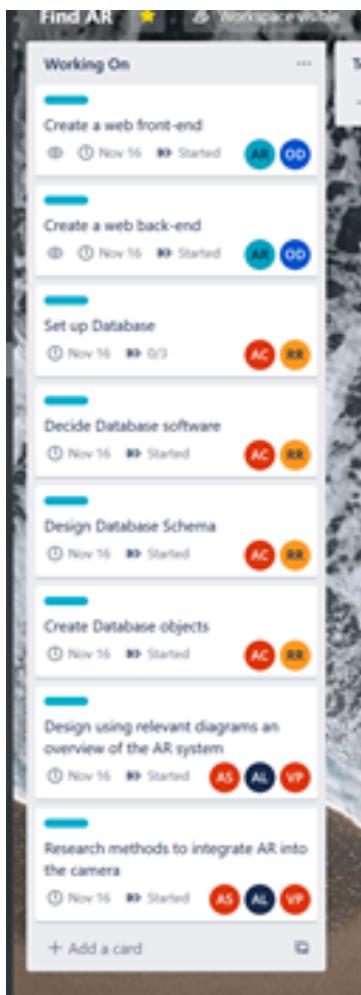


Sprint 1 (09/11/2022 -> 16/11/2022)

In this sprint, we decided to first create the foundation of our project. Thus, it was made to be shorter than normal.

The tasks assigned were:



We split the team into 3 subteams:

- AR Team -> Designing/researching how to use AR in websites.
- Website Team -> Setting up a website that everyone can develop on in the future
- Database Team -> Designing some extensible database that reflects our current goal of a viewable list of tags and a user login system.

We have also set up the sub-teams to reflect the skills of everyone by allowing people to be in the teams which they have the most experience with and most comfortable with. However, for the AR Team, no one in the team has much practical experience, this is why we made it the largest team so that we can focus more on researching and understanding the AR aspect.

Our plan is to have members in each team switching over throughout the course of the project to allow the whole team to collaborate with each other and develop a firm understanding of every aspect within the project.

Sprint 1 Retrospective

We had a retrospective on 16/11/2022 to discuss Sprint 1.

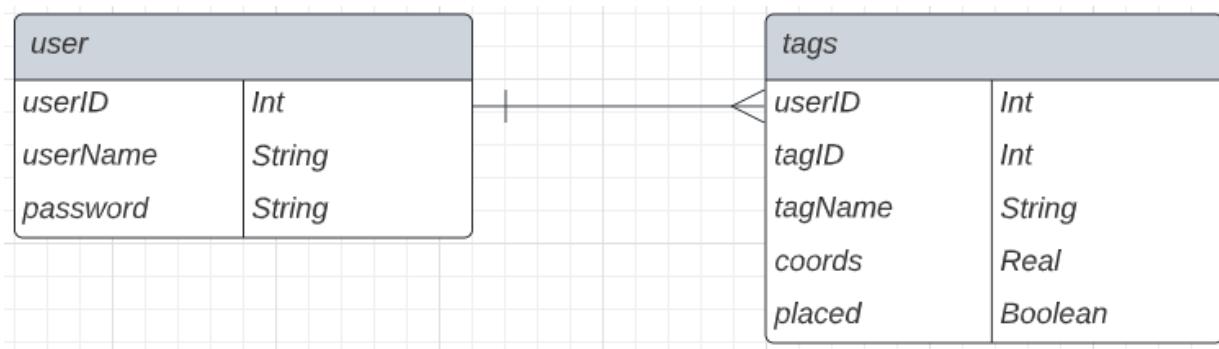
Sub-Team Website Creation:

- We were unable to progress much due to being unable to access IBM services.
- During this we with the team lead set up a meeting with our sponsor and got everyone to sign up to IBM services as the prerequisite to access the services.
- Thus, these tasks have been moved to the next sprint to do.

Sub-Team Database Creation:

- We were able to decide on a database creation software.
- We decided on IBM's db2 as this software is stored on IBMs cloud which will allow us easier integration with our IBM cloud server when it is up and running.
- This software operates using SQL documents which we can load into the program to perform actions on the database.

Next we designed a preliminary database schema which includes record that we have deemed key to this project



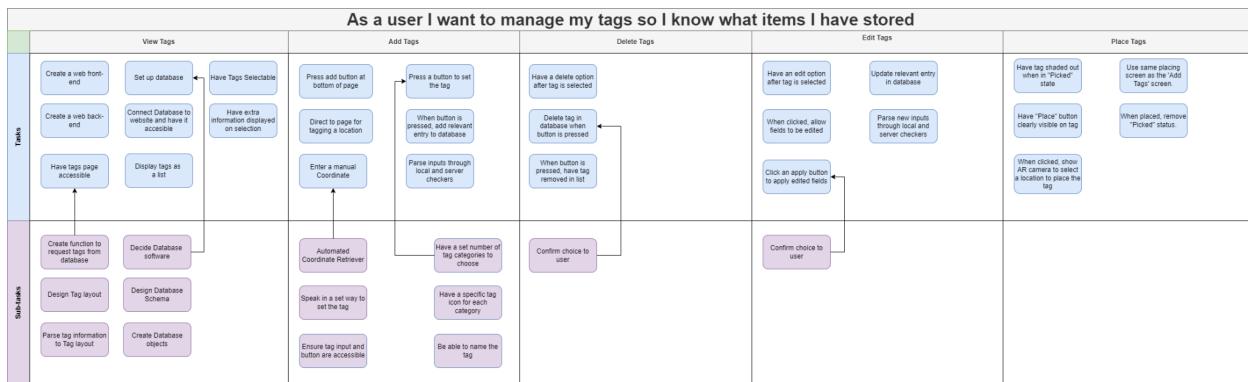
Using the above schema we drafted SQL code to create the relevant tables and the connection between them. This code also insures that the ID values are auto-incrementing so we don't have to manually keep track of IDs.

Sub-Team AR:

- Our primary goal during this sprint was to research methods to develop a prototype of a location based AR token system.
- The main tool we found that will help is the AR.js studio code generator which will generate code for us to use.
- Something we realised after this sprint was that we severely overestimated how much time the AR aspect of this program will take us, we don't think it will be necessary for us in the future to have in depth knowledge of AR methods we just need to use the AR.js pre-written code and embed it into our website.
- The specific AR Problem we are solving also seems to be well documented.
- The goal for the next sprint is to get some kind of actual prototype up and running so that we can demonstrate location based AR tokens.

Sprint 2 (17/11/2022 -> 1/12/2022)

After reviewing our previous sprint, we found that we needed to change our 'View Tags' Epic to accommodate for the next step, integrating and connecting that database to a web-server:



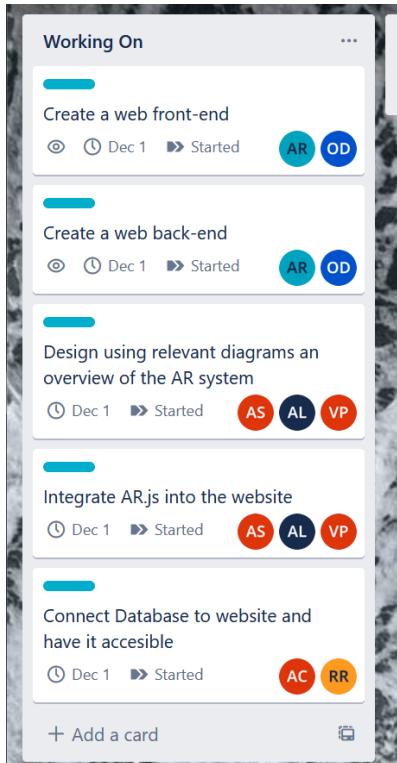
A story map example in a cross-functional flowchart shape (Misc shape library)

We assigned "Connect database to the website and have it accessible" to the story map, as it makes more sense in the context of this retrospective.

We discussed to include a new user-story to 'integrate AR.JS' into the soon-to-be webserver, as a more specific variation of the problem:

As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have an AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera Integrate AR into the website Design using relevant diagrams an overview of the AR system Have a page for viewing the tags in AR Have the website access the camera Form an icon to float above the exact location 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located 	<ul style="list-style-type: none"> Have a button to pickup a tag when near it When clicked, print the tag into "Picked" status Implement reminders to place tags back
		<ul style="list-style-type: none"> Perform a visual indicator of distance 	

Finally, We assigned new tasks to be completed:



Sprint 2 Retrospective

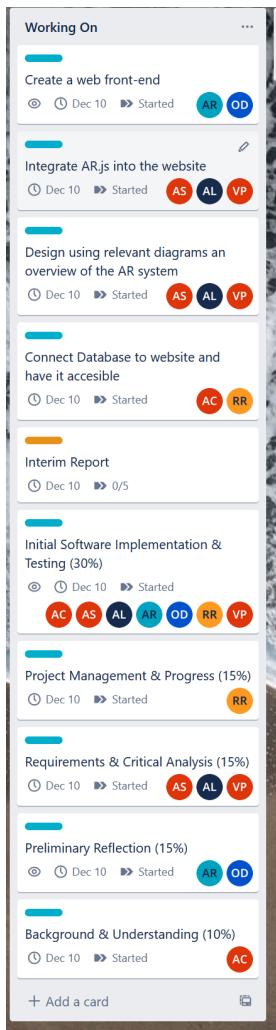
We had a retrospective on 01/12/2022 to discuss Sprint 2.

In the Sprint 2 retrospective we discussed the difficulties in server creation. Due to delays in access to IBM's server, half of the sprint was in a deadlock, and eventually the Website Creation sub-team found a serverless implementation of Node.JS in IBM Cloud that could function as a stop-gap before getting full access to IBMs resources.

However, Severless has its one challenges, the most prevalent being the requirement of local copies to properly develop, due to the CI deployment taking more than 3 minutes. In any case, this causes a week delay in being able to start, so we decided to move over the remaining user stories to a week +2 day sprint.

Sprint 3 (01/12/2022-07/12/2022)

During the meeting, we discussed the Interim Report, and thus assigned parts of it to everyone to be done in the same sprint, from which we will send the draft to our supervisor for advice:



At the moment, the only completed task was the “Create a web back-end”, which was the Node.JS server creation using a Serverless implementation on IBM Cloud.

We also discussed difficulties for the Database Creation Sub-Team, as IBM only allowed proprietary Database solutions to be created on our limited accounts.

Thus, they had to move to another hosting solution for MongoDB, which is a more traditional, and open source implementation of MySQL which doesn't require absurd drivers to access.

The AR Sub-Team has had more trouble, as the week delay, coinciding with the difficulty in setting up a local repo Docker container to develop in, made progress difficult. This is also ignoring the debugging nightmare of not having an easy way to test the website on your phone, because it is a local server all the way until deployment via a merge to master.

However, their task is to just integrate AR.JS into the front-end, so it should be plausible to achieve, which we all agreed on.

Sprint 4 (31/01/2023-07/02/2023)

Here we discussed what user stories to create next.

We found that certain user stories were missing:

- The AR icon was missing a follow-up from integrating AR.js into the website
- To then use the AR for our purpose we then need to have the facility to get geo-location from the user.
- Add an explicit design for the tag layout. On the tag layout page that will show the tags the user has, we have yet to properly decide how it will function, so it needs to be designed first before implementing a template on the front-end.

As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR Icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Integrate AR.js into the website Be able to get geo-location data in longitude and latitude Design using relevant diagrams an overview of the AR system Have the website access the camera Form an icon to float above the exact location 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located Have the user press a button to state they have reached their location. 	<ul style="list-style-type: none"> Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Subtasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR Make the page automatically open the camera 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

As a user I want to manage my tags so I know what items I have stored					
	View Tags	Add Tags	Delete Tags	Edit Tags	Place Tags
Tasks	<ul style="list-style-type: none"> Create a web front-end ✓ Create a web back-end Have tags page accessible Set up database Connect Database to website and have it accessible Parse tag information to tag layout Create function to request tags from database Design Diagrams for Tag Layout and Usage Design Tag layout 	<ul style="list-style-type: none"> Have Tags Selectable Have extra information displayed on selection Press add button at bottom of page Direct to page for tagging a location Enter a manual Coordinate Automated Coordinate Retriever Have a set way to set the tag Be able to name the tag 	<ul style="list-style-type: none"> Have a delete option after tag is selected Delete tag in database when delete button is pressed When button is pressed, have tag removed in db 	<ul style="list-style-type: none"> Have an edit option after tag is selected When clicked, allow fields to be edited Click an apply button to apply edited fields 	<ul style="list-style-type: none"> Update relevant entry in database Parse new inputs tag input and server checkers Have tag shadowed out when in "Picked" state Use same placing screen as the Add Tag screen Have "Place" button clearly visible on tag When placed, remove "Picked" status When clicked, show AR camera and select a location to place the tag

Integrate AR.js into the website	Feb 7	Started	AS
Be able to get geo-location data in longitude and latitude	Feb 7	Started	AS
Design using relevant diagrams an overview of the AR system	Feb 7	Started	AS
Set up Database	Feb 7	2/3	AR OD RR
Connect Database to website and have it accessible	Feb 7	Started	AR OD RR
Create Database objects	Feb 7	Started	AR OD RR
Have Tag's page accessible	Feb 7	Started	AC AL VP
Design Diagrams for Tag Layout and Usage	Feb 7	Started	AC AL VP
Design tag layout	Feb 7	Started	AC AL VP

On that note, assigning tasks went into three teams. Alfie, Rodion, and Oliver were tasked with migrating the existing MySQL schema to a NoSQL implementation due to technical constraints, these being the tasks: “Connect Database to website and have it accessible”, “Create Database Objects” and “Set up database”. We assigned Alfie and Oliver as they were previously working on the server, and so haven’t had any experience with the database software, but still have backend knowledge to help with eventual integration.

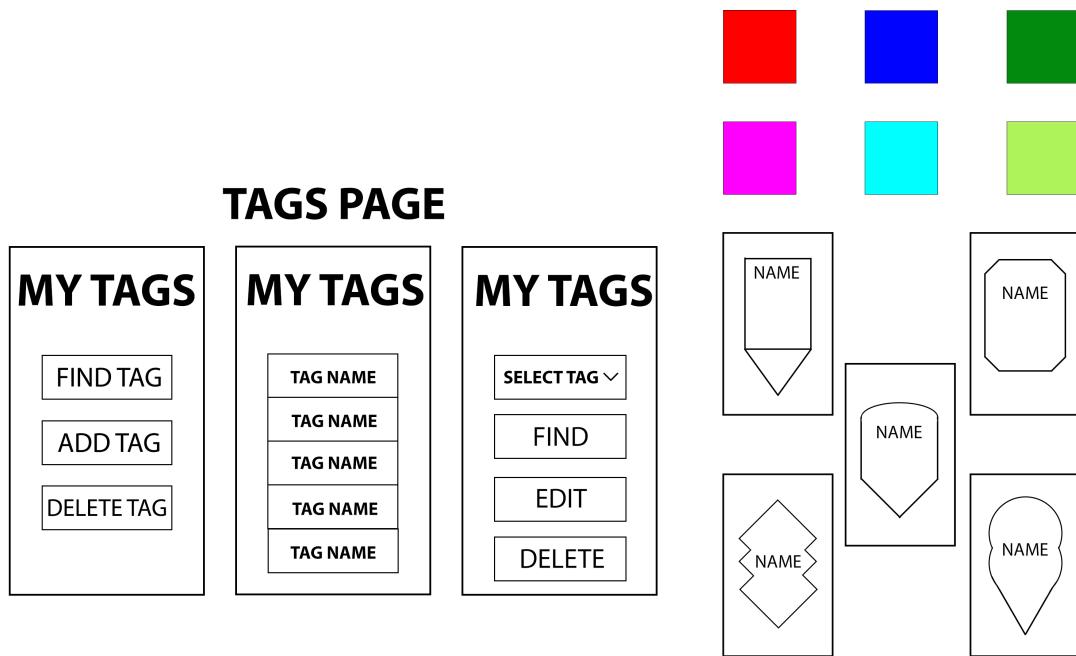
After this, we needed to create the tags page which will display the users’ tags that they have saved. This tags page needs to be designed with how it will function, and how the tags will look and behave, and we also need a route to get to this page that isn’t the root directory. So, we assigned Aidan, Vishal and Alex. Aidan and Vishal are both doing Human Computer Interactions, so we felt that they would be best tasked to designing an interface. All of them need to gather experience working with Svelte. The tasks assigned are “Have Tag’s page accessible”, “Design Diagrams for Tag Layout and Usage” and “Design Tag layout”.

The last assignment was for the AR theme, and generally Alan has been working on integrating the AR.js library on and off during the holiday. He aims to integrate the AR.js code so that you can view an item in AR that spawns on a specific location and keeps track of your location relative to it. This is all done via AR.js and so isn’t a major hurdle. The tasks assigned were, “Integrate AR.js into the website”, “Be able to get geo-location data in longitude and latitude” and “Design, using relevant diagrams, an overview of the AR system”.

Sprint 4 Retrospective

Design Front End Team

Aidan tackled the user stories focused on the design of the tag page and the tags itself. He created mockups that can be used as a basis for the design of the front end as well as a focal point for design discussions. These designs are not final and are meant to be used to explore which direction we want to take with the UI.



*Note the 3 designs of the tags page are separate concepts but they may be interwoven to produce a final product. For example design 1 could pop up over design 2 after the user has selected a tag from the list.

These designs were built upon research into what makes a good UI and specifically what makes a good UI for the elderly/disabled. In summary, from this research we decided to keep things as simple as possible so as to not overload the user. The main keyword is comfortable, we want our user to feel relaxed while achieving their end goal. Taking this into account, the design uses simple shapes with their actions/information clearly indicated.

≡ What Makes A Good UI

Moving forward we hope to refine these designs and then implement them into the webpage.

AR Front-End Team

Most of this sprint was spent actually implementing AR into code, unfortunately due to the lack of a server we were unable to deploy our working prototype which was necessary as we needed to be able to access our application from a phone in order to use the camera and gps facilities that it would provide.

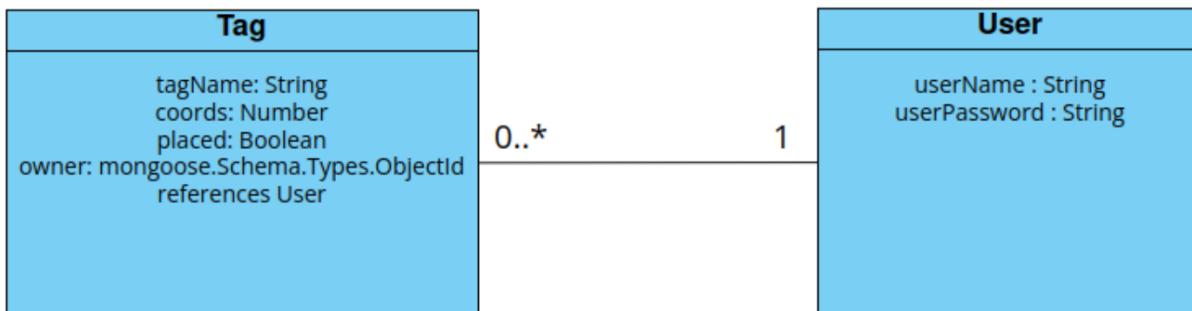
Alan found a workaround to this issue by creating his own private repo on github and using the github pages features, which allows easy deployment of the website meaning we could test our website on your phones.

After the working implementation was made, testing was done and another problem was found. The accuracy of the marker was lacking, the main problems seems to be with the accuracy of GPS data whilst indoors, more research on this problem will be done next sprint. Additionally there is no way for the marker to indicate elevation, further research will again be done to try solve this issue.

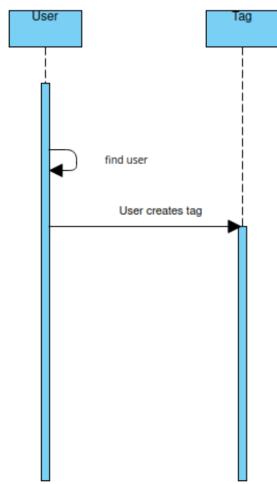
Database Back-end team

The goal of this team was to convert the MySQL schema and code into MongoDB, and then implement a connection in the serverless website.

