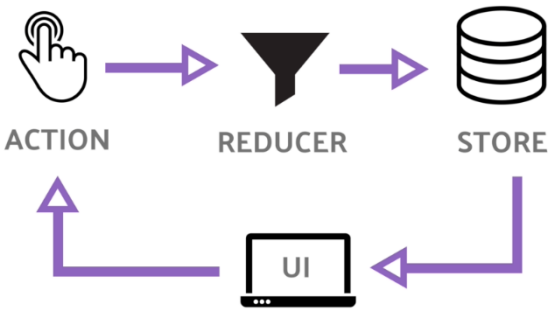
Multi eval range: defines table type questions, where the user receives several elements in a table that he must evaluate in a range between 0 and X, where X is the maximum number of

options. the options are always the same and can be defined with an icon or a word, for example, how do you feel, very good, good, normal, bad, very bad.



the data structure for the user of type patient does not change with respect to the initial model already implemented in the client version.

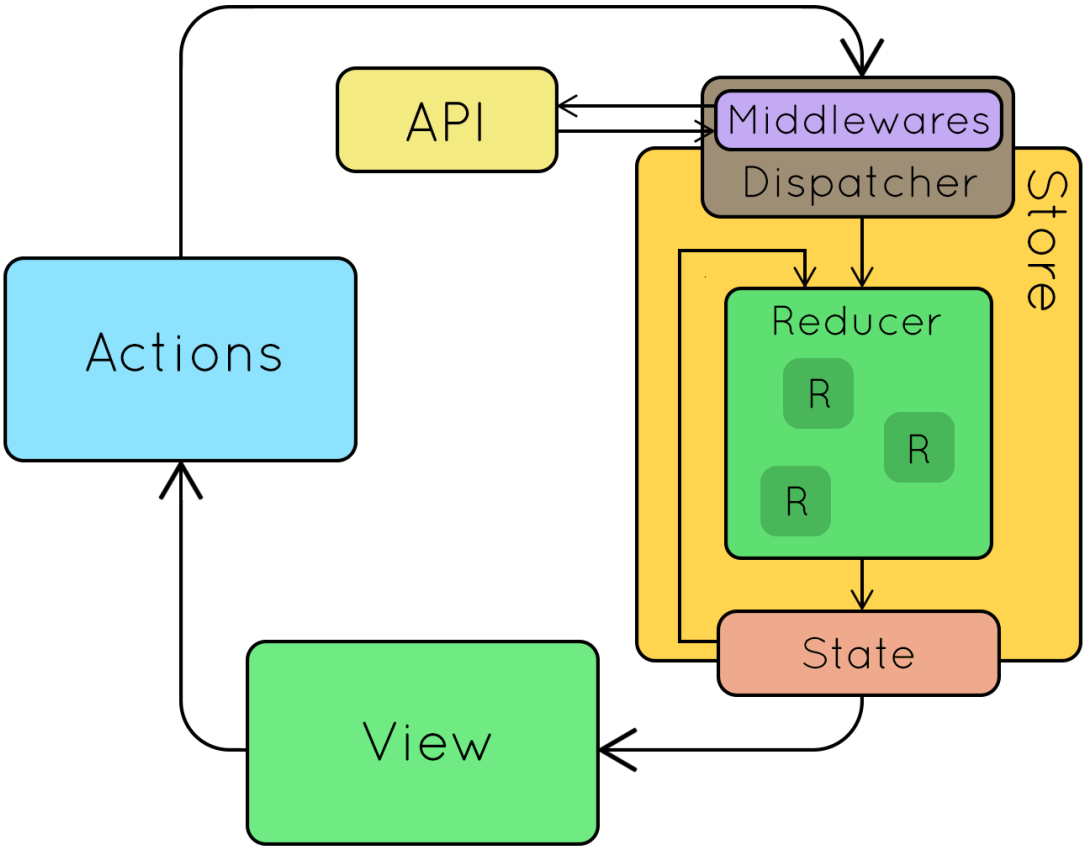
5. Data Flow



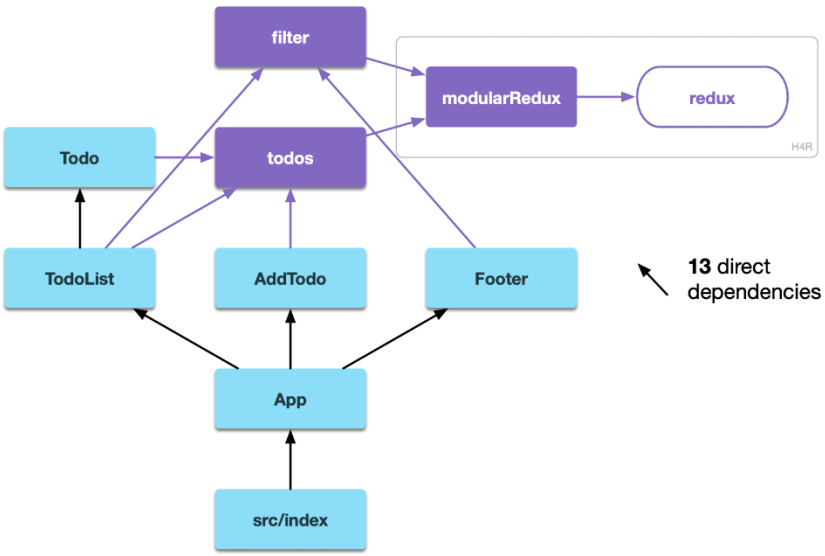
The client makes requests to the data loaded and stored locally in its device through Redux, as long as the data is not modified no request to the server is established although the socket connection is maintained in case the server wants to send something.

