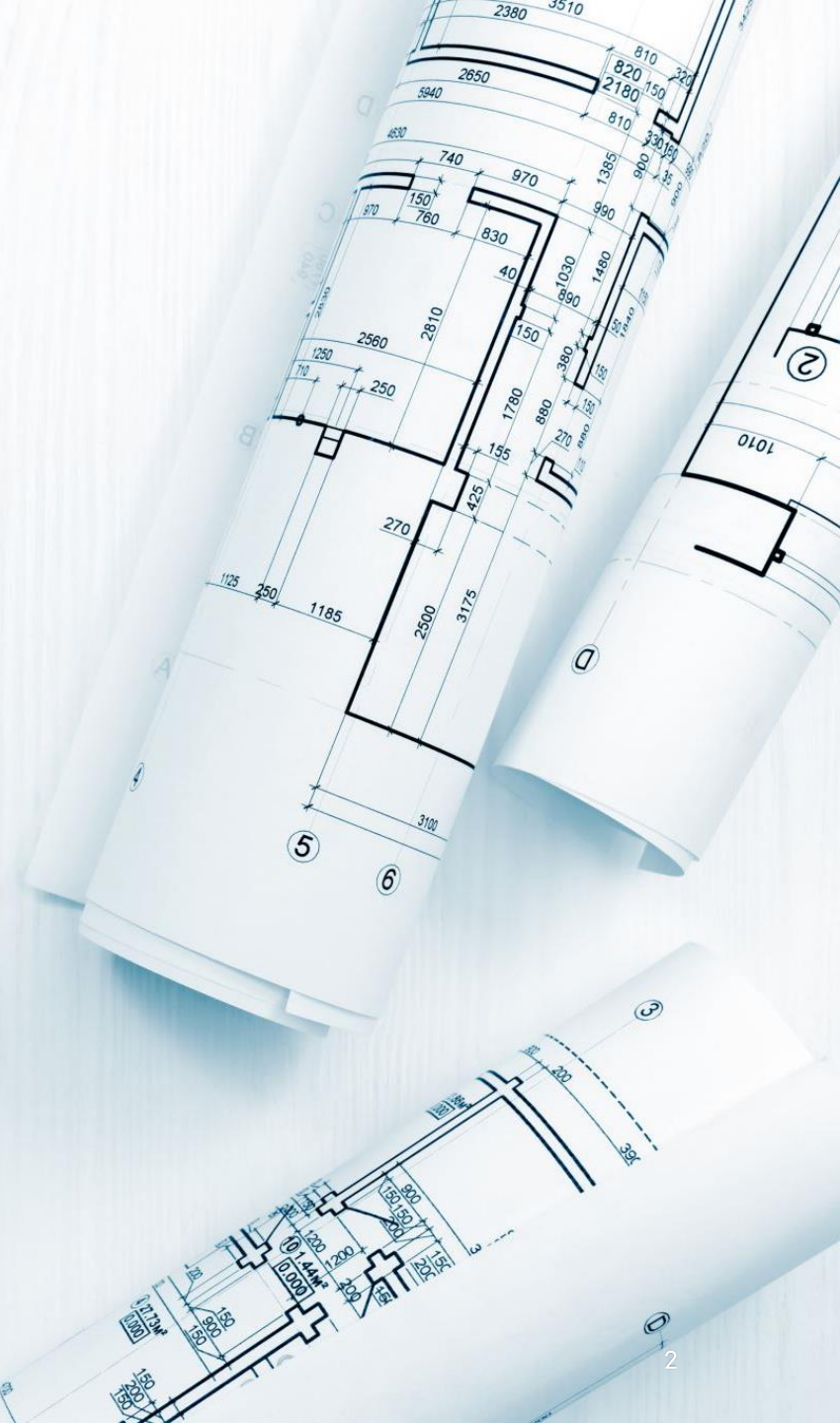


LoMap_en2b



Table of Contents

- Work division
- Architecture
- Requirements
- Technologies used
- Testing
- Deployment
- Remarkable Architectural Decisions



Work division



- Webapp: it was mainly developed by Jorge helped by Diego, with some help from Andrés and Pedro in some pages.

Restapi: it was mainly developed by Andrés and Pedro.

SOLID persistence: it was mainly developed by Diego.

Tests: they were mainly developed by Pedro with help of Diego and Andrés, as well as from Jorge.