To outline the new schema, Alfie implemented this into node.js to connect to the MongoDB free server so that we could implement these schemas. Rodion then outlined these two objects:



And as a usage:

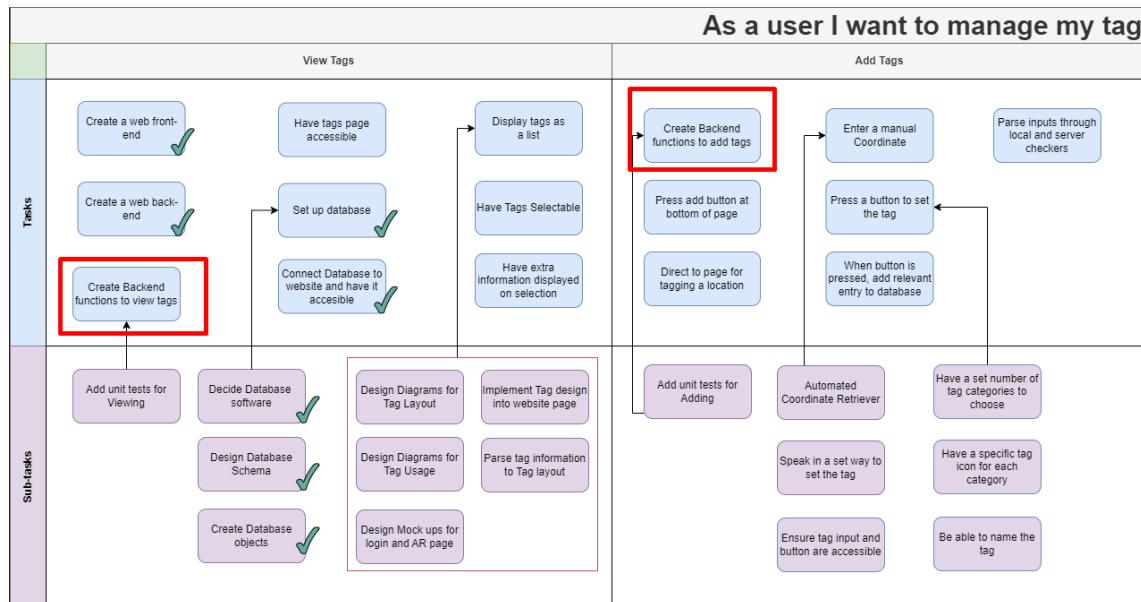


Therefore, we were able to complete all three user stories: "Connect Database to website and have it accessible", "Create Database Objects" and "Set up database".

Sprint 5 (07/02/2023-14/02/2023) (Sprint 4 Retrospective)

We discussed a focus on the diagrams of how AR.JS will be used when going to the AR page in the website, and integrating AR.js into the website is a trivial exercise of moving the code from github to the svelt front-end.

For the Backend Database team, all of their user stories were completed. However, after this we found a severe lack of explicit user stories to design parts of the backend, such as viewing tags, deleting them, etc. So, we redesigned the story map and conjured these additions:



So I know what items I have stored

Delete Tags	Edit Tags	Place Tags
<p>Create Backend functions to delete tags</p> <p>When button is pressed, have tag removed in list</p> <p>Have a delete option after tag is selected</p> <p>Delete tag in database when button is pressed</p> <p>Confirm choice to user</p> <p>Add unit tests for deleting</p>	<p>Create Backend functions to edit tags</p> <p>Click an apply button to apply edited fields</p> <p>Have an edit option after tag is selected</p> <p>When clicked, allow fields to be edited</p> <p>Add unit tests for editing</p> <p>Confirm choice to user</p>	<p>Create Backend functions to place and pickup tags</p> <p>When clicked, show AR camera to select a location to place the tag</p> <p>Have tag shaded out when in "Picked" state</p> <p>Have "Place" button clearly visible on tag</p> <p>Add unit tests for placing and picking up</p>

These user stories allow us to explicitly assign tasks to the backend functionality, and of course are required for any functionality needing the database. So, we decided to assign:

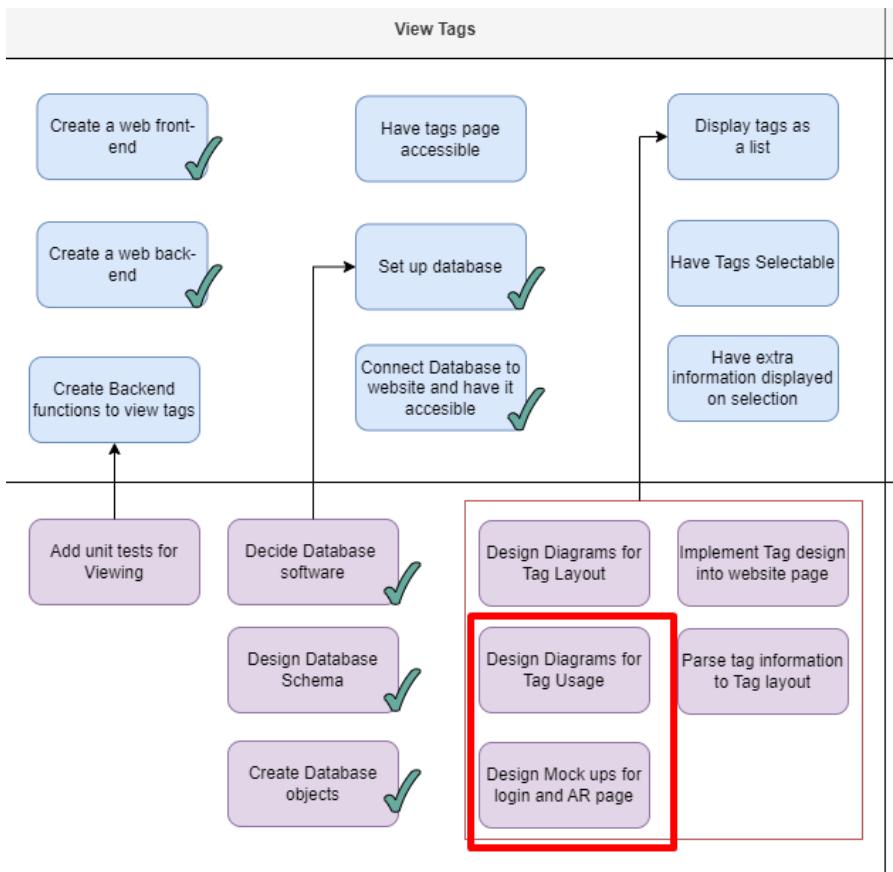
The image shows three Jira cards in a board view, all assigned to the 'Backend' column. Each card has a blue progress bar at the top.

- Create Backend functions to view tags**
Due Feb 14, AR, OD, RR
- Add unit tests for Viewing**
Due Feb 14, AR, OD, RR
- Secure Password**
Due Feb 14, AR, OD, RR

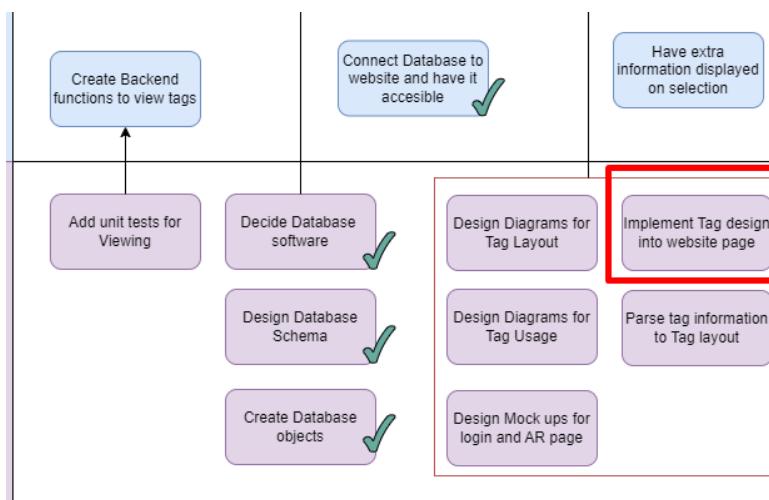
This will implement the functionality of getting tag data, and unit tests for this. This also implicitly includes diagrams for the expected usage of viewing tags. If this is done quickly we will assign more of these backend functions, because they can be done in parallel.

For the Designing and creation of tags, the sub-team were able to come up with designs of how the tag will look in AR, and some prototypes of the tag page. Due to it being a new framework, the team needed more time to properly implement these designs into Svelte.

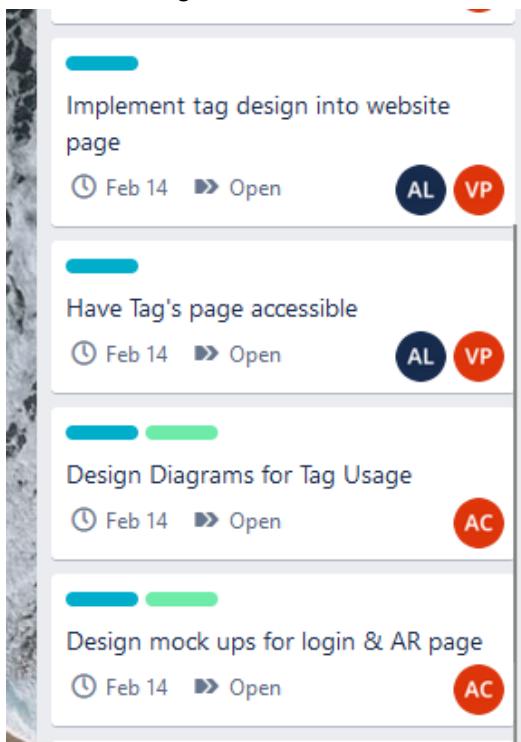
We also added a new user story to make explicit the design of the tag usage, and the mock ups for the login and AR page and how they will flow:



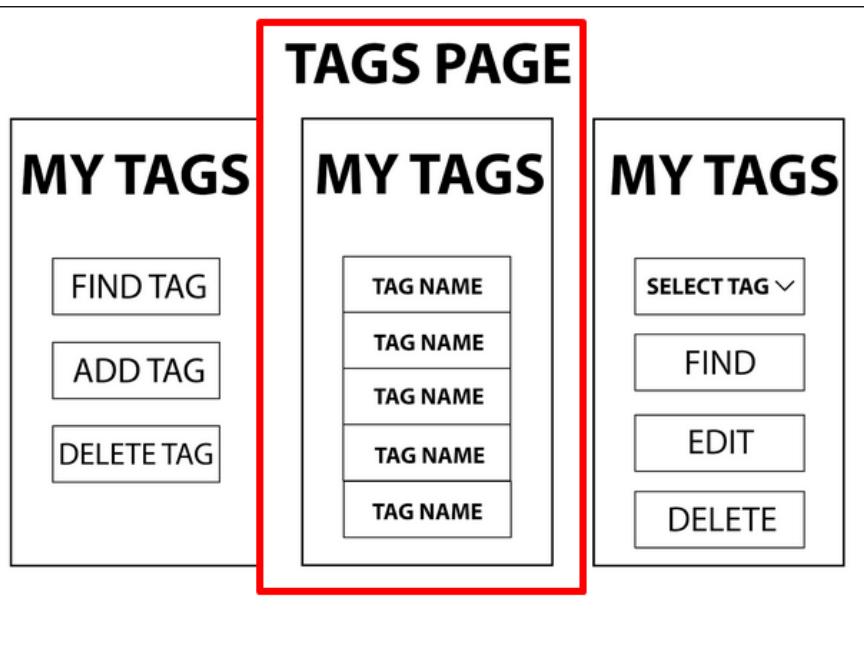
We also added a new user story to make it explicit for implementing the tag design we chose into the website:



Thus the assignments were:



And, for the page design we chose, it was the list one:

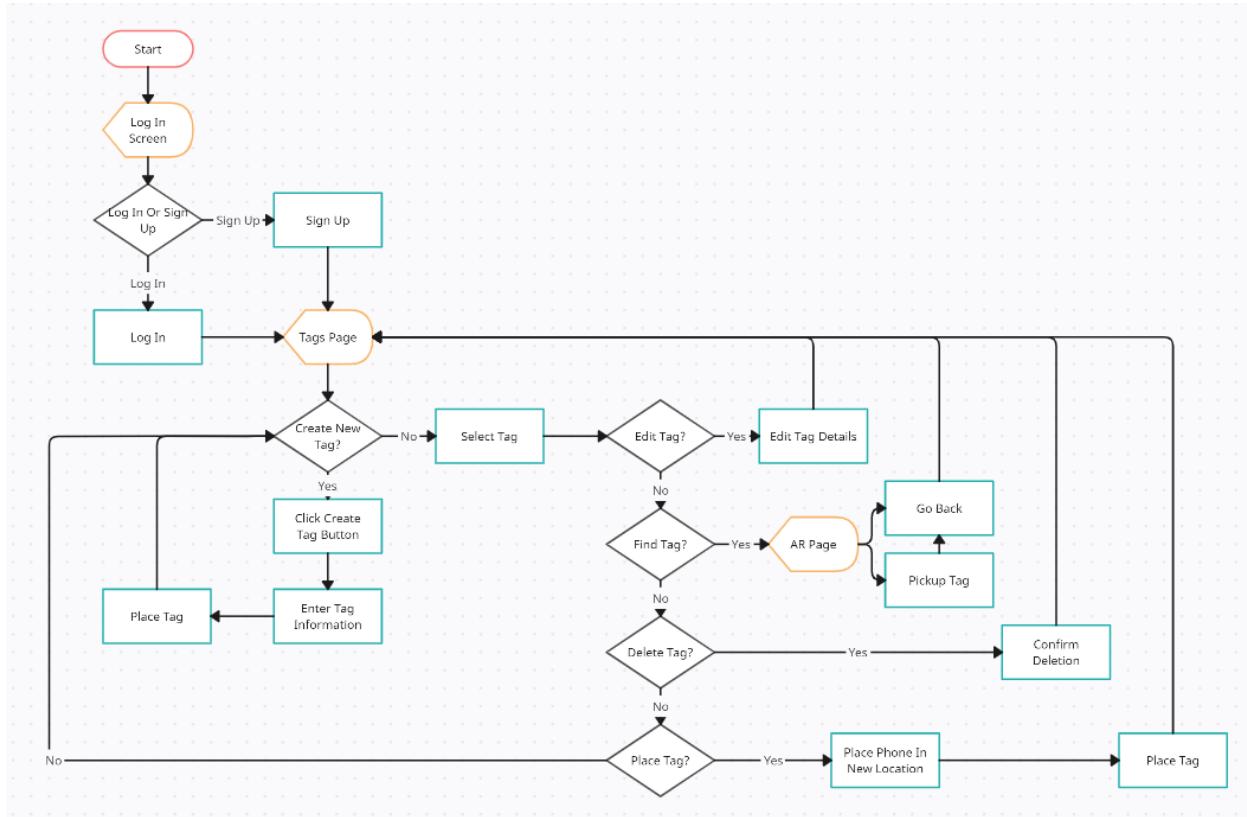


We chose this design because we felt that it is the easiest one to use and understand. This is because it does not require the user to navigate through many pages or click lots of buttons to find their tags since they are all displayed in one screen.

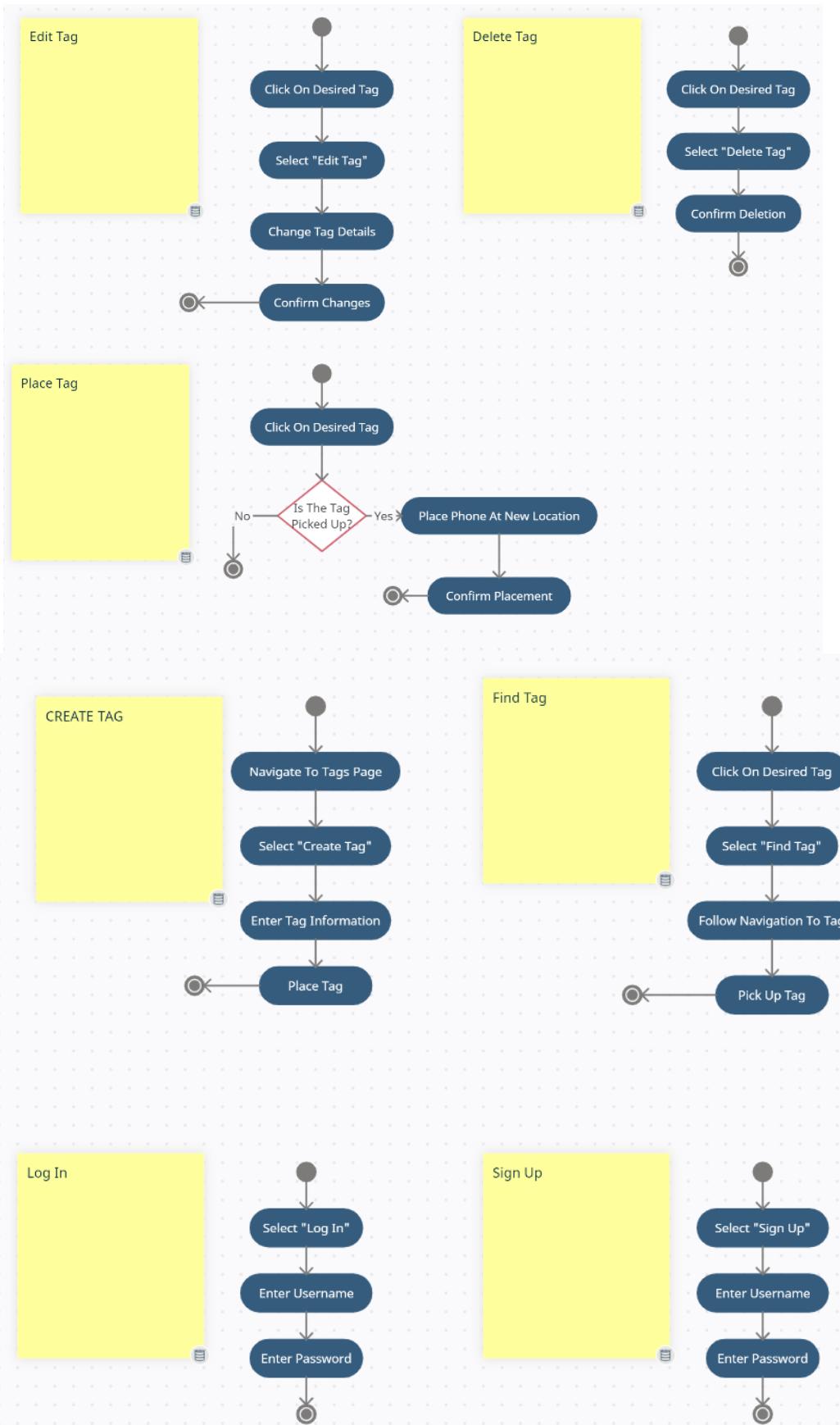
Sprint 5 Retrospective

Design Front End Team

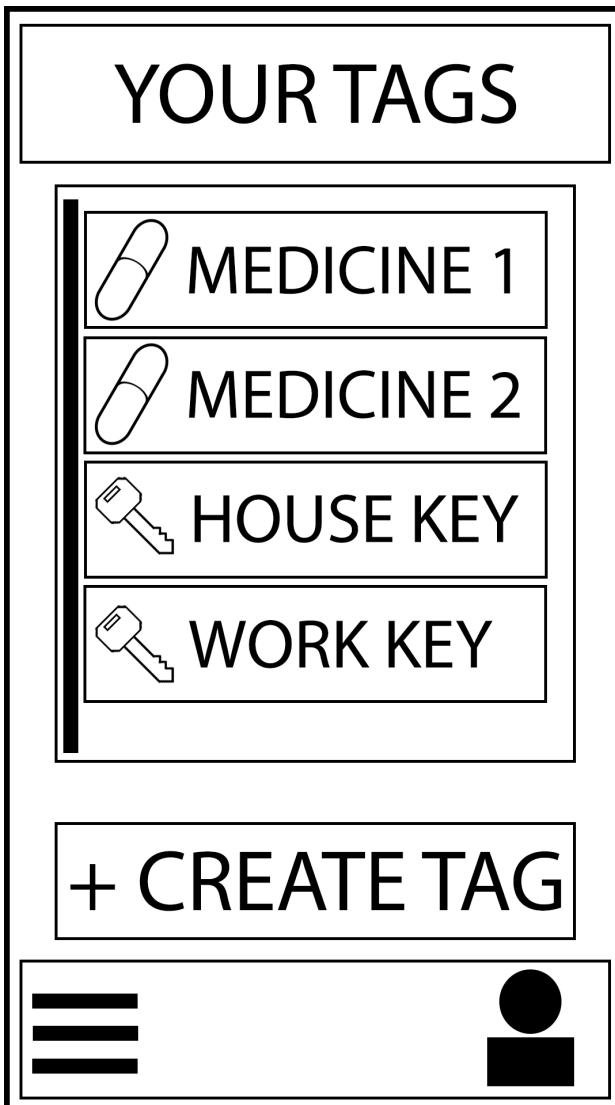
After Aidan presented the mockups to the team a discussion was had about the overall flow of the web app. Based on these discussions we drew up an overall flow diagram to depict how the user will use our product.



Building upon this diagram we also developed a series of activity diagrams showing the flow of specific actions the user will carry out. They involve simple steps which will aid the user in completing their end goal.