Every time the client modifies the data structure inside its user session that is stored inside the Redux store, an event will be reproduced in chain, and will end up sending a single request through WebSokect to the server to support the change on the server side, then the server will respond to the data swith and respectively Redux will notice the change and will generate a new render that will be effective only for the components involved in the data change, or that are subscribed to it.

Active update by detonators:



data propagation:



6. Triggers and events

The architecture has an independent microservice that manages the triggers and is fed by an event manager.

currently there are 5 events that are classified in 2 types,

Data Event:

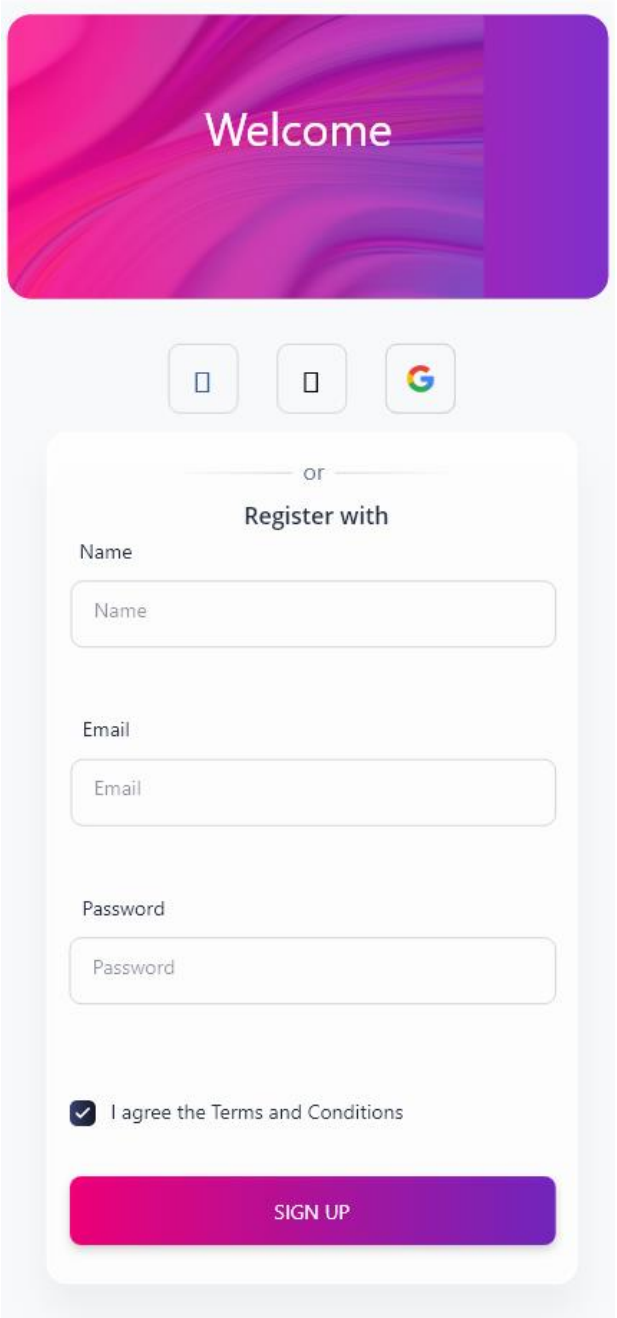
- when submitting a survey response.
- when a program is completed.
- when declaring a change in primary information.