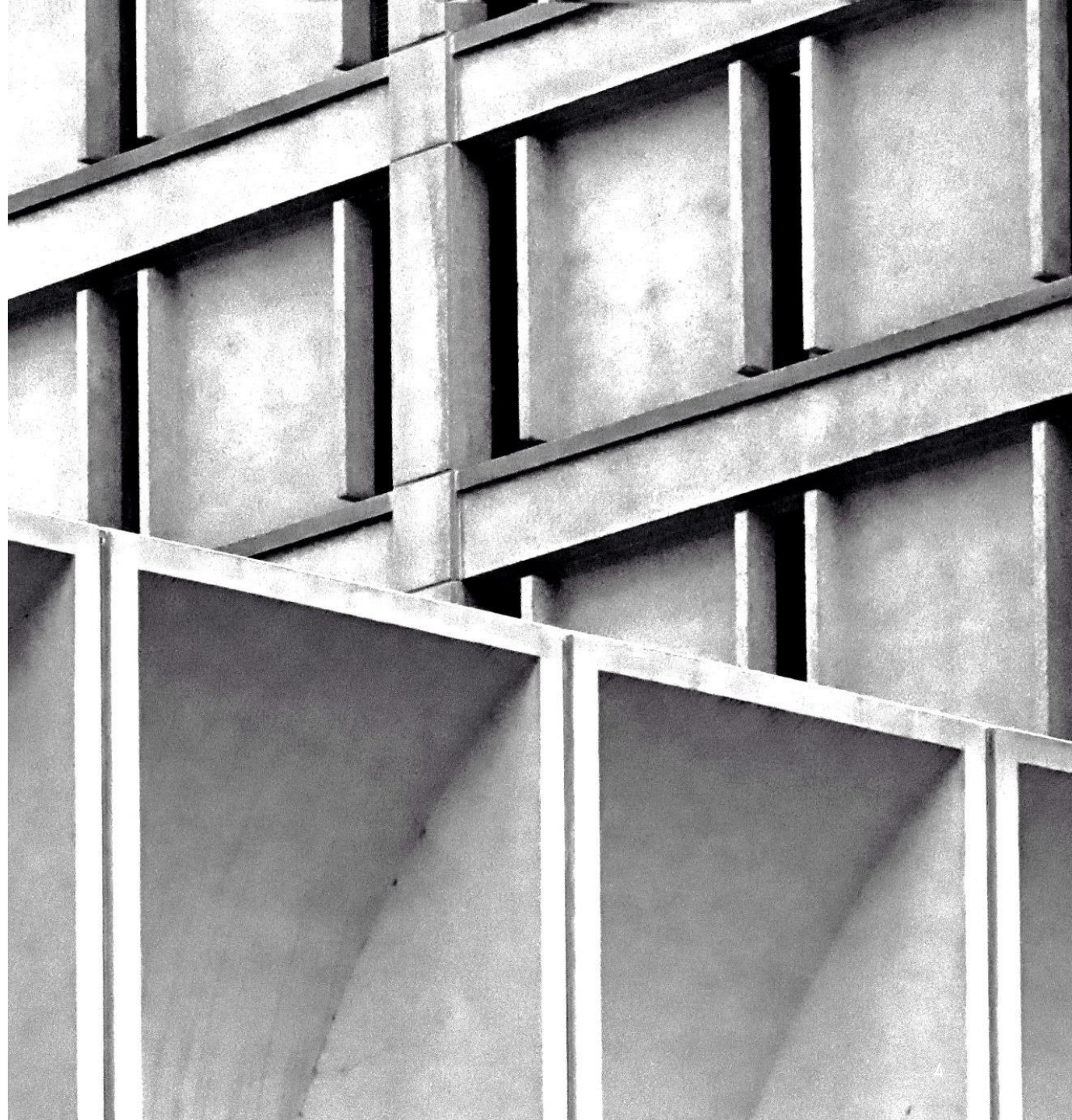
Deployment: done by Andrés.

Documentation deployment: done by Jorge.