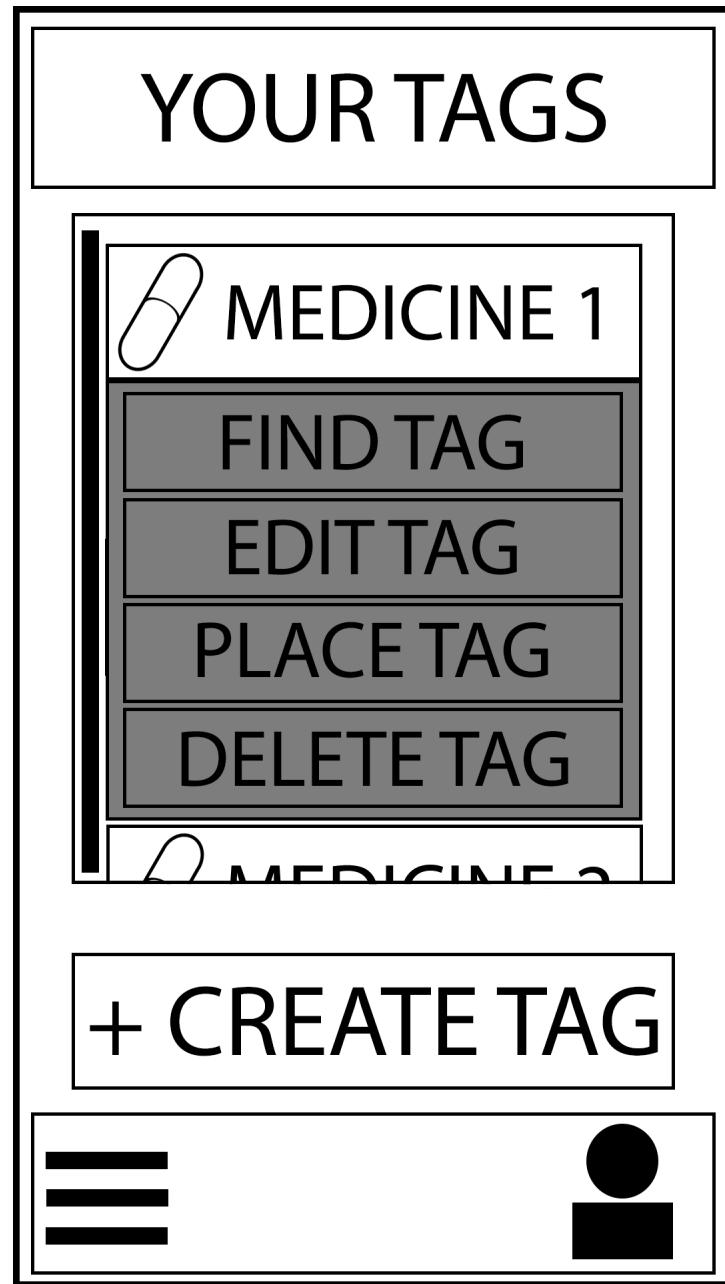
During this sprint we also took the feedback about the initial mock ups and refined the design to more reflect what we want our end product to look like.



The new design takes the core element from the old design (the list of tags) and builds upon that to increase usability. This includes the clear “create tag” button, the navigation bar on the bottom and example tags listed in the scroll list.

In this design when you press a tag it will expand to show the operations you can do on the tag, such as find, edit, place or delete.

This is done to make the UI as simple as possible while also retaining all actions on one screen to help the retention of the user.



Since the tags page is not the sole page of our web app, Aidan also designed mock ups for the other pages (Find, Login, Signup)

AR TAG FINDER

USERNAME

PASSWORD

LOG IN

— OR —

SIGN UP

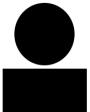


AR TAG FINDER

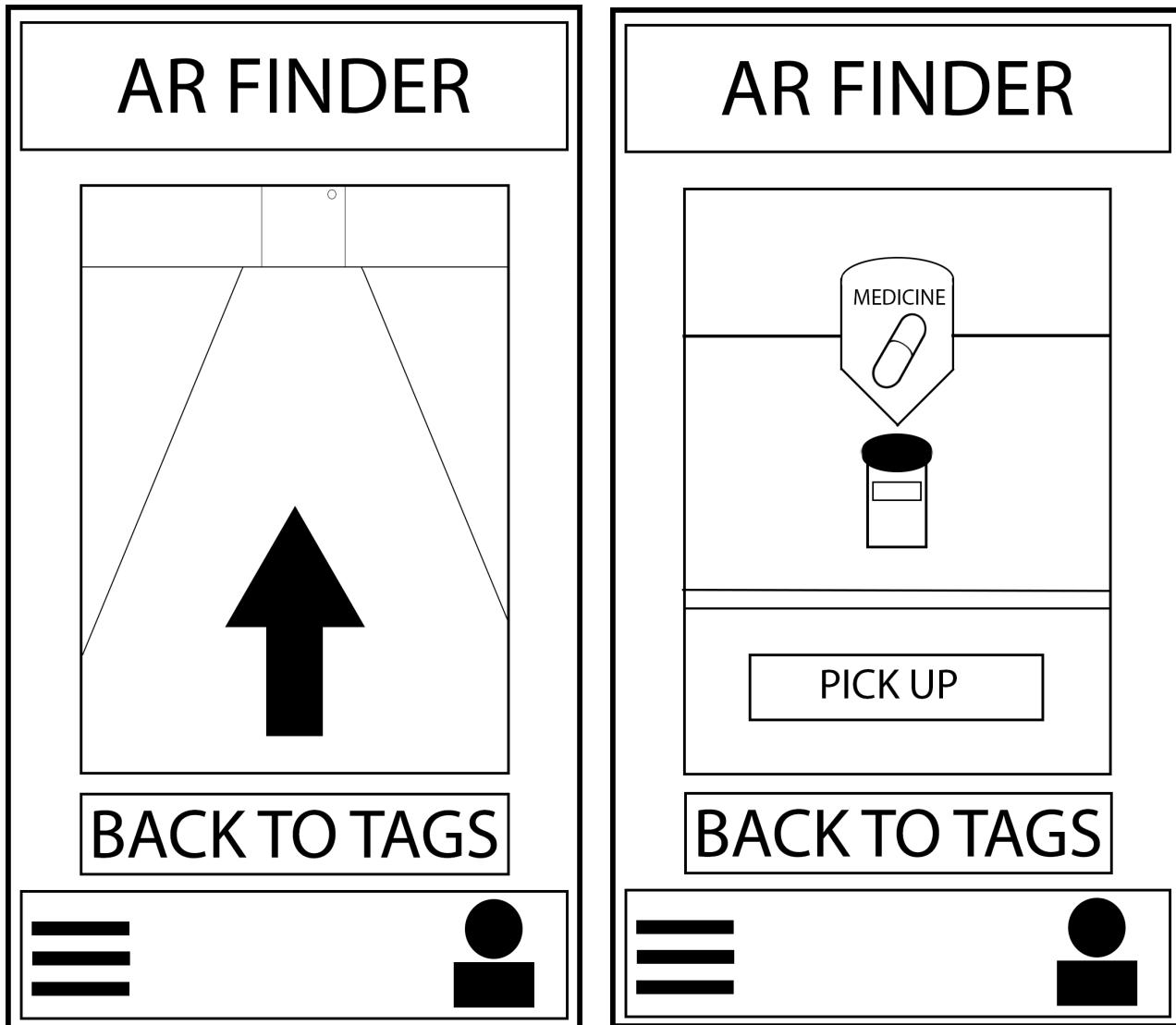
USERNAME

PASSWORD

SIGN UP



The Login & Sign Up Page



The Finder Pager

Left represents looking for the tag using an arrow to guide the user

Right depicts locating for the tag and picking it up

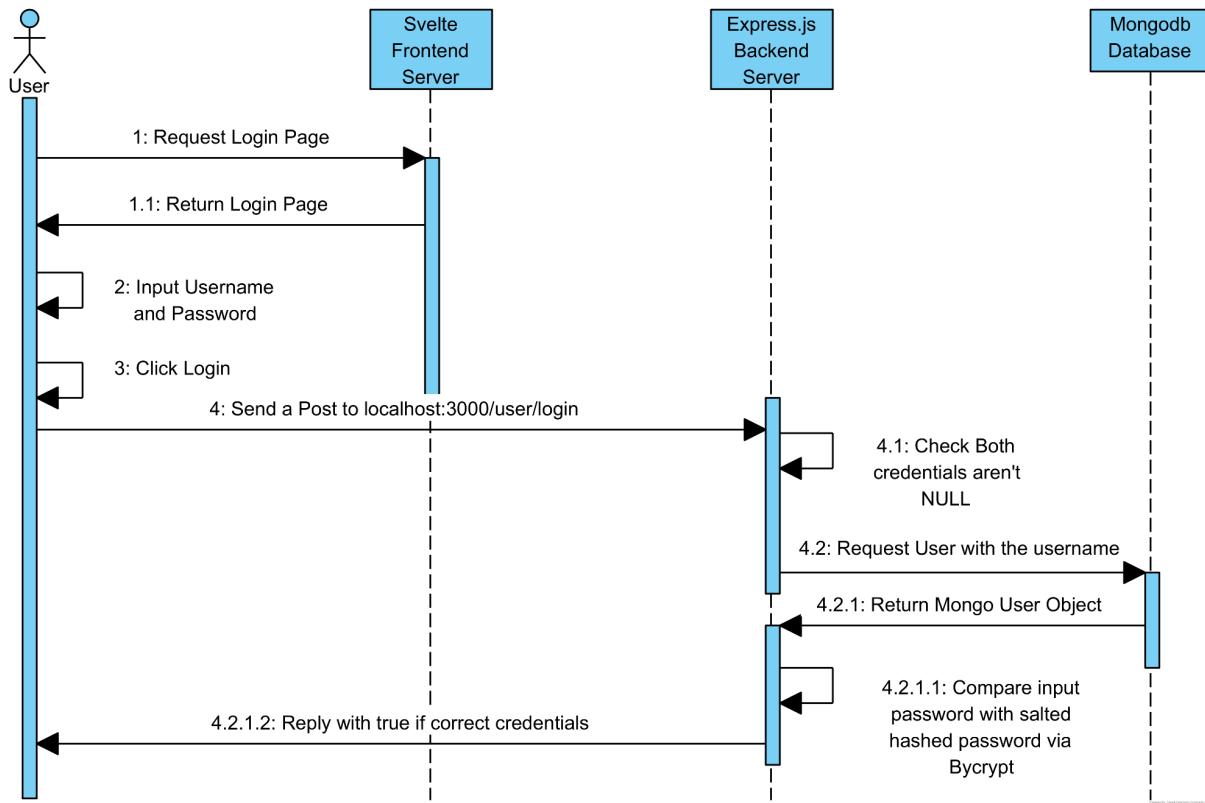
User Back End Team

From the user stories, we quickly found that it would be better to first set up the user backend to a rudimentary degree before starting the tags. Through this, we were able to set up the express.js server to respond to Adding Users, Deleting Users, Updating Users, and a function that tells you if you can login.

We have ensured that every password saved is hashed and salted with bcrypt from Mongodb's best practices such that no plain text password can ever be saved.

We created the User backend functions to create, 'login', update, and delete and also made the framework for encrypting and checking passwords, which we then implemented into the MongoDB framework.

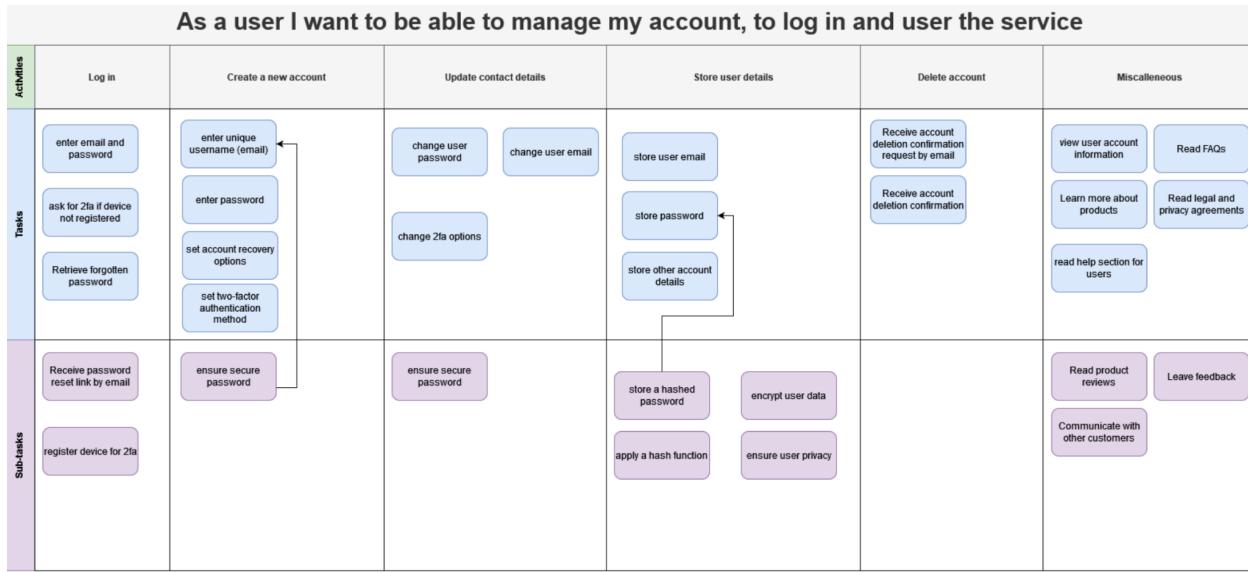
From this, we created a diagram to summarise a scenario using the backend:



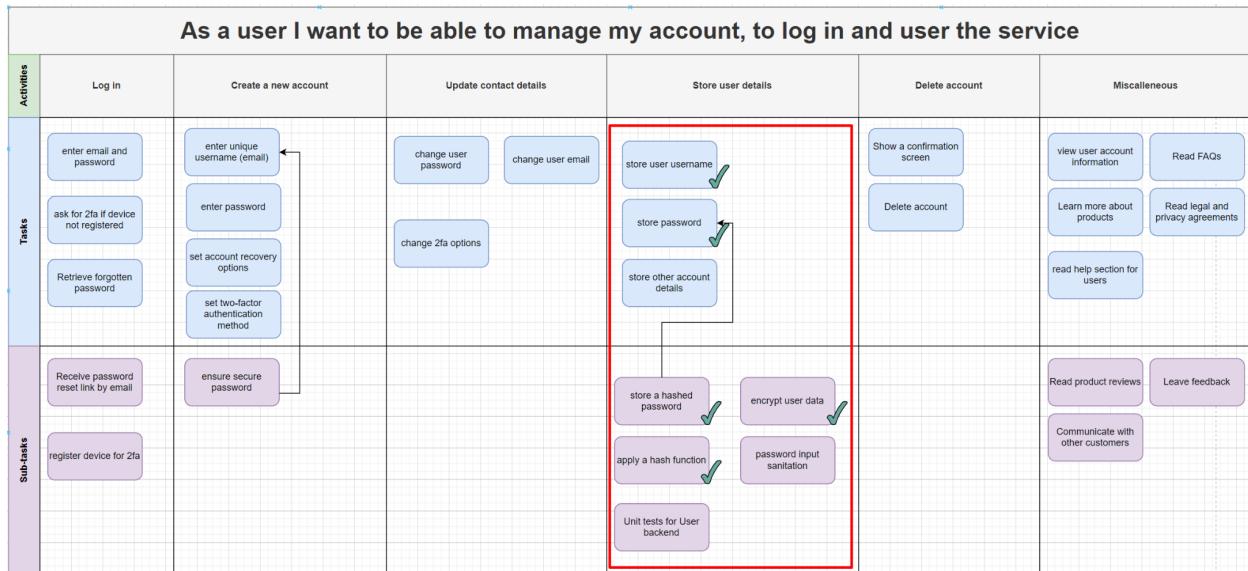
AR Team

This sprint was mainly focused on integrating the AR system Alan had in his own github repo into the node.js website. He also did additional research on APIs to use when doing geolocation, we found out that javascript had built in geolocation support but required a secure HTTP context to use it in.

This entails that the story map for managing accounts from:



Has become:

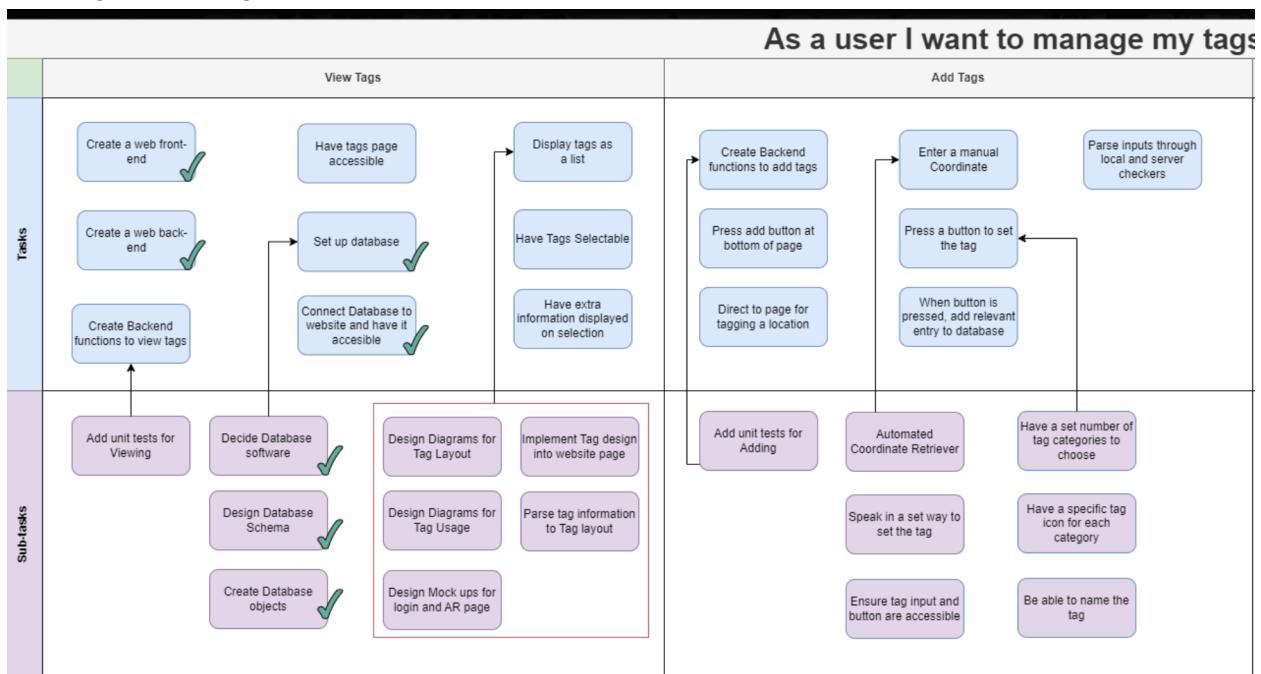


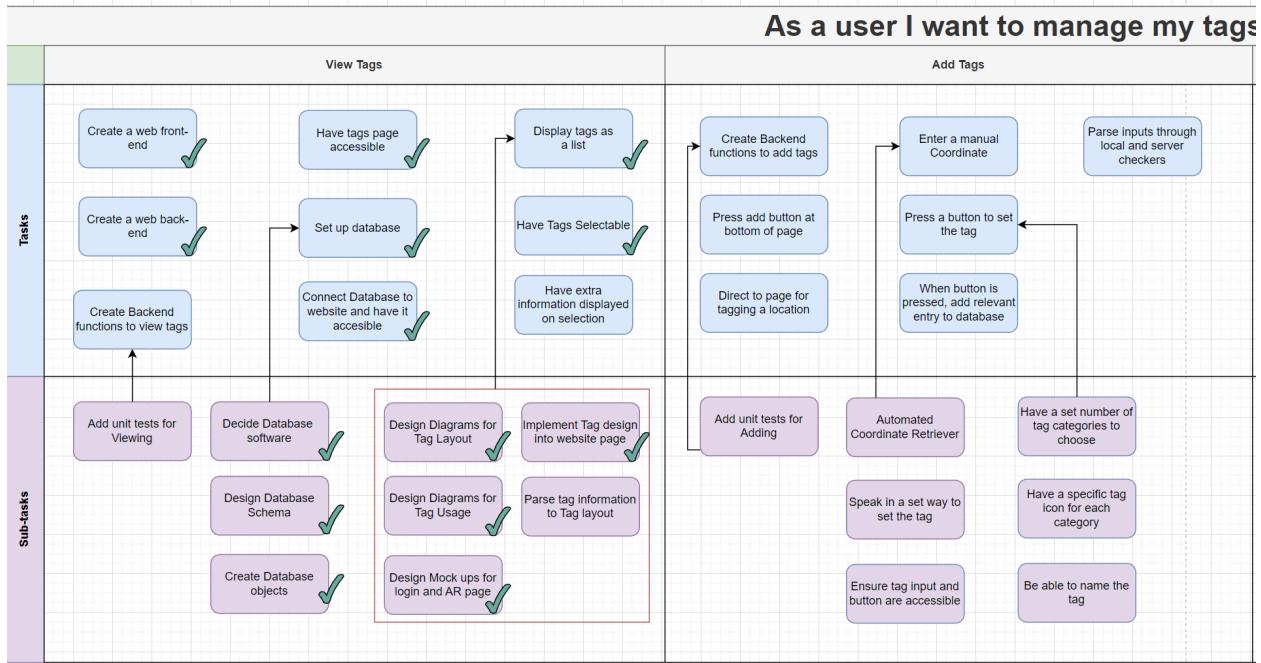
Sprint 6 (14/02/2023-21/02/2023)

For this sprint, we decided to assign specific people to each of the remaining user stories from the previous sprint. Rodion will work on creating the Backend functions for viewing tags, Alfie will work on adding unit tests for these, sanitising passwords in the database. Oliver has been tasked to add unit tests for the User backend once Rodion is done.