Monitoring Event:

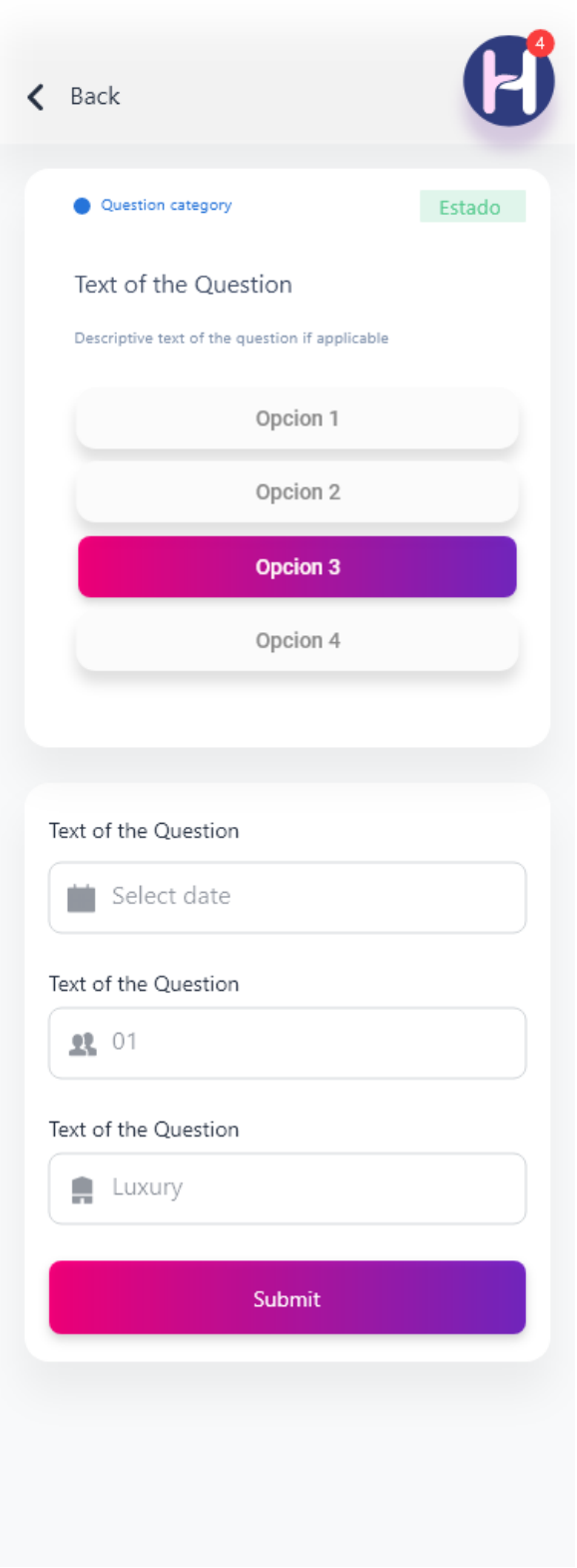
- when replying to a personal message.
- when viewing a notification.