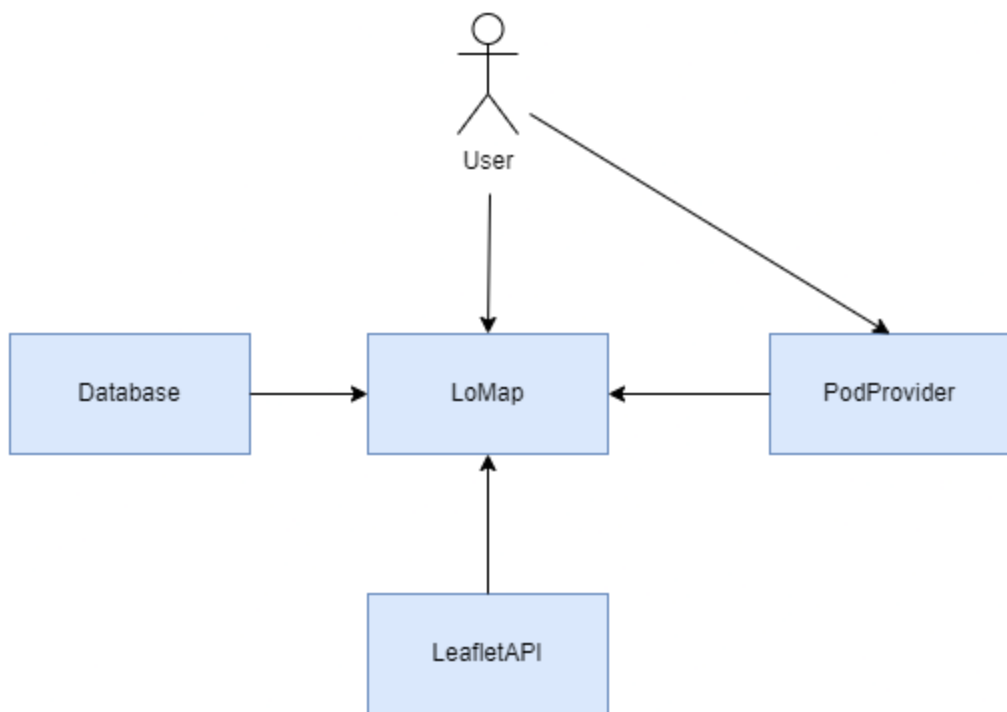
We divided the points from the docs equitatively

Architecture

We decided to follow a simple two-layer architecture, composed of the frontend (or *webapp*) and the *controller* (or *restapi*), though currently some bits of controller logic are mixed within the frontend for practicality and interoperability purposes.

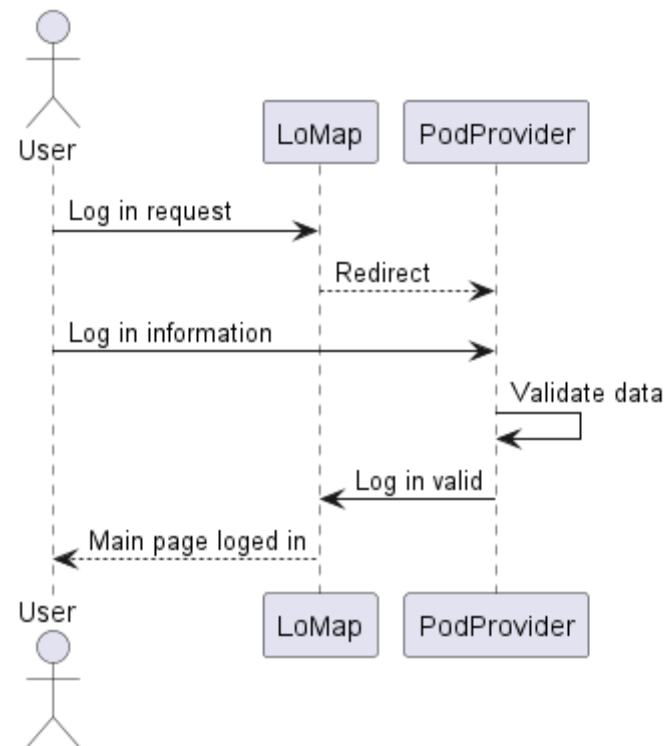


Architecture



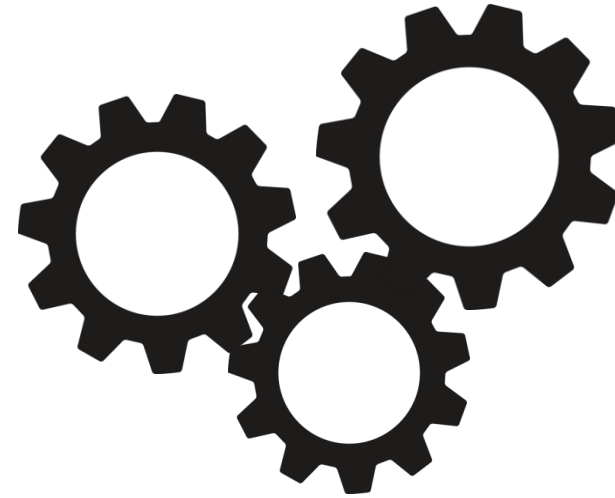
System context

Log in to the application



Log in process using the
SOLID available library
@solid-ui-react

Requirements



- Add landmarks to the map ✓
- Filter the information on the map ✓
- See other users' additions to the map ✓
- Add a review, comments or pictures to the uploaded places ✗
- Manage their data from their SOLID pods ✓