For Aidan's work, he has completed all the user stories assigned to him, designing all the mockups for the login and tags page, and how we expect it to function with activity diagrams. These will help a lot with implementing the front end in svelte. Alex and Vishal were also able to create a basic animated svelte implementation of a Tags page that has expanding tags in a list.

Moreso, they have also completed all their user stories. This has caused the story map for the user tag theme to go from:





We have nearly completed an Epic.

From this, we assigned Alex to parsing the tag information in the backend into the frontend and displaying them. We then found that we could assign Aidan and Vishal to create the login page from the user story:

“Enter Email and Password” for the Login Epic.

We were able to create a diagram for the AR system, research how to get the Geolocation from a library, and have stated it works for HTTPS connections.

From this the AR story map has changed from:

As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website Form an icon to float above the exact location ✓ Be able to get geo-location data in longitude and latitude 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located Have the user press a button to state they have reached their location. 	<ul style="list-style-type: none"> Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR Research GeoLocation API Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

To:

As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website Form an icon to float above the exact location ✓ Be able to get geo-location data in longitude and latitude ✓ 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located Have the user press a button to state they have reached their location. 	<ul style="list-style-type: none"> Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR Research GeoLocation API ✓ Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

Sprint 6 Retrospective

Front End Team

The aim of this sprint was to flesh out the svelte pages that we already had:

- We managed to produce a login and sign up page in svelte. These pages were purely a visual layout of the elements to which functionality such as checking inputs against the database were meant to be added. However this was unable to occur and will be attempted in the next sprint.

The image shows two separate login forms side-by-side. The left form is titled "Please Log In" and the right form is titled "Please Sign Up". Both forms consist of two input fields labeled "Name" and "Password", followed by a "Log In" button on the left and a "Sign Up" button on the right. A horizontal dashed line with the word "OR" in the center separates the two forms.

Please Log In

Name

Password

Log In

Please Sign Up

Name

Password

Sign Up

----- OR -----

Sign Up

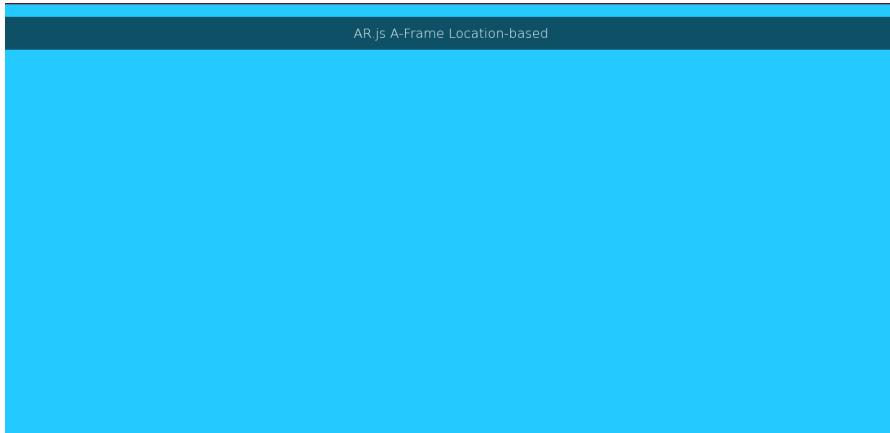
AR Team

This sprint was focused on debugging the AR.svelte page, initial debugging research didn't yield much as the use of AR.js in svelte specifically was undocumented. Fortunately Alan was able to find a git repo containing an example of AR.js used within a svelte project after examining this repo further it was found out the reason ar.js did not work was due to the fact the imported scripts below had to be put between a svelte:head tag instead of regular html head tag.

The git repo in question : <https://github.com/sectorxusa/svelte-aframe-arjs>

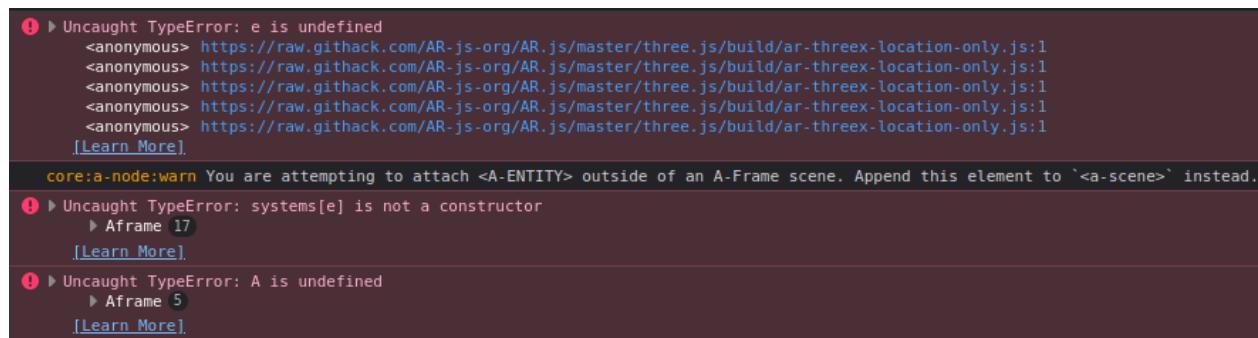
After examining source/routes/index.svelte we found the scripts tag were being placed in svelte:head, further inspiration was also gained from this repo about the possible need to **resize ar.js camera according to page dimensions.**

After implementing the fix, selecting the ar page lead to this appearing:



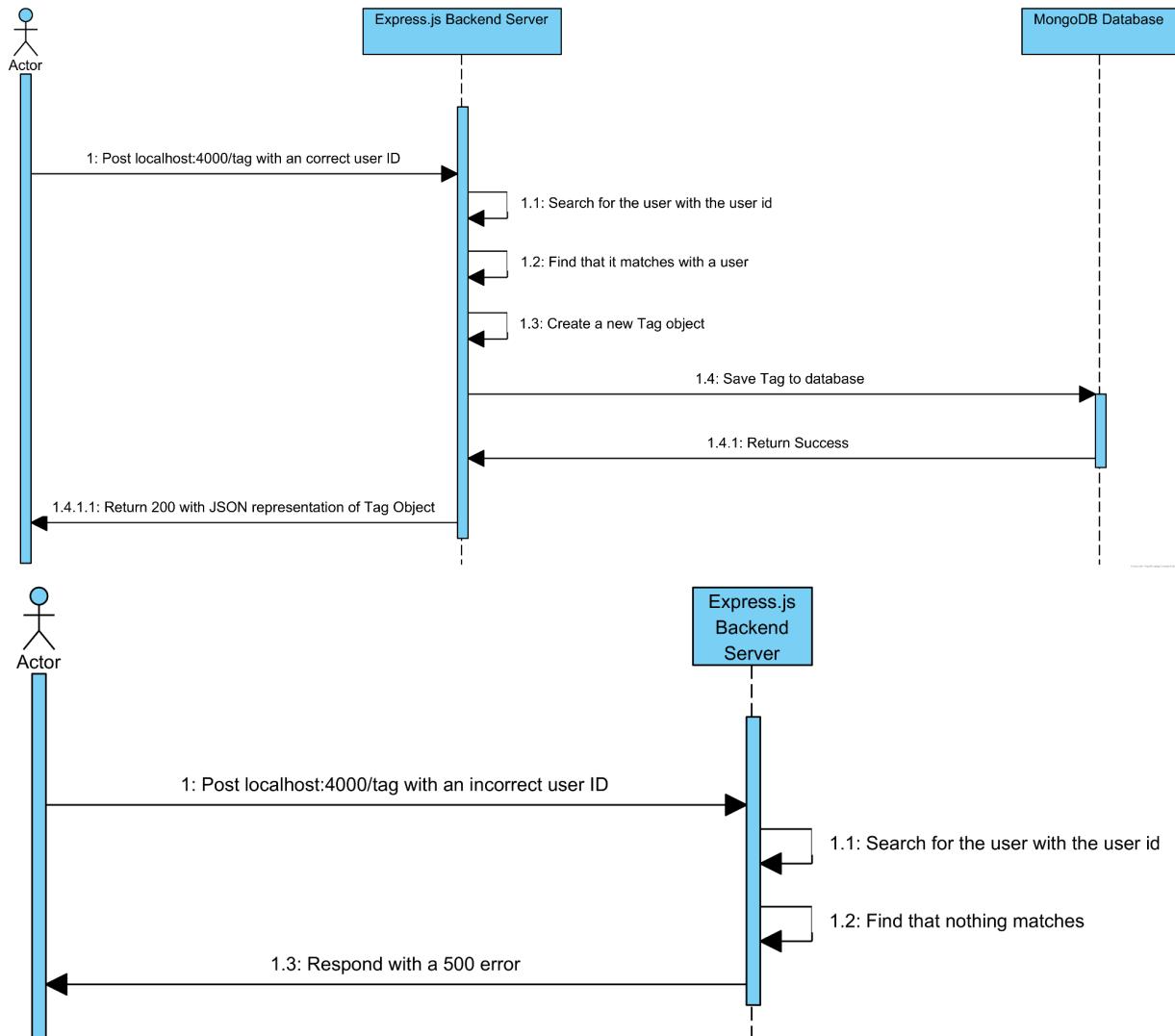
The reason the background is blue is because Alan was testing on his laptop which did not have access to a camera. Further testing will be done this sprint on a phone to see if it fully works. Additionally there are a significant amount of errors to handle, this is

cause for concern for the next sprint.



Backend Team:

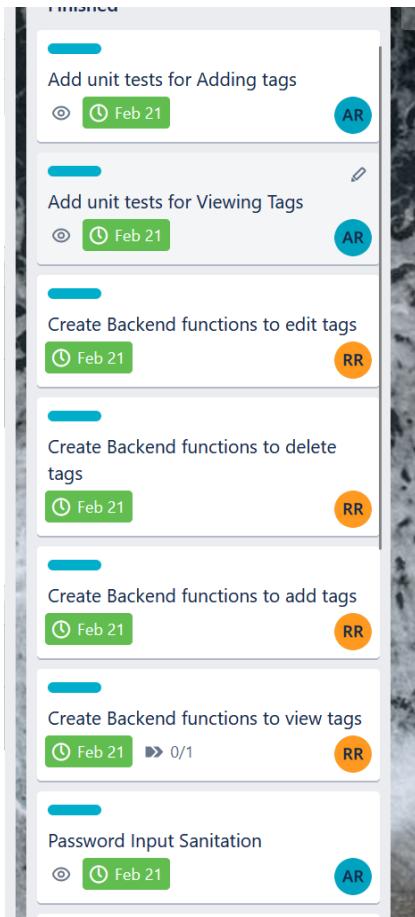
- Created the Tag Backend through modifying the User backend from the previous sprint.
- Formulated a number of unit tests for the Tags backend
- Created password sanitation for the User backend. Due to IBM architecture it took some time to make the mocha test framework.



These two diagrams outline some of the unit tests that were done for the Tag backend.

Sprint 7 (21/02/2023-28/02/2023)

For this sprint, most of the tasks will have to be carried over. The finished tasks from this sprint were:

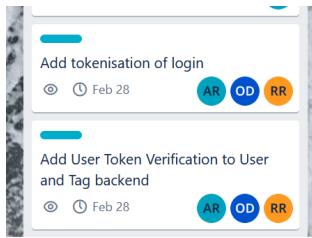


From this, a serious part of the backend will now be set to be implemented, which is Cookie Login Tokenization, which:

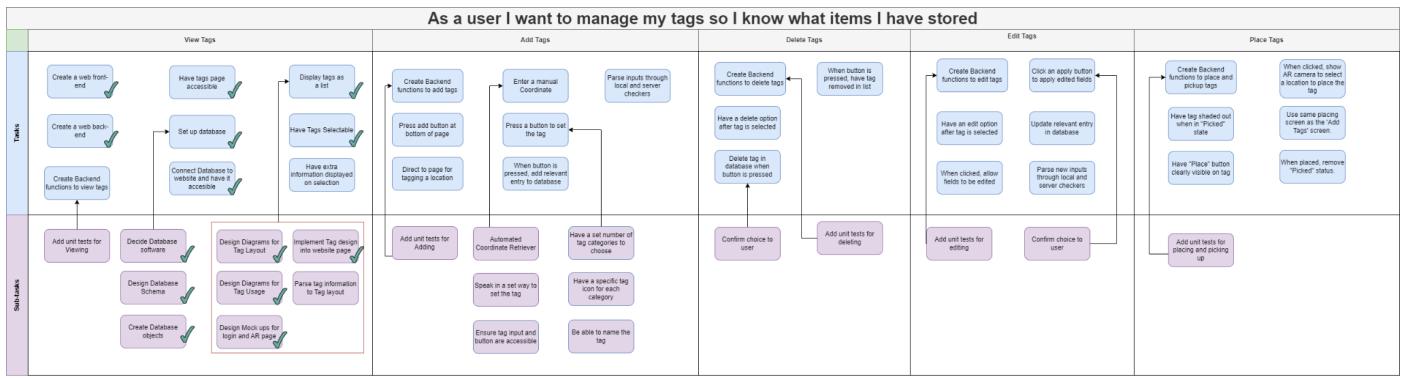
- allows the ability to login and save a cookie of some token that lets the website stay logged in.
- is vital to create a coherent backend. Currently, every function is able to be called by anyone, which needs to change as users should only be able to add tags for their own self, not for other people. Delete calls can delete any user, etc.

This token method will allow the frontend to send requests holding said token and thus allow the backend functions to authenticate. This is much more secure compared to just storing a plaintext password in the browser, thus our reason to choose this.

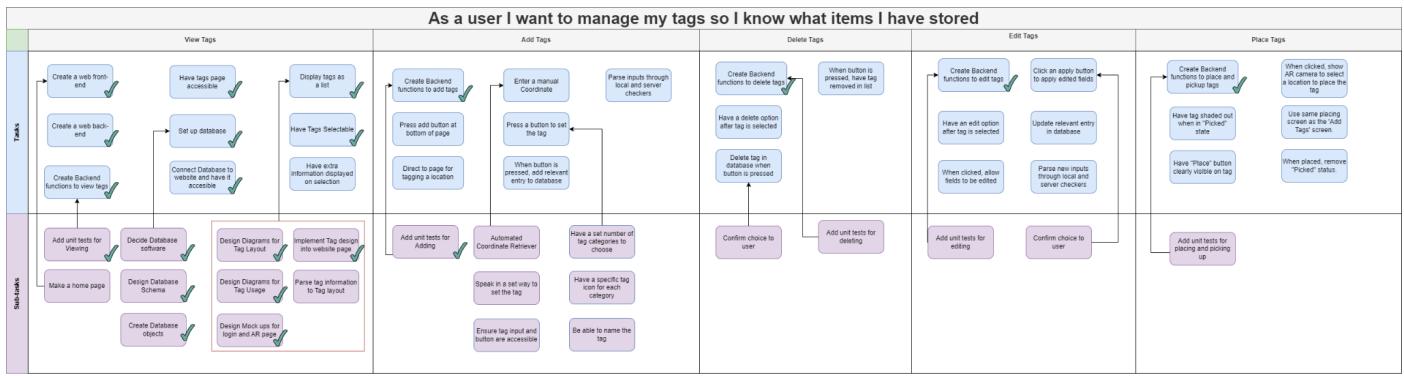
Thus, the new user tasks assigned are:



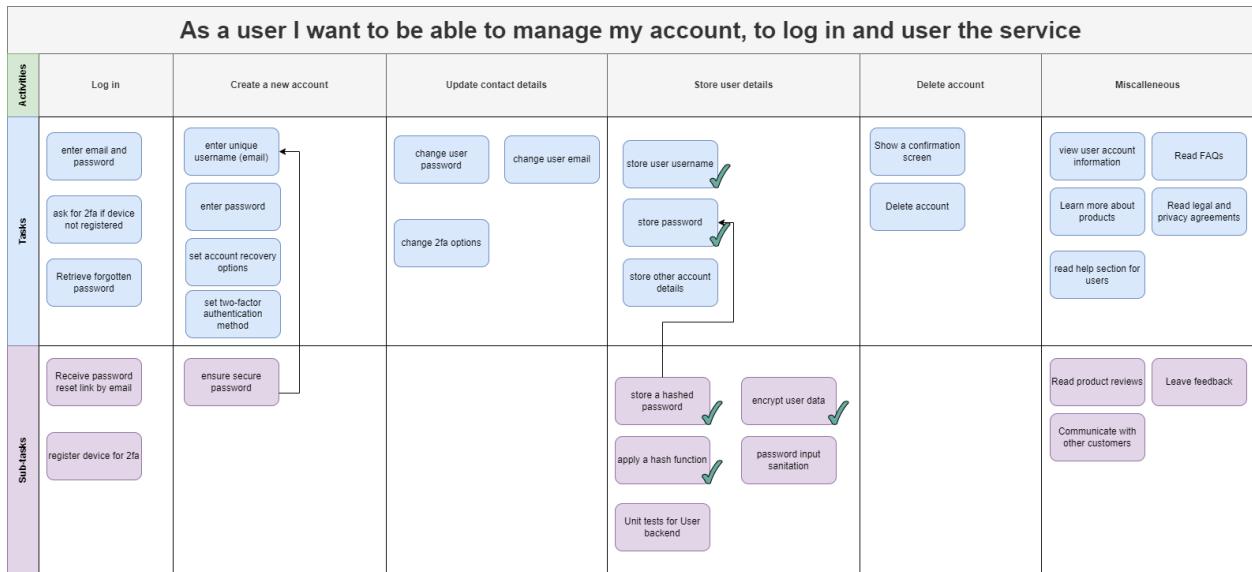
So, to summarise, our story maps have gone from:



To:

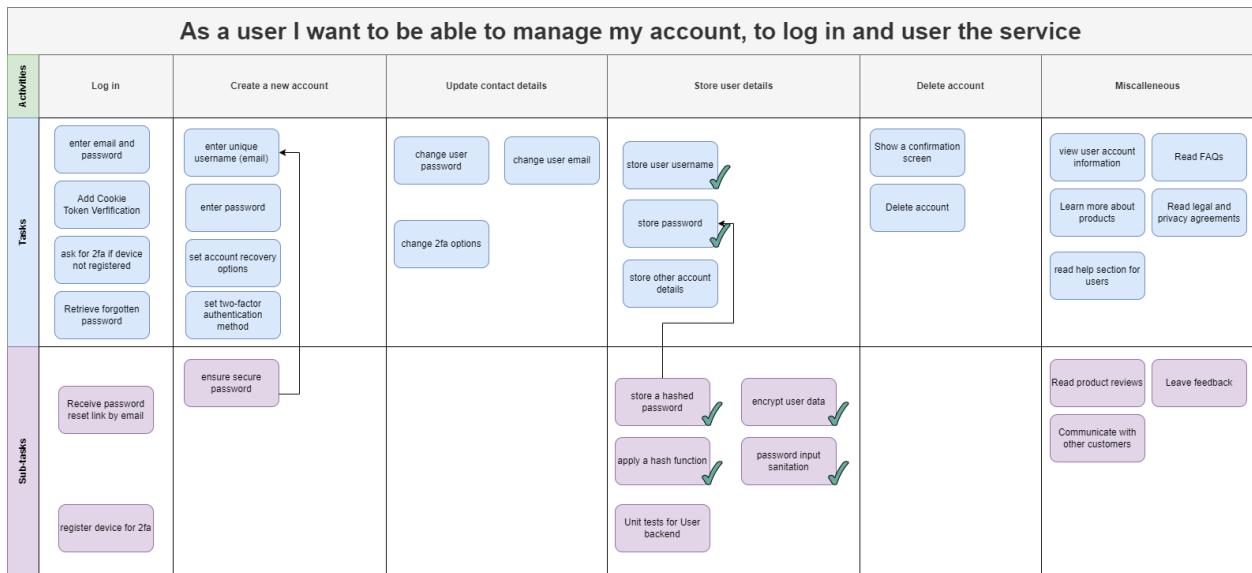


And with the Account theme:

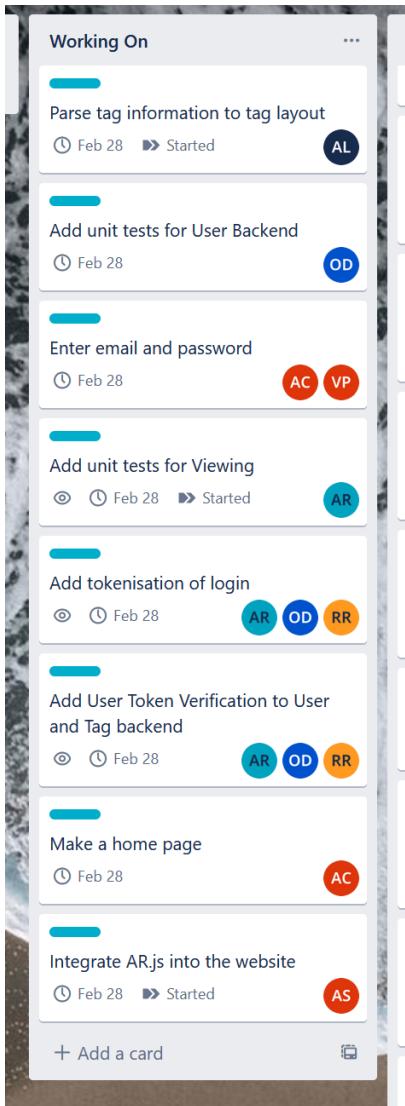


A story map example in a cross-functional flowchart shape (Misc shape library)

To:



For this sprint, the assigned stories are:



It is mostly identical to the previous sprint. We deemed it necessary to speed up.