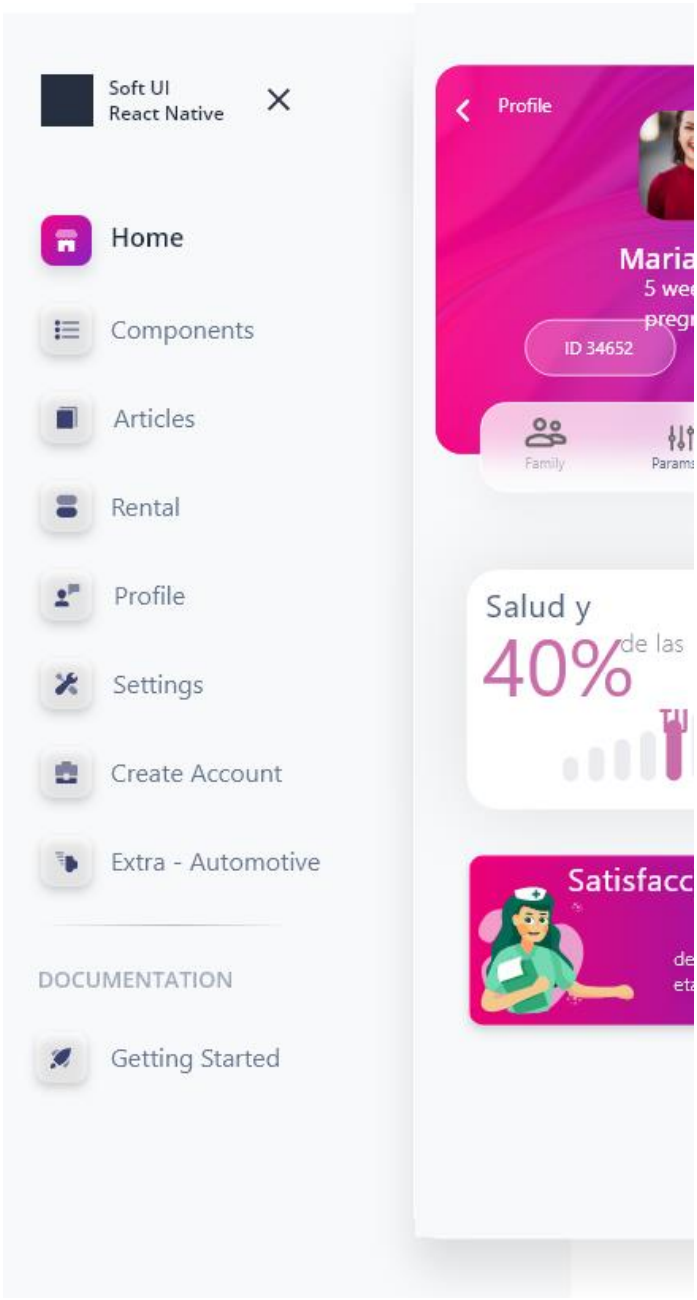
the trigger manager, detects the events and acts accordingly, the events will be accumulated in a processing queue, so that each one will be responded to according to its order of arrival, although if some of them are marked with an "urgent" label, they will take priority in the queue.

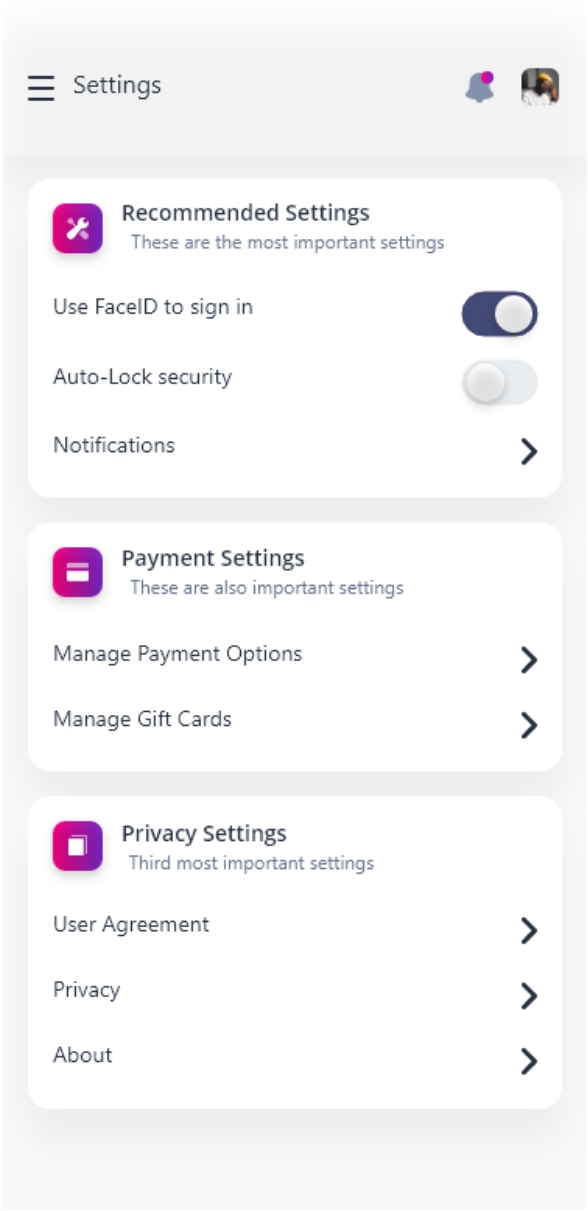
There is also a management of scheduled events, these are executed every day at 23 hours and 59 minutes, which causes the update of the server data and respectively the scheduled tasks are processed, at the end of which notifications are sent to users regarding the available surveys that they can answer.