Technologies used



- Node.js 16.19.1
- React
- Typescript (using ES2020)
- SOLID (for data storage)
- MongoDB (for user reference storage)
- Material UI for React
- Leaflet & React-Leaflet

MongoDB



- Semi Decentralized architecture
- Database is used to store reference of solid users who are using lomap.
- This helps to navigate the webpage and load the different users
- Only friends of solid that use lomap are shown, therefore avoiding “undesired” friends

React & nodejs (Typescript)



- It was, for most of us, the first time developing an application with these technologies.
 - Not knowing which modules to use, and how
 - Using properly the webapp and restapi too time-expensive in the start.
 - Problems regarding the types.
 - In some computers at home, the maps never loaded correctly.
 - We are not used to code using a declarative programming language as complex as React



Webapp (frontend) features

- We used the Material UI library for React.
- The React-leaflet library was used for the maps (which had a very limited documentation)
- React 18
- A lot of logic for the frontend is in the frontend itself.

SOLID

- We faced some trouble, remarkably:
 - Lack of experience in working with new, unexplored technologies
 - Managing the permissions of the data in the PODs



LOAD, JEST & E2E

Testing



Jest testing

src/pages/addLandmark	53.48	20	71.42	54.76	
AddLandmark.tsx	53.48	20	71.42	54.76	29,39-71
src/pages/friends	75	50	60	75	
Friends.tsx	75	50	60	75	26,43
src/pages/home	71.42	50	80	70	
Home.tsx	71.42	50	80	70	19,26-27,34-40
src/pages/login	75	100	50	75	
Login.tsx	75	100	50	75	27
src/pages/otherUsersLandmark	32.81	18.75	31.25	36.2	
LandmarkFriend.tsx	32.81	18.75	31.25	36.2	27-29,34,42,52-56,60-61,96-144
src/pages/profile	67.85	42.85	42.85	79.16	
Profile.tsx	67.85	42.85	42.85	79.16	31,39-42
src/pages/users	75	50	60	75	
Users.tsx	75	50	60	75	25,43

- Divided in two parts: webApp and restApi
- In webApp: we test pages and app.tsx
- Making sure all that has to load, has loaded
- In the restApi, we tested the calls to the api
- We made one correct call and one incorrect to each call
- All the tests in the webApp and in the restApi should run

e2e testing

about:blank x Login x +

← → ↻ inrupt.net/login?client_id=50098e346619d9a0ce013a2f9625445d&redirect_uri=http%3A%...

Un software automatizado de pruebas está controlando Chrome.

Login

Username ArqSoftLoMapEn2b

Password •••

Log In

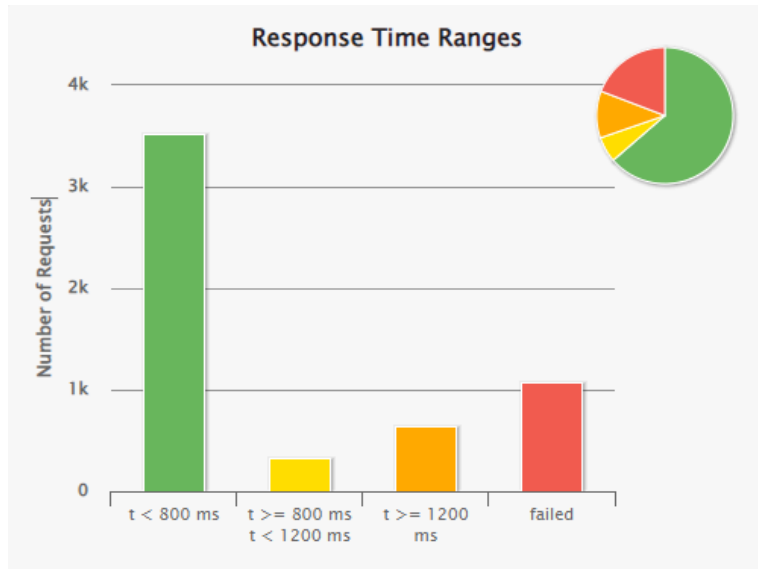
Forgo

Log in with Certificate (WebId-TLS)