Sprint 7 Retrospective

Front End Team

Our aim this sprint was to implement the login functionality that we were unable to implement the previous sprint.

- We managed to build a fetch API post request including the inputted name and password to query the database.
- Upon the query the database would return either true or false depending on whether that user existed and the password matched.

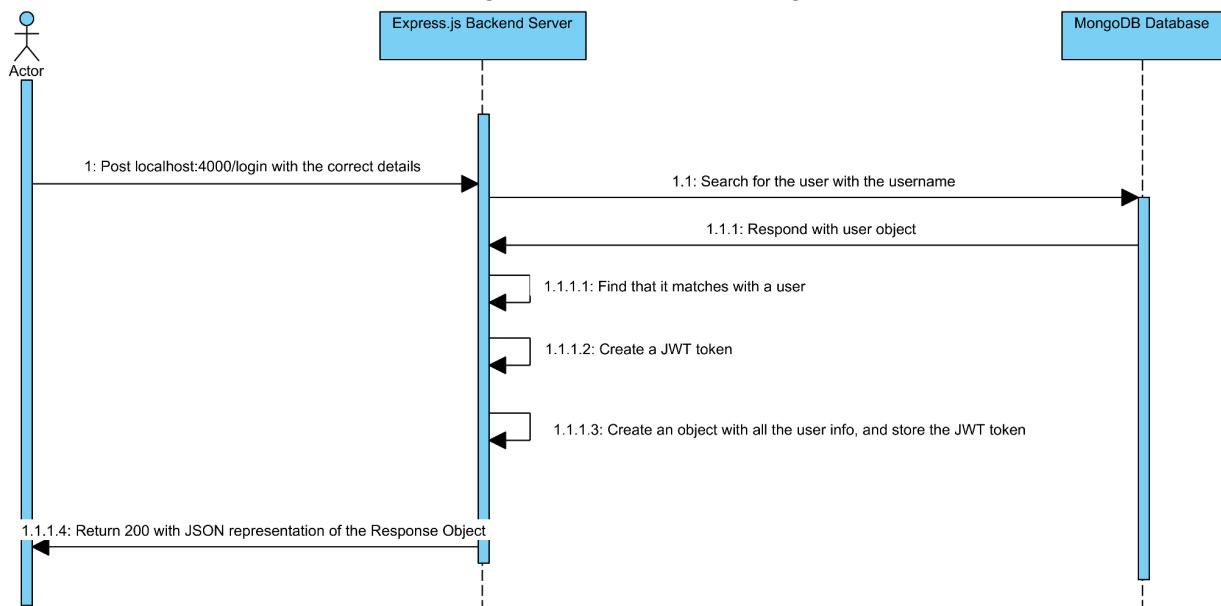
We also implemented a rudimentary home page with buttons linking to the tags, login, signup and AR page. Currently only the links to tags and login work.

The AR link will be included when the AR and Front End branches are merged to main and the AR svelte page becomes accessible to the front end team.

Backend Team

The aim for the sprint was to add tokenization to the login functionality. Moreso, we wanted a method for the server to create a ‘token’ that can verify someone is logged in as someone.

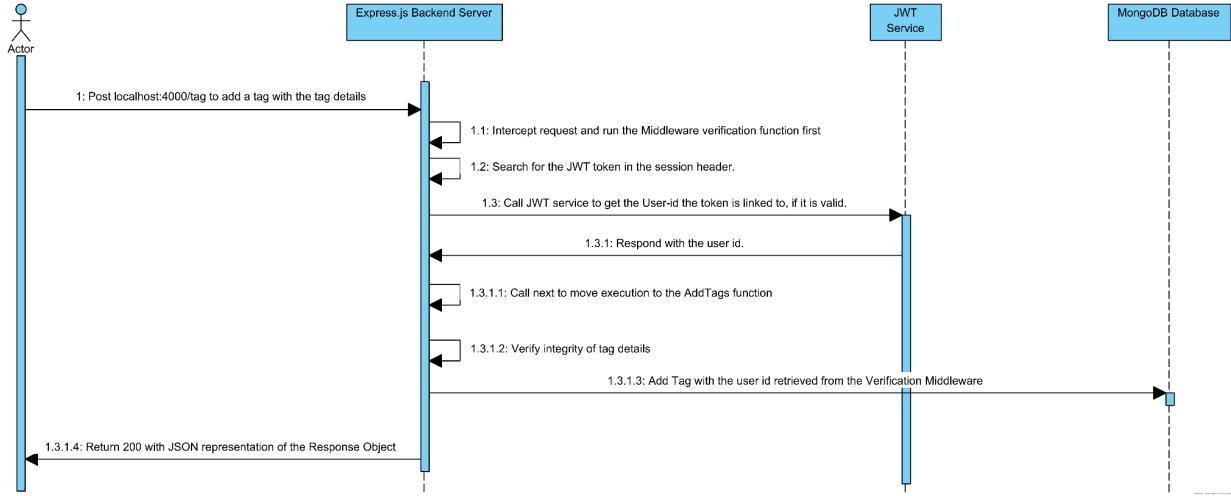
We were able to implement it, with the general framework being:



The JWT token is part of express.js, and handles all the annoying details for us. This architecture will thus allow us to be able to store tokens in cookies and verify server requests easily.

We have also created the middleware to verify the token is in the request and valid. This function is something called in between the true function, such as if we wanted to add a tag, we would need to verify that the user is logged in, and if they are, get the userID they are logged into (from the tag), from which we can then add the tag specified. This guarantees that every request with the middleware cannot be done without explicit authentication via the tokens.

This is surmised here:



When posting, the authentication tokens stored in the cookies, which we will implement next sprint, will be automatically included in all requests.

In summary, with the story maps. For the Tags Theme, it went from:



To:



And for the account theme, it went from:

As a user I want to be able to manage my account, to log in and user the service						
Activities	Log in	Create a new account	Update contact details	Store user details	Delete account	Miscellaneous
Tasks	enter email and password Add Cookie Token Verification ask for 2fa if device not registered Retrieve forgotten password	enter unique username (email) enter password set account recovery options set two-factor authentication method	change user password change user email change 2fa options	store user username ✓ store password ✓ store other account details	Show a confirmation screen Delete account	view user account information Read FAQs Learn more about products Read legal and privacy agreements read help section for users
Subtasks	Receive password reset link by email register device for 2fa	ensure secure password		store a hashed password ✓ encrypt user data ✓ apply a hash function ✓ password input sanitization ✓ Unit tests for User backend		Read product reviews Leave feedback Communicate with other customers

To:

As a user I want to be able to manage my account, to log in and user the service						
Activities	Log in	Create a new account	Update contact details	Store user details	Delete account	Miscellaneous
Tasks	enter email and password ✓ Add tokenisation of Login ✓ Add Cookie Token Verification ✓	Create a Sign-up page frontend set two-factor authentication method	change user password change user email change 2fa options	store user username ✓ store password ✓ store other account details	Show a confirmation screen Delete account	view user account information Read FAQs Learn more about products Read legal and privacy agreements read help section for users
Subtasks	Receive password reset link by email register device for 2fa	ensure secure password		store a hashed password ✓ encrypt user data ✓ apply a hash function ✓ password input sanitization ✓ Unit tests for User backend		Read product reviews Leave feedback Communicate with other customers

Note the 'Add tokenization of Login' was not added last sprint due to a technical fault. 2 user stories were completed in the account domain.

Sprint 8 (28/02/2023-07/02/2023)

For this sprint, the assigned stories are:

Add unit tests for User Backend	⌚ Mar 7	AR OD RR
Add User Token Verification to User and Tag backend	⌚ Mar 7	AR OD RR
Integrate AR.js into the website	⌚ Mar 7 ➡ Started	AS
Create Design for Tags and AR page	⌚ Mar 7	VP
Create a new account	⌚ Mar 7 ➡ 0/3	AC AL
Create a Sign-up page frontend	⌚ Mar 7 ➡ Started	AC
enter unique username (email) for signup	⌚ Mar 7 ➡ Started	AL
enter password for signup	⌚ Mar 7 ➡ Started	AL

Sprint 8 Retrospective

Backend Team

- Completed the backend token verification user stories. This means that every route in the backend now requires the user to be logged in, as they have to send a JWT token via a cookie they get after they login.
- This token can be verified, and if real will return the UserID the token was created from,

The screenshot shows a Postman interface with the following details:

- Method:** POST
- URL:** localhost:3000/tag
- Body:** JSON (selected)
- Body Content:**

```
1
2   "tagName": "testertag",
3   "coords": 123,
4   "placed": true
```
- Response Body:** JSON (selected)
- Response Content:**

```
1
2   "data": {
3     "tagName": "testertag",
4     "coords": 123,
5     "placed": true,
6     "owner": "63fe1edaa83e6f2bc7640b88",
7     "_id": "640756b049f684017b8fea7d",
8     "__v": 0
9   }
```

Although no owner ID is specified, the id is retrieved from the 2 cookies you get after logging in:

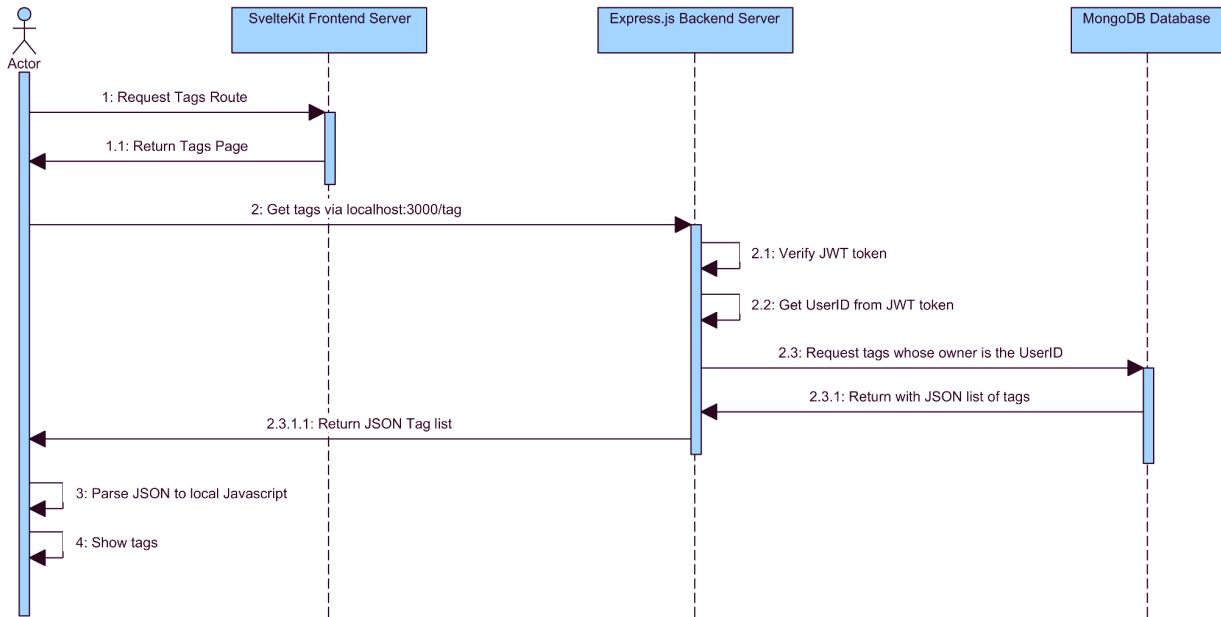
Name	Value	Domain
ar-session	eyJ0b2tlbi6I... ...	localhost
ar-session.sig	Ap6ANeSj2rk	localhost

This essentially guarantees users can only create tags if they are logged in, and such a tag will always be linked to them.

This idea applies to all routes, such as deleting tags, or updating tags. You can only update tags you own, and this is verified by using the session token to get the userID that the user is logged in as. The token acts as a password to verify who they are.

We found that updating the `getTags` route to only show the tags the user owns was propagated to the tags page with little effort to get working, so we had completed that user story as well.

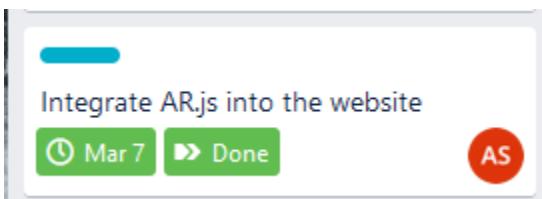
An example can thus be shown. This diagram assumes the user has already logged in and has a JWT token cookie:



AR Team

For the AR.js team, Alan was able to finally get the ar.js page to work with reliability. This means that when going to the ar page, it will display a red cube in a specified location. We won't worry about changing what this looks like for now, as functionality is most important.

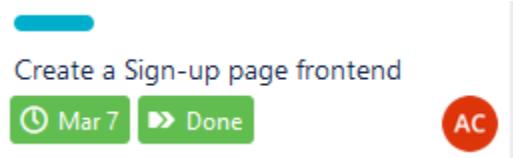
So:



Was complete.

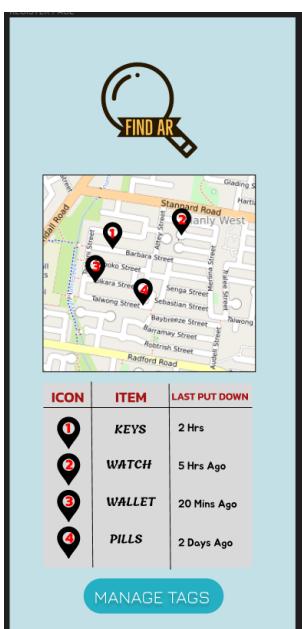
Front-end Team

For the frontend team, there seemed to be some issues with the sign up page, but the design portion of it was completed by Aidan, so all that's left is for Alex to integrate the backend call to add a user from the forms inputs.

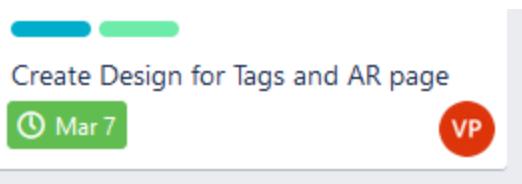


So, this was completed.

Vishal was able to complete some diagrams on the tags page, as some more views on how it would look like.



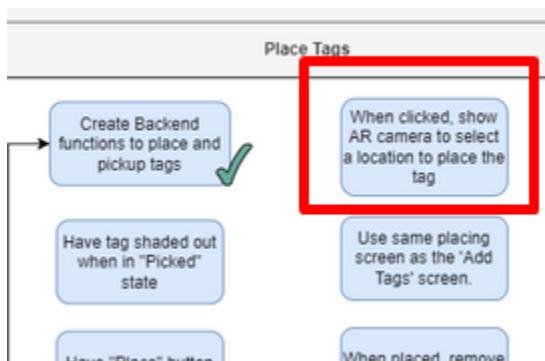
We felt the map would probably be unneeded, as the accuracy is low to a house level. But, the general flow was positive.



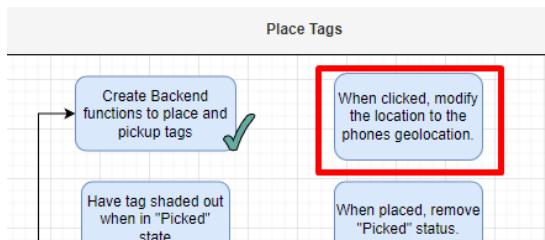
So, this was completed.

We also found a discrepancy in the user stories, one where placing a tag entailed entering AR to do so.

We discussed that there wouldn't be much point doing that, as we could just get the phone's GPS and place it there. AR would provide no benefit because there would be nothing to find when placing a picked up tag. So, we changed:



To:

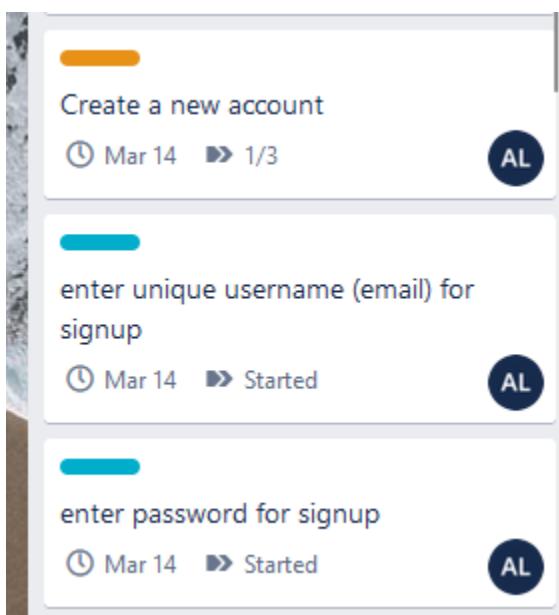
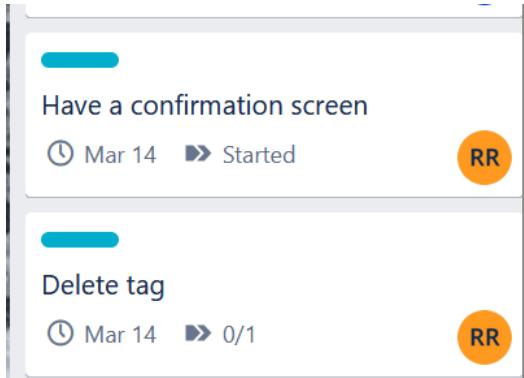


Sprint 9 (07/03/2023-14/03/2023)

From this sprint we have essentially completed nearly all pivotal backend routes. You can add, delete and update users and tags. Such operations are all verified, such as a user only being able to delete themselves, or a user can only edit tags they own.

From this, we can now work completely with the frontend.

These were the assigned tasks for this sprint:



Have an edit option after tag is selected

⌚ Mar 14 ➡ Started

AR

When clicked, allow fields to be edited

⌚ Mar 14 ➡ Started

AR

Press add button

⌚ Mar 14 ➡ Started

AC

Have a pop-up to specify tag details

⌚ Mar 14

AC

Have tag shaded out when in "Picked" state

⌚ Mar 14 ➡ Started

AS

Have "Place" button clearly visible on tag

⌚ Mar 14 ➡ Started

AS

When clicked, modify the location to the phones geolocation.