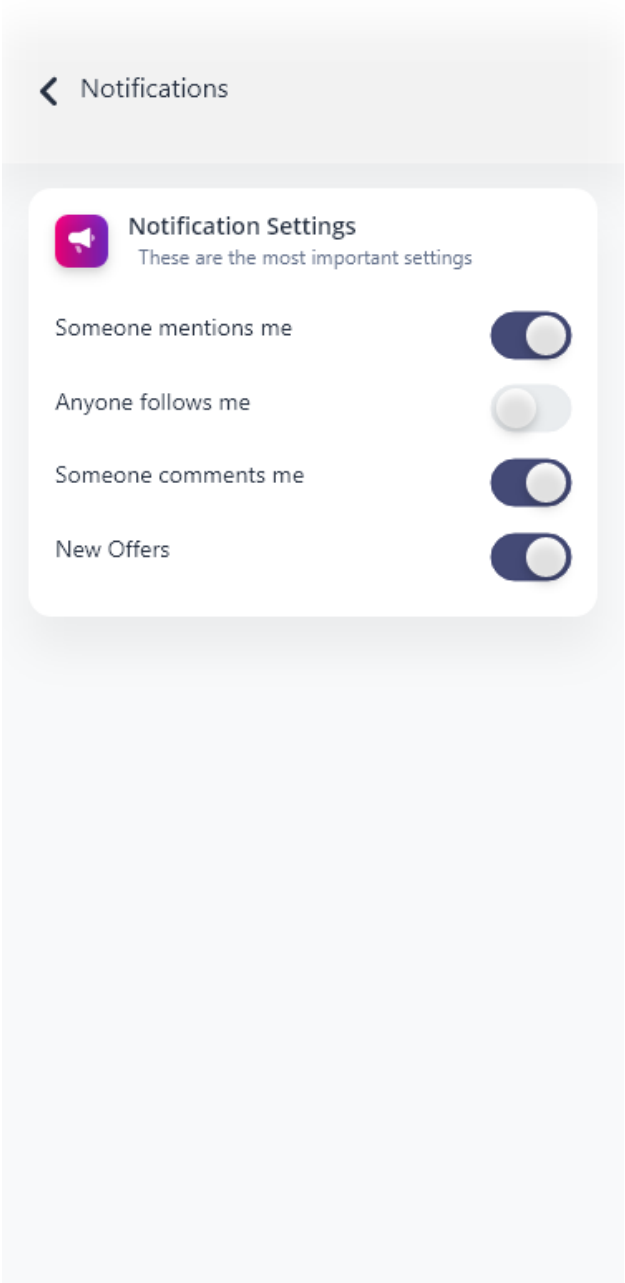

Welcome
or
Register with
Name
Name
Email
Email
Password
Password
I agree the Terms and Conditions
SIGN UP


Profile
Maria Snow
5 weeks of pregnancy
ID 34652
A+
9/12
Family
Params
Documents
Salud y Bienestar
40%
de las mujeres en tu etapa de embarazo se siente mejor.
Satisfacción con la atención
70%
de las mujeres en tu etapa de embarazo consideran que han recibido una mejor atención
Mi hospital
7.9
Salud y bienestar
Mi hospital
7.5
Satisfacción con la atención


Back
4
Question category
Estado
Text of the Question
Descriptive text of the question if applicable
Opcion 1
Opcion 2
Opcion 3
Opcion 4
Text of the Question
Select date
Text of the Question
01
Text of the Question
Luxury
Submit


Soft UI
React Native
Home
Components
Articles
Rental
Profile
Settings
Create Account
Extra - Automotive
DOCUMENTATION
Getting Started
Profile
ID 34652
Family
Salud y
40%
de las
Satisfacc


Settings
Recommended Settings
These are the most important settings
Use FaceID to sign in
Auto-Lock security
Notifications
Payment Settings
These are also important settings
Manage Payment Options
Manage Gift Cards
Privacy Settings
Third most important settings
User Agreement
Privacy
About


Notifications
Notification Settings
These are the most important settings
Someone mentions me
Anyone follows me
Someone comments me
New Offers

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5
Paragraph Paragraph

Gradients


Primary


Secondary


Info


Success


Warning


Danger

Light


Dark

Colors


Primary


Secondary


Info


Success


Warning


Danger

Light


Dark

White

Buttons

Inputs