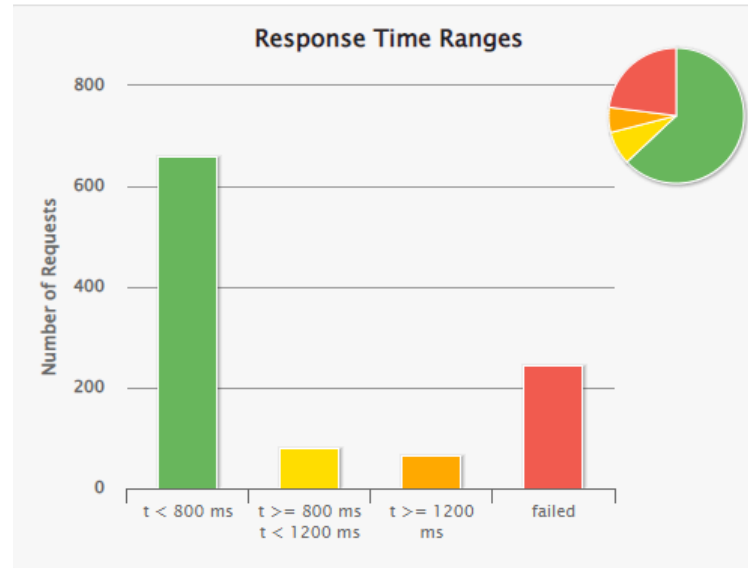
New to Solid? [Create an account](#)

- We decided for the third deliverable all the features to do
- This past three weeks we have been trying to make them work
- Over the three last weeks, we implemented them
- We got lots of problems that we could solve
- This last week we arrived to a complete nonsense of a problem
- We have the correct code for the tests, but when the tool finishes login, it cannot load the home page
- No trace for a possible error is shown and we could not solve that either

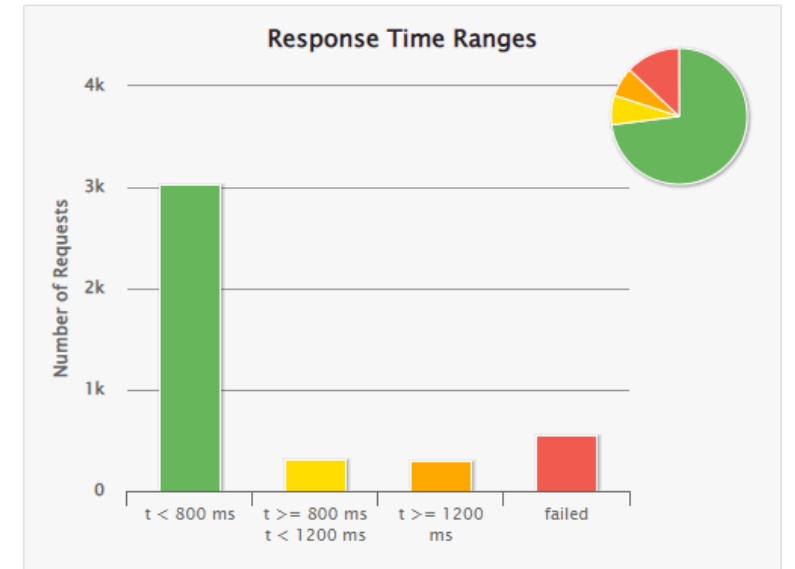
Load Tests with Gatling



250 sims in 1 minute



25 sims at the same time



150 sims in 1 minute

The most common errors were

j.n.c.ClosedChannelException	721	66.759 %
status.find.in[[200, 209], 304), found 500	219	20.278 %
j.n.ConnectException: Connection refused: no further information	140	12.963 %

Deployment

- Public IP <http://172.162.240.176:3000>.
- It uses both Docker images in an Ubuntu machine.
- Due to a warning deletion error tests are skipped.
- Accessed through ssh
- The deployed version does not store session due to a SOLID error with https



^ Información esencial

Grupo de recursos ([mover](#)) : [DeploymentASWlomap_en2b](#)

Estado : Detenido (desasignado)

Ubicación : Switzerland North (Zona 1)

Suscripción ([mover](#)) : [Azure for Students](#)

Id. de suscripción : c8f13624-32f0-4ebd-8115-052075125941

Zona de disponibilidad : 1

Etiquetas ([editar](#)) : [Haga clic aquí para agregar etiquetas.](#)

Sistema operativo : Linux

Tamaño : Standard DS1 v2 (1 vcpu, 3.5 GiB de memoria)

Dirección IP pública : [172.162.240.176](#) 

Red virtual/subred : [DeployLomapEn2b-vnet/default](#)

Nombre DNS : [Sin configurar](#)

Estado de mantenimiento : -



Remarkable Architectural Decisions

- Architectural Decisions are written in the point 9 of documentation.
- There are both critical and non-critical decisions written.
- These are the ones related with the way we stored the landmarks on solid
- Biggest decision was to move part of the solid logic out from the API
- This change was done to try to make an approach for interoperability with group en2a

LoMap_en2b



THANKS!