⌚ Mar 14

AS

Iterate Tag Page design

⌚ Mar 14

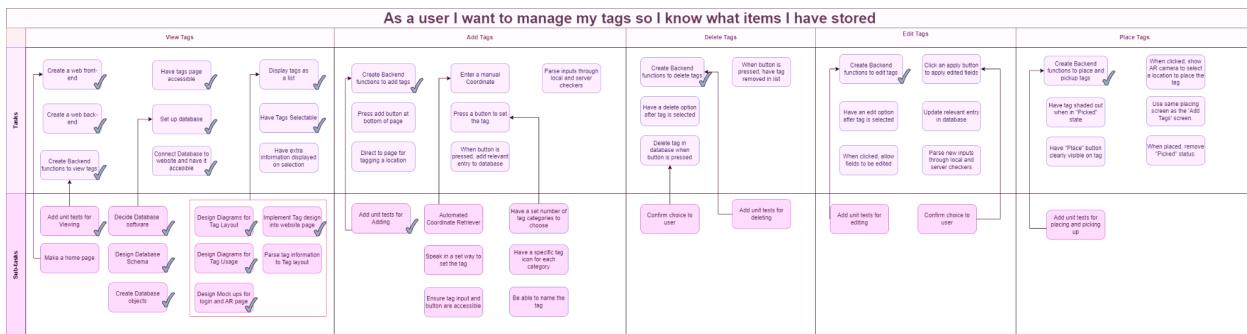
VP

Add unit tests for User Backend

⌚ Mar 14



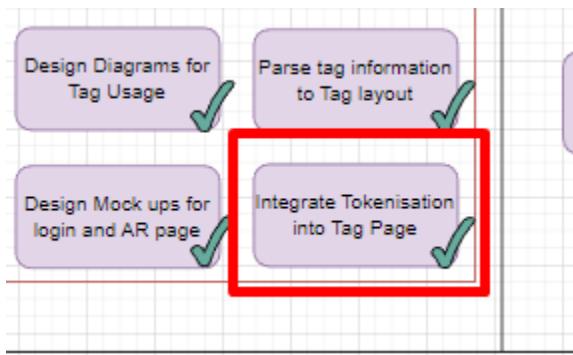
From this, for the Tag Theme, it went from:



To:



'Add Token Verification' was added to every epic to account for the "Add User Token Verification to User and Tag backend" user story, as that encapsulates adding token verification to every one of the routes as shown there.



This user story was added as it wasn't in the story map from last week.

For the AR.js story map, it went from:

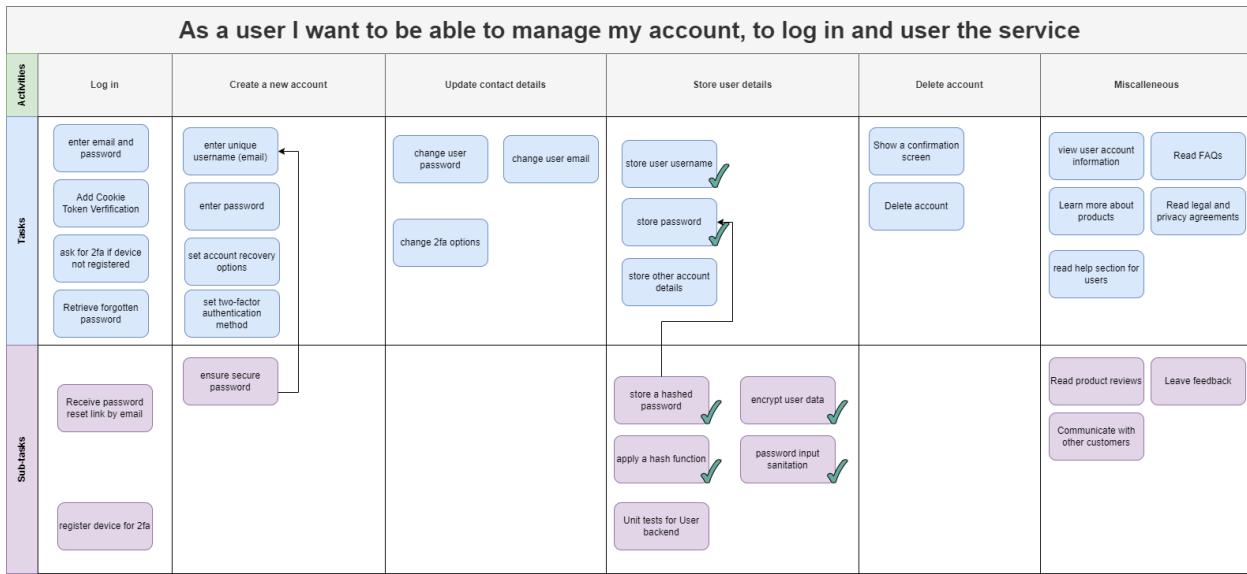
As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website Form an icon to float above the exact location ✓ Be able to get geo-location data in longitude and latitude ✓ 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located 	<ul style="list-style-type: none"> Have the user press a button to state they have reached their location. Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR ✓ Research GeoLocation API ✓ Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

To:

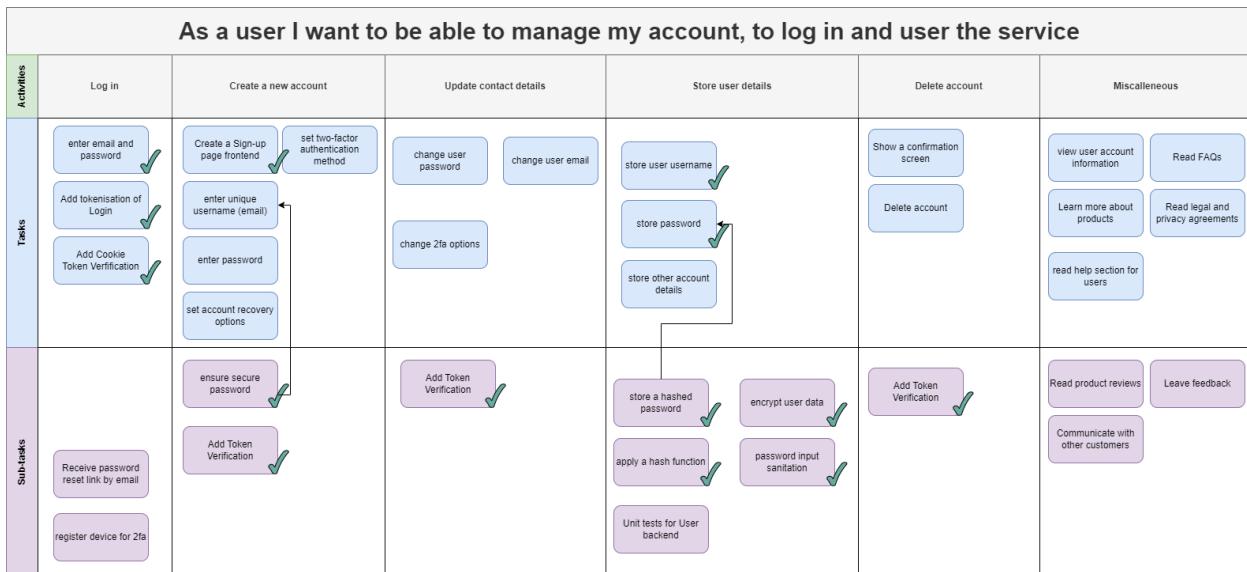
As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website Form an icon to float above the exact location ✓ Be able to get geo-location data in longitude and latitude ✓ 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located 	<ul style="list-style-type: none"> Have the user press a button to state they have reached their location. Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR ✓ Research GeoLocation API ✓ Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

Thus, the first epic is done. The second epic is mostly complete from the virtue of AR.js including those features, but we will leave it for now.

For the account story map:



To:



The same idea of token verification is added to every backend route here. The sign up frontend is also ticked off.

Sprint 9 Retrospective

Front-end Team

Completed the editing of tags. This means that you can now use the frontend UI to edit any tag owned by the user.

The screenshot shows a list of tags on the left and an edit modal on the right. The tags include:

- (Description) Find Place Edit Delete
- coords tag
- test2as
- tester
- test3
- test4ss
- Defaultssssss
- tre22
- tre2223
- tre223452
- tre2224
- testing
- testingss
- tre2224ss
- tag here
- Default

Edit Tag Name: test1

Edit Tag Description: Empty

Edit Tag Coordinates

- Latitude: 745.21
- Longitude: 0.2547005
- Submit
- Cancel

Create Tag +

The design for that is not complete, as we are just focusing on functionality at the moment. This entails more than a few user stories were completed, these being:

Update relevant entry in database after applied
⌚ Mar 14 ➡ Done AR

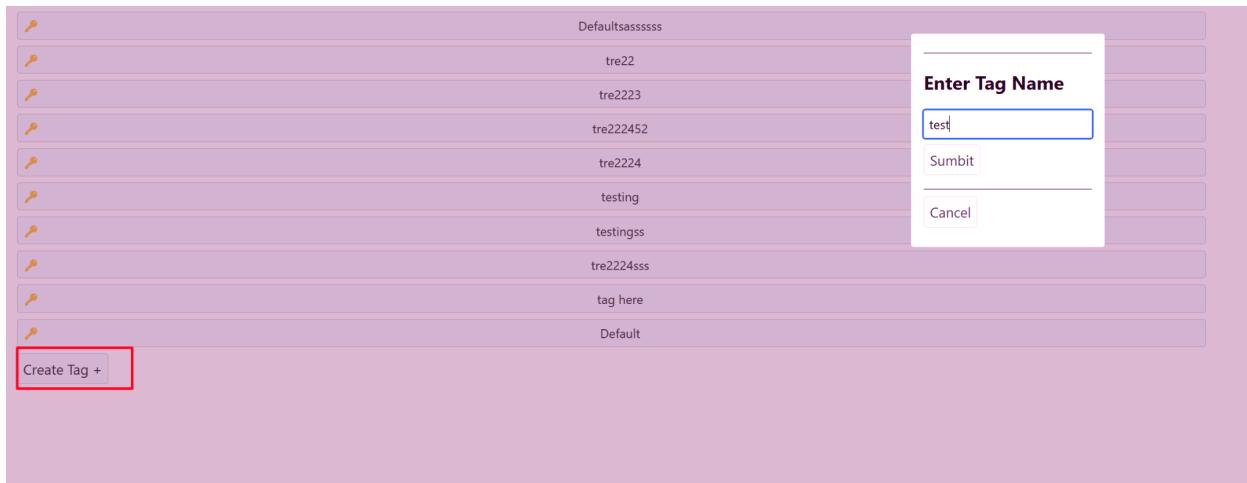
Click an apply button to update tag
⌚ Mar 14 ➡ Done AR

Have an edit option after tag is selected
⌚ Mar 14 ➡ Done AR

When clicked, allow fields to be edited
⌚ Mar 14 ➡ Done AR

A couple tasks outside the scope were completed, so it made up 4 complete tasks.

For adding tags, the assigned tasks were completed, and a few more:



Once again the UI is just for functional demonstration.

Aidan thus completed these user stories:

The image displays a vertical list of six user stories, each with a green button for due date, a progress bar, and a red "AC" button indicating completion.

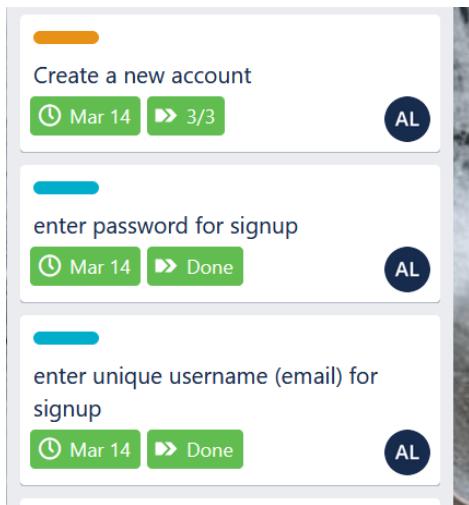
- Be able to name the tag
Due: Mar 14, Progress: Done, Status: AC
- When tag add button is pressed, add relevant entry to database
Due: Mar 14, Progress: Done, Status: AC
- Press a button to set the tag
Due: Mar 14, Progress: 1/3, Status: AC
- Have a pop-up to specify tag details
Due: Mar 14, Progress: Done, Status: AC
- Direct to popup for tagging location
Due: Mar 14, Progress: Done, Status: AC
- Press add button
Due: Mar 14, Progress: Done, Status: AC

We also completed the sign up page after last week's issue,

Please Sign Up

Sign Up

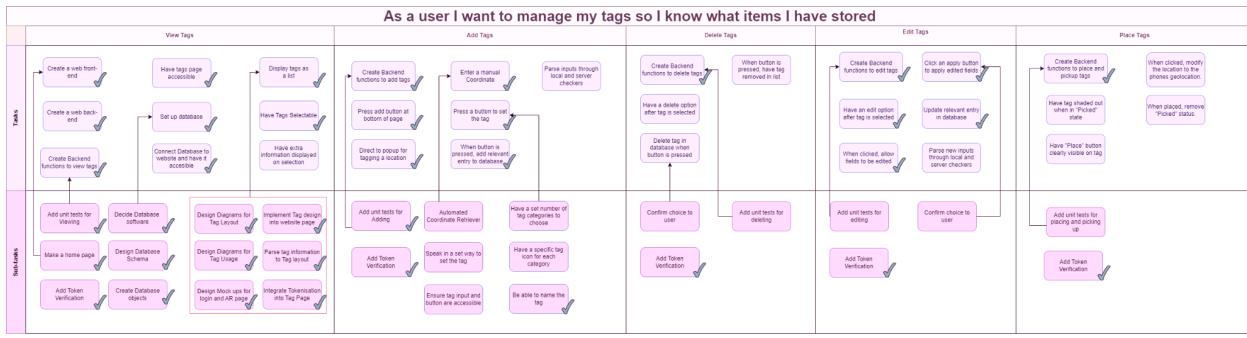
It is just for functionality, but you can now create users of the 'USER' Role. Therefore, the completed user stories were:



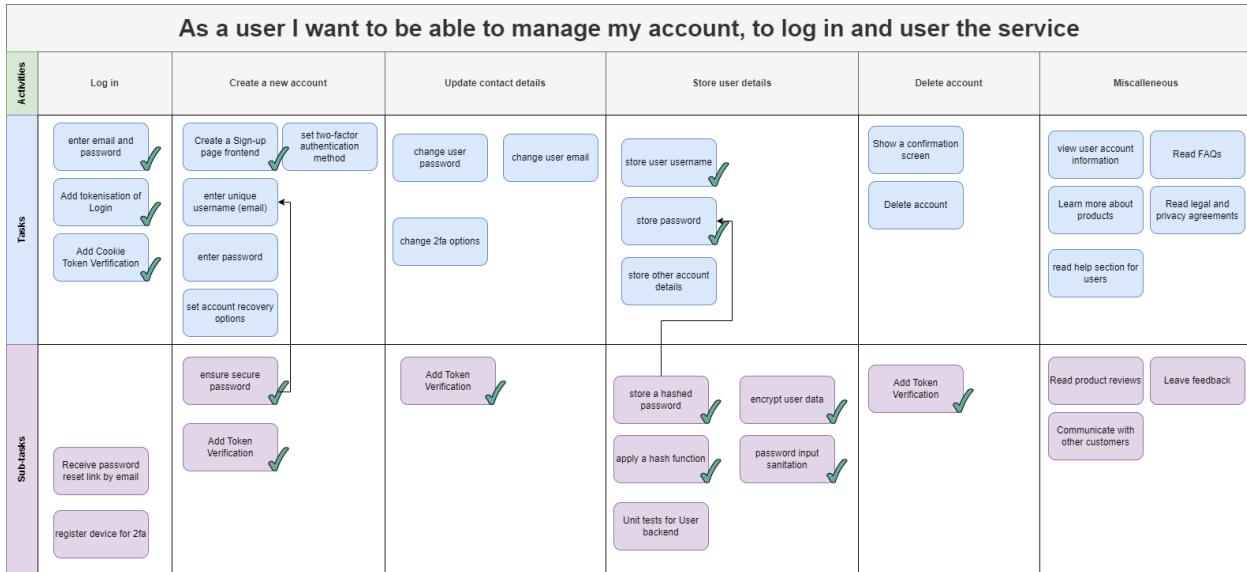
For the tags story map, it went from:



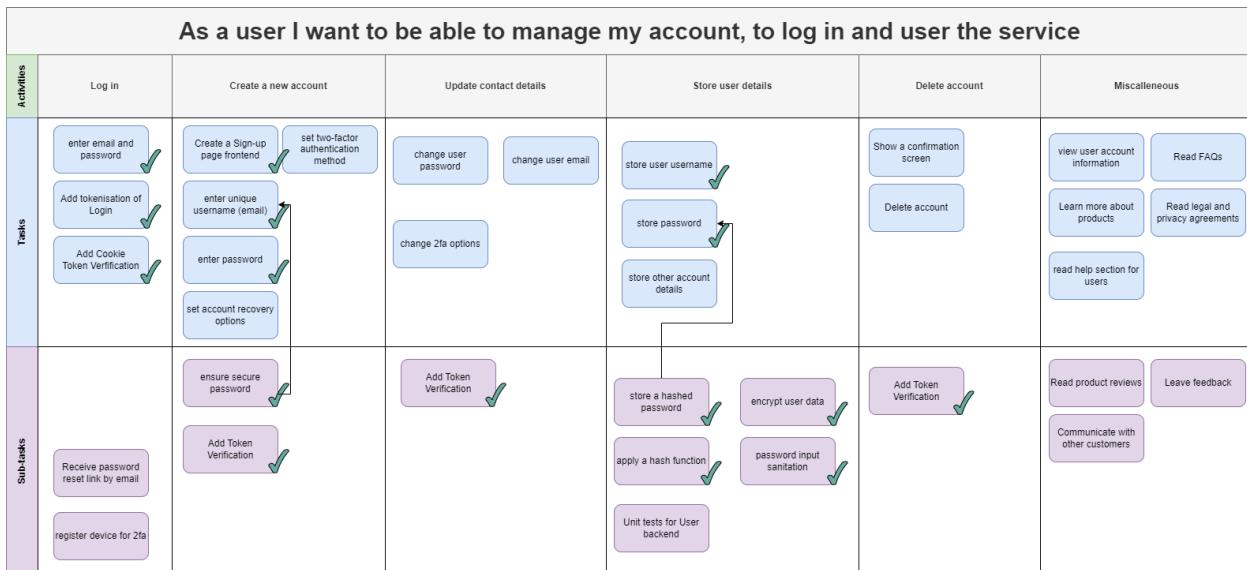
To:



And for accounts, it went from:



To:

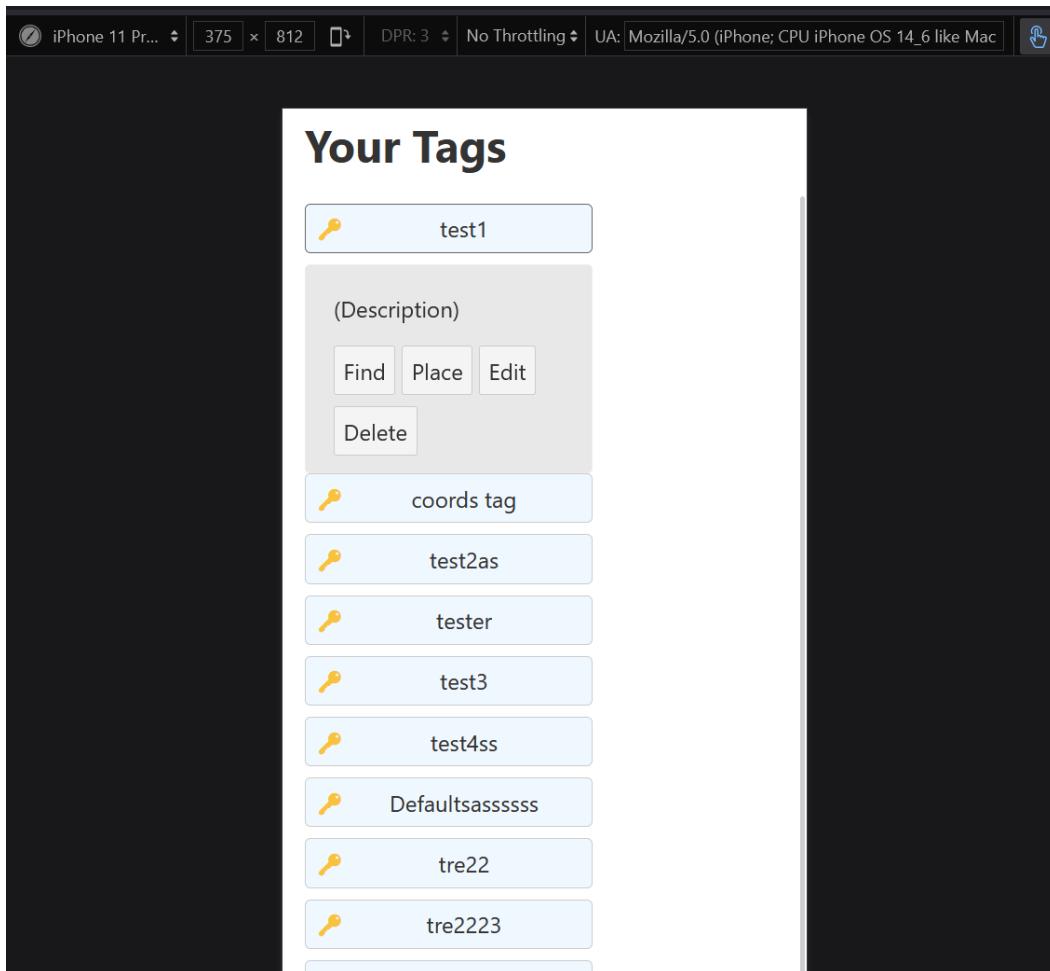


Sprint 10 (14/03/2023-21/03/2023)

We found an issue with the backend for the work to set up a system to place tags. This was fixed quite quickly, but it did delay getting the ability to place tags.

We were unable to complete the delete tag user stories, as we found a rather steep learning curve for the team members moving to the frontend from the backend. Some discussions on how we should go about communicating with the backend were done in the retrospective to bring others up to speed with the task, and we now feel comfortable to finish it this week.

The structure of the pages were not properly taking advantage of a phones small screen size, so we assigned tasks to focus on the tags page, which has a number of issues on that front, for example:



They do not take up all the width space, and the expansion is way too small for a phone's interface. Although we have some designs, we need a more explicit implementation to suit a phone user's needs.

Therefore, for sprint 10, everyone has been assigned:

Add unit tests for User Backend	⌚ Mar 14	OD
Have a confirmation screen	⌚ Mar 21 ➡ Started	RR
Delete tag	⌚ Mar 21 ➡ 0/1	RR
Have tag shaded out when in "Picked" state	⌚ Mar 21 ➡ Started	AR
Have "Place" button clearly visible on tag	⌚ Mar 21 ➡ Started	RR
When clicked, modify the location to the phones geolocation.	⌚ Mar 21	RR
Enter a manual coordinate	⌚ Mar 21 ➡ 0/3	AC
Have a set number of tag categories to choose	⌚ Mar 21 ➡ Started	AC
Have a specific tag icon for each category	⌚ Mar 21 ➡ Started	AC
Set a description for the tag	⌚ Mar 21	AC

Have a button to pickup a tag when near it	AS
When clicked, put the tag into "Picked" status	AS
Create a cohesive CSS design	VP
Make Tags Page mobile friendly.	AL

Sprint 10 Retrospective

In this sprint we were able to finish nearly all pivotal functionality. We were also starting to make it more mobile friendly. So we:

- Were able to properly scale the tag's page such that it took up the maximum space to make it comfortable to use, with some css.
- Made the find button function. Ie, when you click find on a tag, it will redirect you to the AR page and show where the tag is in AR
- Made the AR page have a parameter to specify what tag to get for its coordinates.
- Implemented setting description, and some icons in the form of emojis.
- Implemented the delete tag button with a confirmation screen, and have the place button set the coordinates to the phone's geo-location. This is important, as this method will be used in the add tag method so you don't have to set manual coordinates.
- Made the 'Place' button disappear when the tag is placed, as you should only be able to find a placed tag.
-

An extra user story was noted to be completed, that being "Have extra information displayed on selection". This was similar to 'set a description for the tag', but both can be said to be completed with the ability to see and set the description of the tag, known as extra information.

Make Tags Page mobile friendly.

⌚ Mar 21

AL

Have a button to pickup a tag when near it

⌚ Mar 21 ➡ Done

AS

When clicked, put the tag into "Picked" status

⌚ Mar 21

AS

Have extra information displayed on selection

⌚ Mar 21 ➡ Done

AC

Set a description for the tag

⌚ Mar 21

AC

Have a specific tag icon for each category

⌚ Mar 21 ➡ Done

AC

Have a set number of tag categories to choose

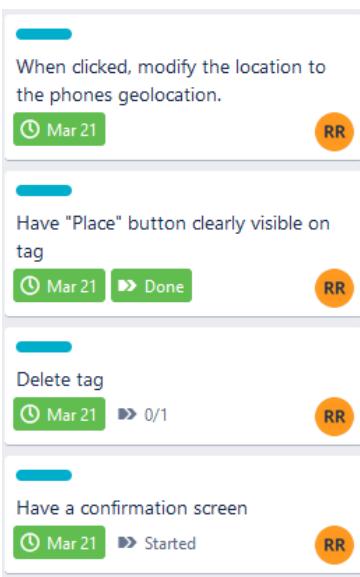
⌚ Mar 21 ➡ Done

AC

Enter a manual coordinate

⌚ Mar 21 ➡ 0/3

AC



Have tag shaded out when in "Picked" state

⌚️ 🕒 Mar 21 ➡ Started AR

Fix Ngrok Cross-domain Cookies

⌚️ 🕒 Mar 21 AR

Sprint 11 (21/03/2023-28/03/2023)

After review, we have found we are pretty much done for functionality in the main areas of the website. However, we found there is still neglect in account management. We found that being able to edit the user email and password, and also allow the deletion of their account, to be important.

We also want a sign-out option for the users, in case they want their token's invalidated. The motivation is that the data stored is quite important, and should they wish, they can delete all their account data from the database.

So, we assigned Alan and Rodion to parts of this process:

Create Account Page	AS
Update user email	AS
Update user password	AS
Delete Account	RR
Sign out	RR
Show confirmation screen when deleting account	RR

Rodion is focused on deletion and signing out, whilst Alan is on updating details.

We also have some user stories not completed in the tag theme, so Alfie was made to make the add tag button automatically retrieve coordinates, and also add proper local parsing to the user input fields for signing up, adding tags, etc.

Automated Coordinate retriever when adding a tag

⌚ Mar 28 ➡ Started

AR

Parse inputs through local and server checkers

⌚ Mar 28 ➡ Started

AR

Outside of our planned user stories, for design we wish for a top-bar of the page to be properly visible. This also needs to have a user account icon on the top right, like most typical websites which should redirect to the account page.

So, we got Alex to go through this process:

Make Login & Sign up Pages Mobile Friendly

⌚ Mar 28

AL

Make a top-bar that shows navigational pages

⌚ Mar 28

AL

Make top-bar show the user who's logged in and allow navigation to account's page

⌚ Mar 28

AL

For Aidan, he felt he could properly design the user instruction booklet, now that we have a clear usage for the website, so we assigned him to that. This will hopefully give us a clear reference on website usage in design, aswell.

Have a instruction booklet clearly accessible and usable

⌚ Mar 28

AC

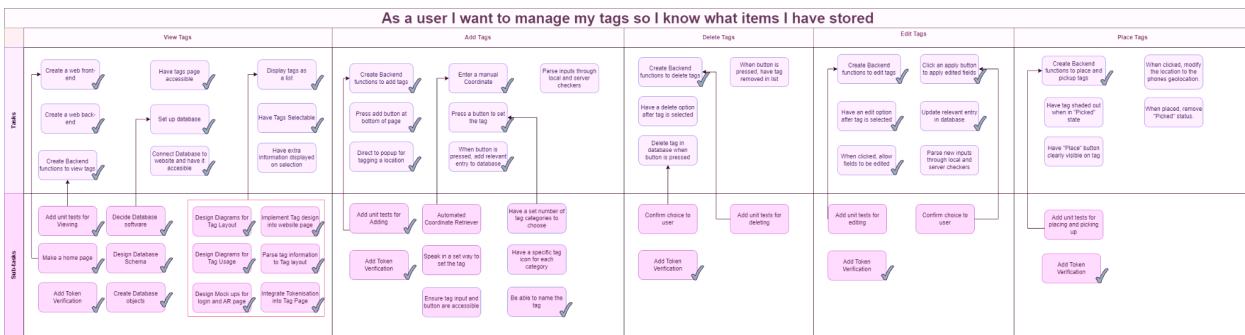
Fix Icon displaying on edit

⌚ Mar 28

AC

He also noted some bugs to fix, so we assigned that explicitly.

Showing these changes on the story maps, only the tag theme had changes, from:



To:



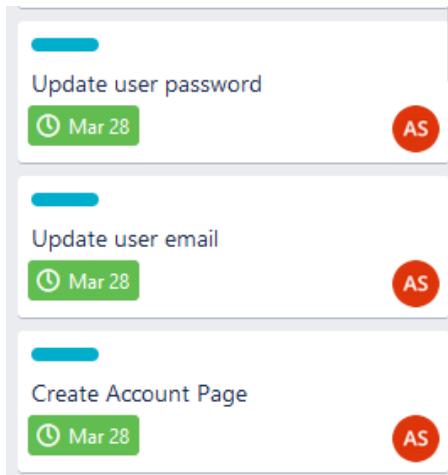
So, in total the assigned user stories for sprint 11 are:

- Add unit tests for User Backend
⌚ Mar 28 OD
- Create delete and sign out backend functions for users
⌚ Mar 28 OD
- Create a cohesive CSS design
⌚ Mar 28 VP
- Create Account Page
⌚ Mar 28 AS
- Update user email
⌚ Mar 28 AS
- Update user password
⌚ Mar 28 AS
- Delete Account
⌚ Mar 28 RR
- Sign out
⌚ Mar 28 RR
- Show confirmation screen when deleting account
⌚ Mar 28 RR
- Automated Coordinate retriever when adding a tag
⌚ Mar 28 ➡ Started AR
- Parse inputs through local and server checkers
⌚ Mar 28 ➡ Started AR

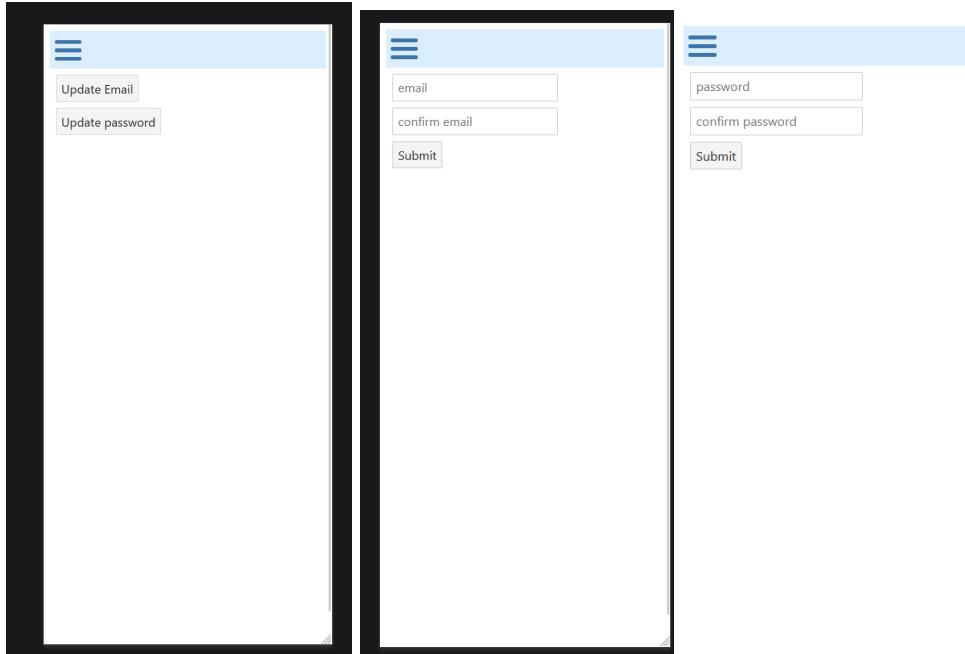
- Make Login & Sign up Pages Mobile Friendly
🕒 Mar 28 AL
- Make a top-bar that shows navigational pages
🕒 Mar 28 AL
- Make top-bar show the user who's logged in and allow navigation to account's page
🕒 Mar 28 AL
- Have a instruction booklet clearly accessible and usable
🕒 Mar 28 AC
- Fix Icon displaying on edit
🕒 Mar 28 AC

Sprint 12 (28/03/2023-04/04/2023)

For this sprint, our general goal was to finish the remaining features that are 'needed', as users would expect these features in websites.



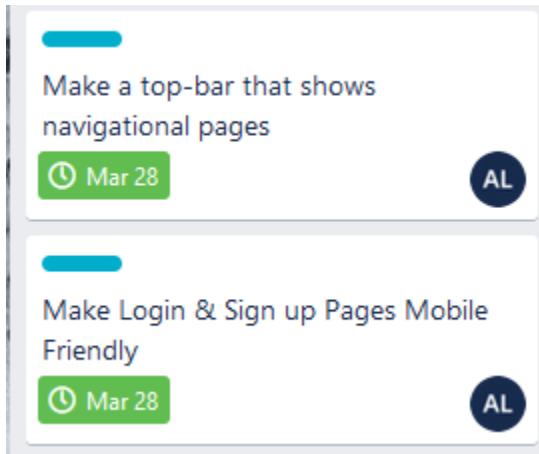
We created the account page, and added two buttons that direct to two pages to update the email and password:



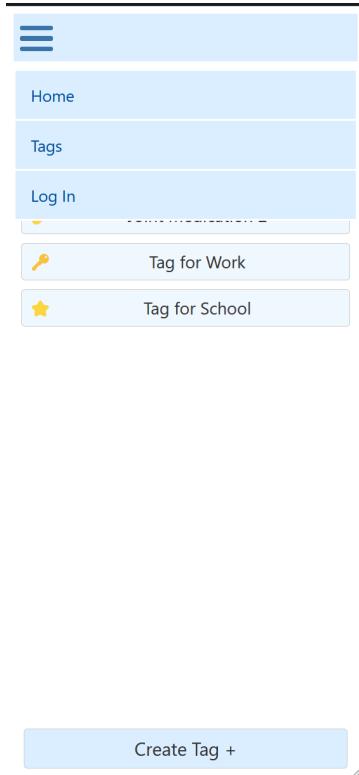
No styling has been done, as our design currently isn't set as done, so we won't do work without following a strict guideline.

With this, the users can change their email and password should they need to, and the motivation is that this is functionality all websites with user management have.

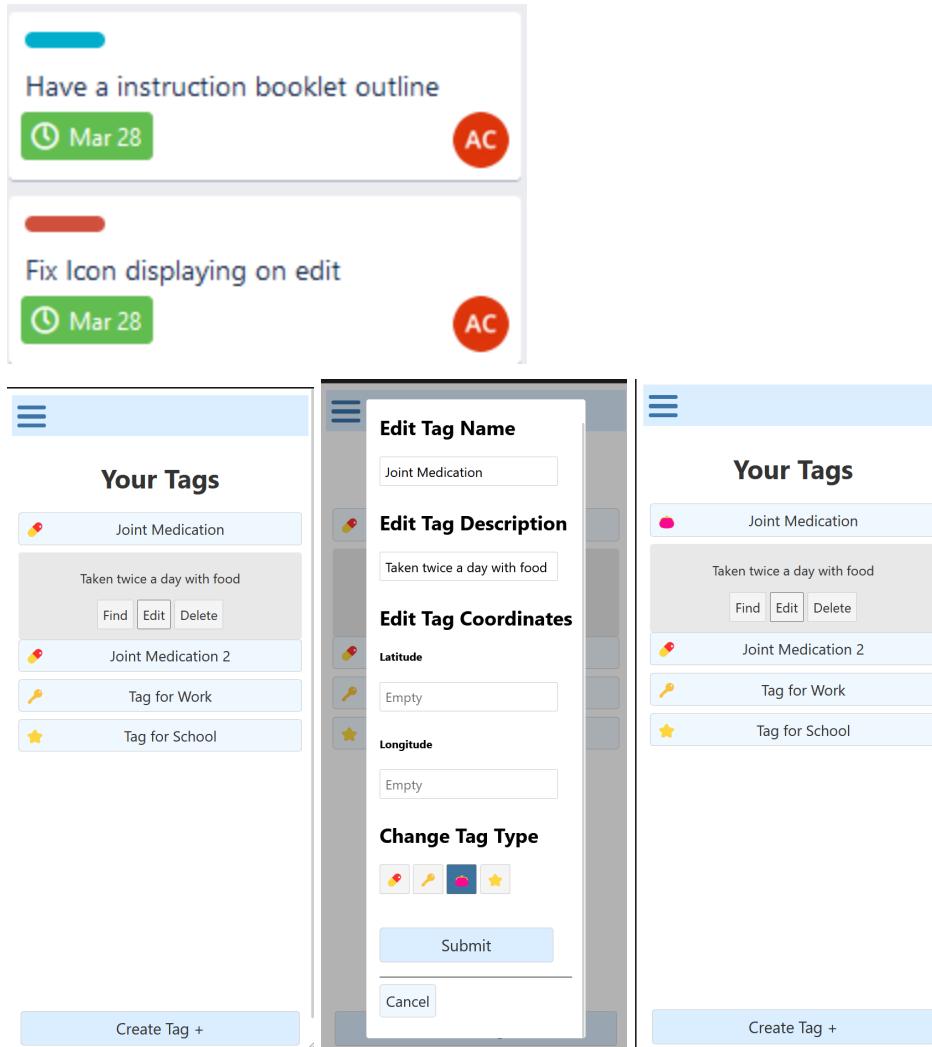
From this, Alex was tasked to make a proper top-bar for our website. This would allow easy navigation through our website, and will need to be large and easy to press for phones and those that are tech illiterate.



Alex completed two of his 3 user stories, missing the one to create the account icon in the top right. Showing this:

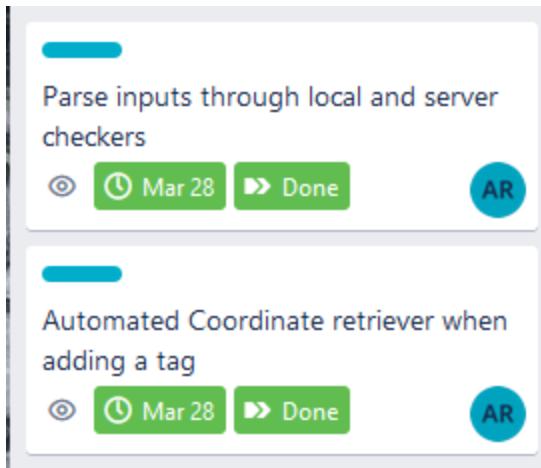


Moving on to Aidan, he was able to fix the icon display issue, and was also starting the outline of the instruction booklet. As the website main appearance is variable, he felt that he should wait until he could get a reliable method of navigation before creating the booklet, so we edited his user story to show he created an outline, instead.



The bug was just an issue with displaying the icon you see on the left part of the tag. Editing the tag now allows it to change easily.

From this, Alfie completed the task of parsing the inputs through the local functions when signing up, and also made it so that adding tags will automatically use geolocation to get their coordinates.



Showing this, with adding a tag:

Your Tags

Enter Tag Name
Keys

Enter Tag Description
Don't forget today

Select Tag Type

Joint Medication

Submit

Cancel

Create Tag +

localhost:4000/tags

Allow localhost:4000 to access your location?

Joint Medication

Joint Medication 2

Tag for Work

Tag for School

Adding tag, please wait...

Create Tag +

And, allowing the location, it adds the tag:

Your Tags

- Joint Medication
- Joint Medication 2
- Tag for Work
- Tag for School
- Keys

Don't forget today

Edit Tag Name

Keys

Edit Tag Description

Don't forget today

Edit Tag Coordinates

Latitude
52.9518824

Longitude
-1.1515526

Change Tag Type

Submit

Create Tag +

Cancel

For local validation, sign up can show the error checking:

Please Sign Up

Name

Email

Password

Sign Up

Your password must be at least 8 characters

Your password must contain at least one letter.

Your password must contain at least one digit.

Please Sign Up

Name

Email

••

Sign Up

Your password must be at least 8 characters

The notification system was also implemented to do this, and so Alfie implemented this elsewhere to show action reactions. This will make it easier to tell that something is happening, and this is important for people who lack confidence in using technology.

Alfie also worked on styling the pages so that they were centred, in the tags page, and login and sign up. This is some preliminary work on polishing the UI.

Rodion felt he needed a few more days to complete his user stories, which were:

	Create delete and sign out backend functions for users	
	Delete Account	
	Sign out	
	Show confirmation screen when deleting account	

His job is to have deletion and sign out capability, with both being important for data control and security for the user. So, we made it due an extra due alongside his other work due over easter in easter sprints.

From this, we felt as a team we have created most functionality that is important to manage tags, and manage user accounts. This, around the AR component, has created a good functional base to now polish and beautify.

For Rodion, we assigned him:

Make loading screen for placing tags
and long processes

⌚ Apr 4

RR

Comment Backend code

⌚ Apr 4

RR

These two tasks are just to comment code that doesn't have enough documentation, and add loading screens to reassure users when something is taking a couple seconds to process.

For alfie, we assigned:

Have a message asking to place the phone where they want the tag, when they place it.

⌚⌚ Apr 4

AR

Implement reminders to place tags back

⌚⌚ Apr 4 ➡️ Started

AR

When placing a tag, we felt it would be useful to add an explicit message to tell the user that they should put their phone where their tag is, otherwise we would be assuming they would know to do that, and that assumption can never be held.

And, one last piece of functionality requested from the customer, was to have reminders when you pick up tags for a long period of time. This will help customers remember their actions with light intervention.

For Aidan, as he does HCI we felt he would be a good choice to design the color scheme and general style for our website. This will be a formal design, using research, so we have assigned him:

Make colour schemes and general design

⌚ Apr 4

AC

This will be used to implement the CSS on the website.

For Alan, he felt he could improve the AR functionality:

Make AR system more polished and try relative positioning

⌚ Apr 4

AS

As the AR functionality relies solely on GPS, which is inaccurate, he felt that trying some sort of relative positioning may be possible, but as this is an optional area, we are just asking for some preliminary ideas on this focus, as relative position is much more accurate once we have a point of reference to get our location from.

This would entail needing the user to go to this fixed location whenever they want to find a tag, however.

Alex, with his previous user story, was tasked to implement the CSS derived from Aidans work he will complete:

Make top-bar show the user who's logged in and allow navigation to account's page

⌚ Apr 4

AL

Implement design in CSS on other pages

⌚ Apr 4

AL

This is expected to be done after Aidan is completed, but assigning both at the same time is fine as Aidan said he expects to finish it quite quickly.

As Vishal had issues getting on his laptop, we decided to limit the scope of his CSS implementation to two pages:

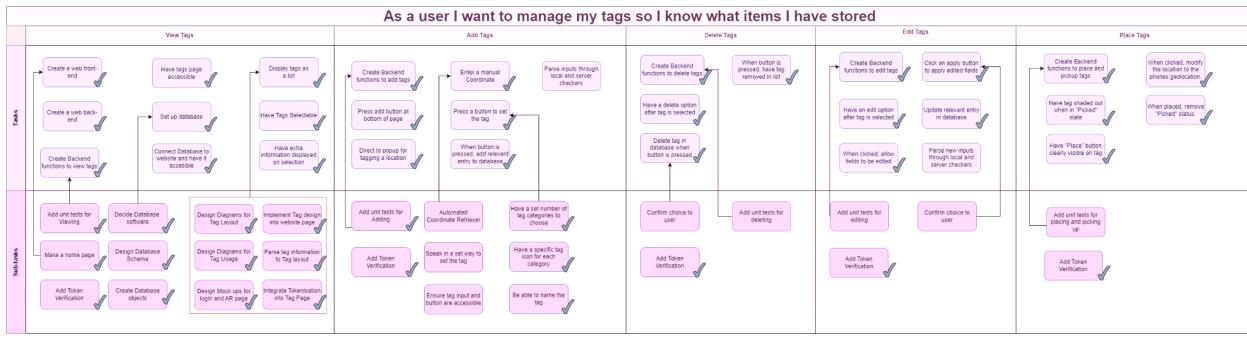
Implement CSS design on login and sign up pages

⌚ Apr 4

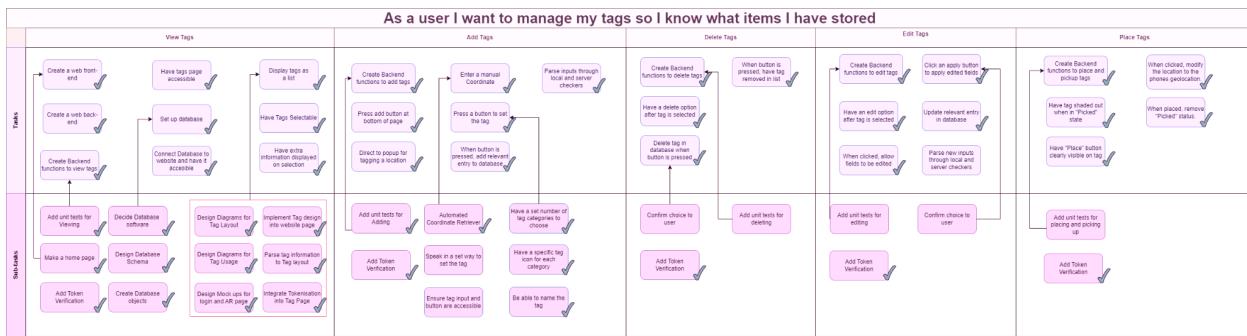
VP

Alex is tasked with implementing the other pages.

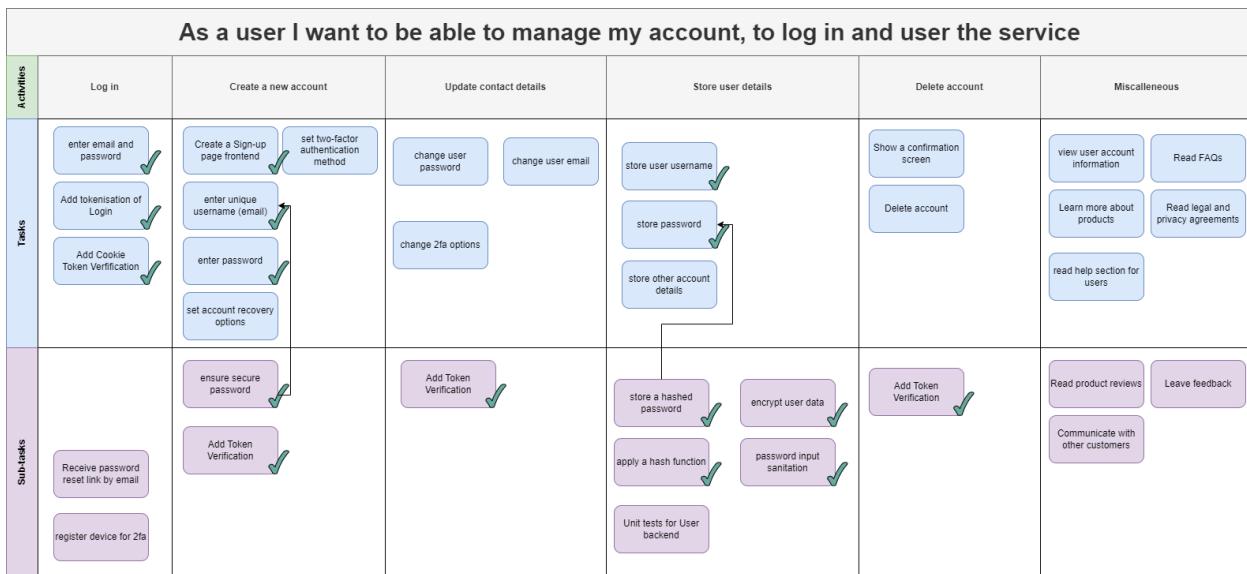
So, in summary, showing the story maps that have changed, for tag management:



To:



And for accounts, from:



To:

As a user I want to be able to manage my account, to log in and user the service						
Activities	Log in	Create a new account	Update contact details	Store user details	Delete account	Miscellaneous
Tasks	enter email and password ✓ Add tokenisation of Login ✓ Add Cookie Token Verification ✓	Create a Sign-up page frontend ✓ set two-factor authentication method enter unique username (email) ✓ enter password ✓ set account recovery options	change user password ✓ change user email ✓ change 2fa options	store user username ✓ store password store other account details	Show a confirmation screen Delete account	view user account information Read FAQs Learn more about products Read legal and privacy agreements read help section for users
Sub-tasks	Receive password reset link by email register device for 2fa	ensure secure password ✓ Add Token Verification ✓	Add Token Verification ✓	store a hashed password ✓ encrypt user data ✓ apply a hash function ✓ password input sanitization ✓ Unit tests for User backend	Add Token Verification ✓	Read product reviews Leave feedback Communicate with other customers

We expect most of that to be done once Rodion has finished his part. The rest of the user stories are optional functionality.

Sprint 13 (04/03/2023-11/04/2023) (Sprint 12 Retrospective)

This sprint was done during easter, and so will be a short version.

Alfie was able to complete both push notifications and the place message:

Implement reminders to place tags back

⌚ Apr 4 ➤ Done

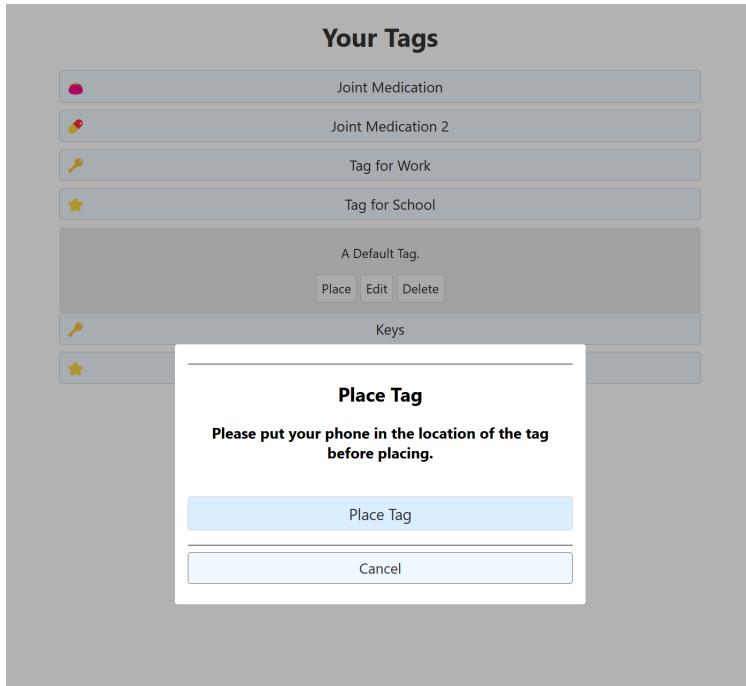
AR

Have a message asking to place the phone where they want the tag, when they place it.

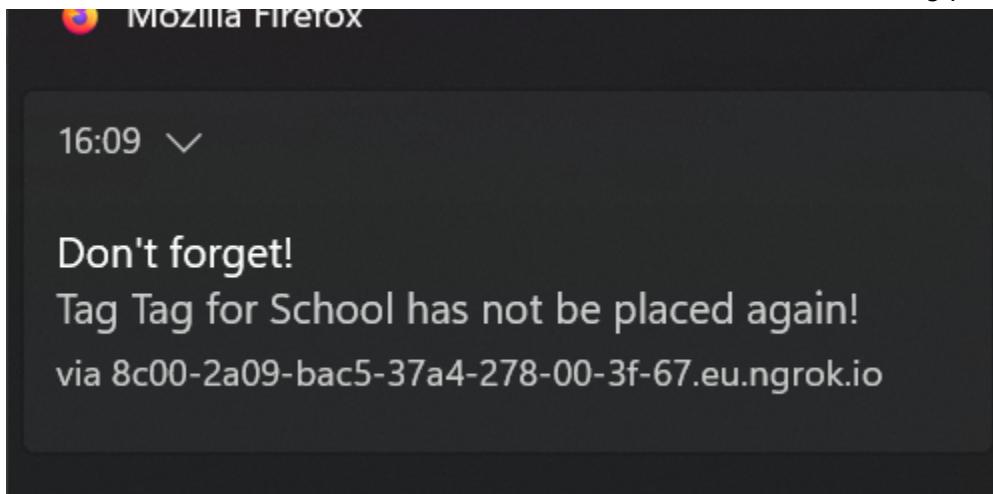
⌚ Apr 4

AR

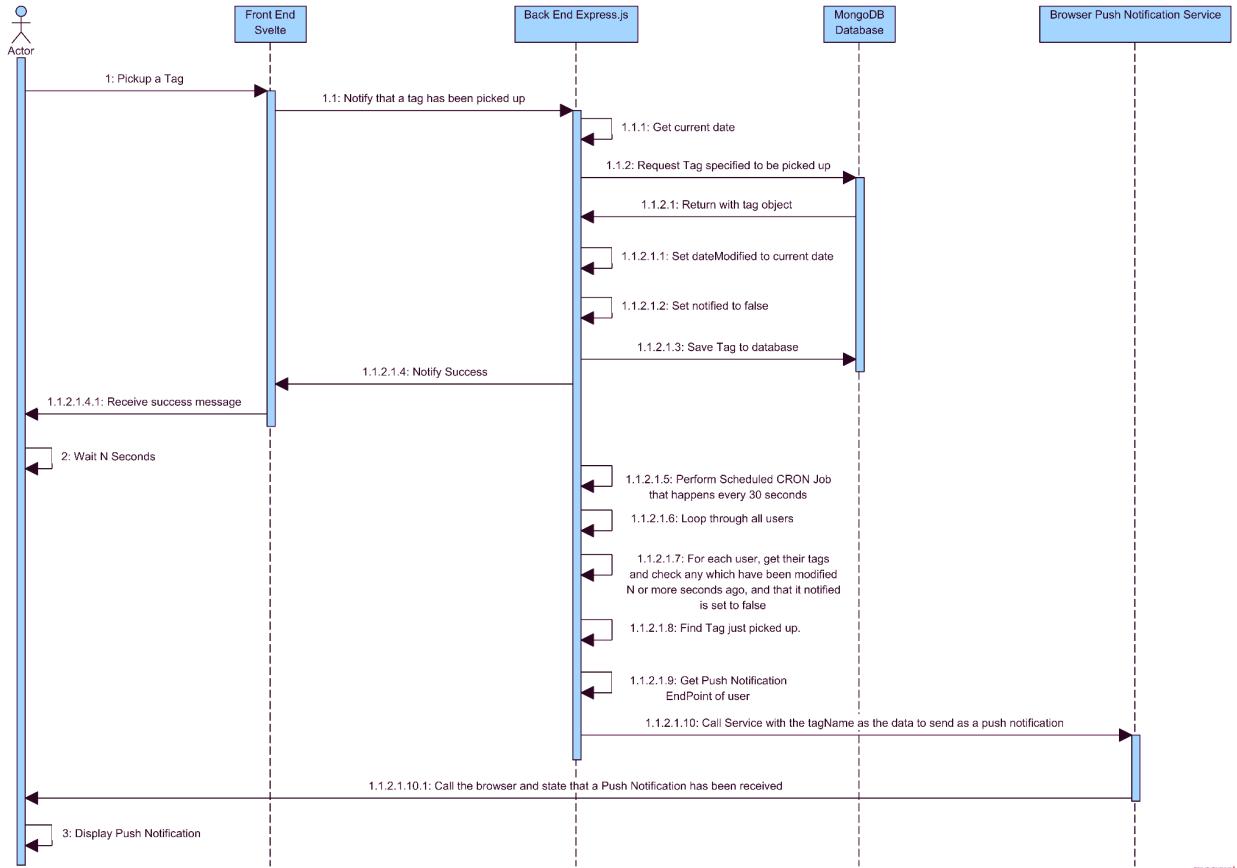
Showing this for placing a tag:



And, push notifications are automatically pushed to the user whenever they pick up a tag, and then leave it for more than n seconds. We set it to 30 seconds for testing purposes, for example:



Is displayed because the Tag for School tag has not been placed 30 seconds after being picked up. This required a number of technical additions, such as adding dateModified fields in the tags so that we can compare the date with the current time in a cron job.



This is the basic idea, assuming the endpoint has been set in the user database already (it is set per user).

Aidan completed a colour scheme outlining the best colours to be used for those that are elderly or have colour blindness.

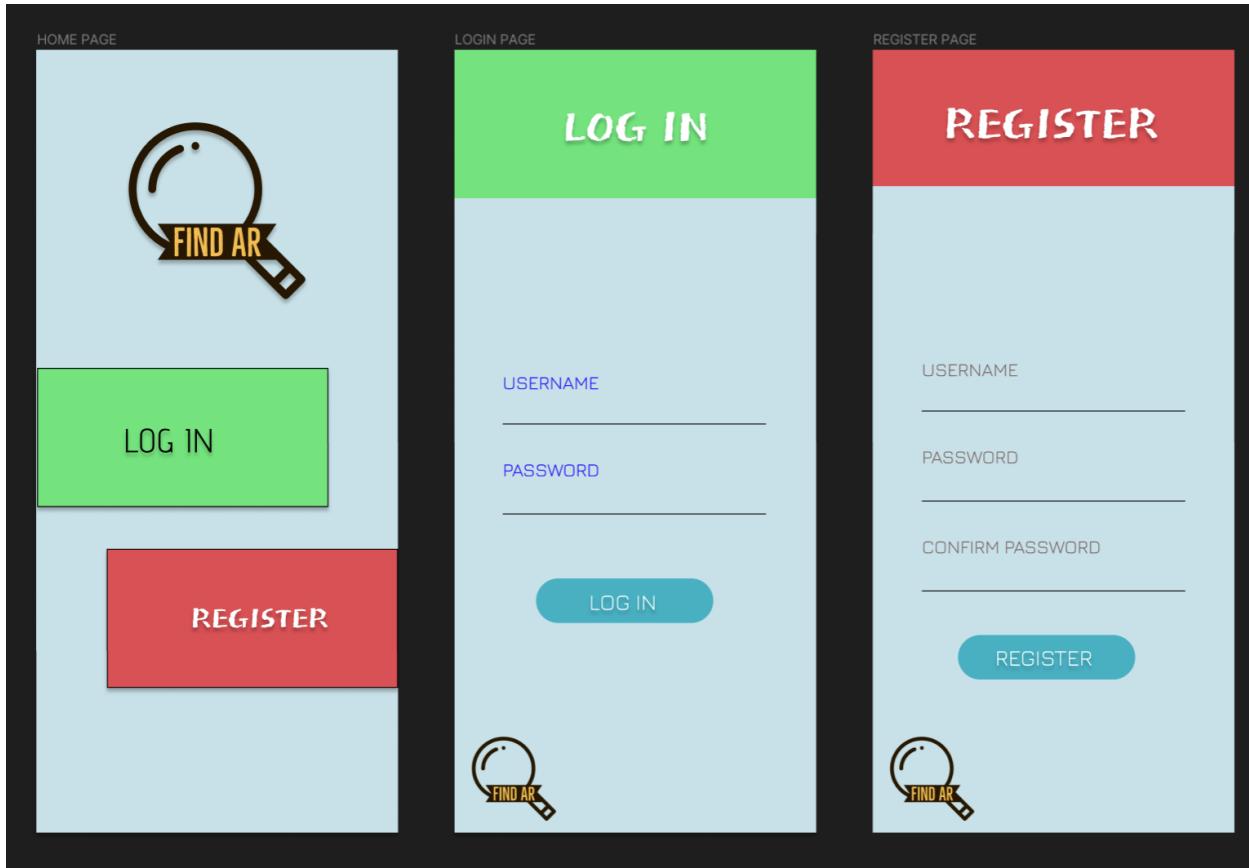
[Colour Scheme](#)

After this, Rodion was able to complete all his user stories except commenting:

Make loading screen for placing tags and long processes	⌚ Apr 4	RR
Show confirmation screen when deleting account	⌚ Mar 29	RR
Sign out	⌚ Mar 29	RR
Delete Account	⌚ Mar 29	RR
Create delete and sign out backend functions for users	⌚ Mar 29	RR

Those due the 29th were from the previous sprint. Rodion said he would like to make a javaDoc style API sheet for the backend, to make the comments more formal. This will help with the software specification, so we agreed.

Vishal created some mock sheet for the design of the webpage, mostly following Aidans colour scheme:



We took these into account and will now iterate them to better follow Aidan's colour scheme.

For assigning user stories, Alfie was assigned:

Display correct routes depending on whether you're logged in or not

⌚ Apr 11

AR

Iterate Colour Scheme for Website

⌚ Apr 11

AR

Ensure tag input and button are accessible

⌚ Apr 11 ➡ Started

AR

Currently, the website doesn't specialise the routes it shows to you, so you can go to the tags page, etc, when you're not logged in. So, we need that fixed for usability. Next, he will iterate over Vishal's design and implement it into css.

This coincides with making the tag input and buttons accessible, as the colour scheme is made to suit the visually impaired, and be bright and clean.

Moving on, Rodion was assigned to do the jsDoc/API of the backend:

Make an API with JsDoc

⌚ Apr 11 ⏹

RR

This simply will help with the software specification.

Alan agreed to also comment up the frontend code and make it more presentable, and fix code smell:

Comment Frontend Code and make it readable

⌚ Apr 11

AS

This will help with further development and prevent large unwieldy code that nobody wants to touch.

Aidan agreed to improve our previous interim report for preparation of the group report:

Add motivation to Project background and understanding section of iterim report

⌚ Apr 11

AC

As most of the functionality is done, we will focus much more on documentation next sprint.

Vishal agreed to work on unit testing the backend's tag delete function, as he felt it would be useful to learn more on how the backend functions, because he had trouble early on the development getting docker working, so didn't have the opportunity.

Add unit tests for deleting

⌚ Apr 11 ➡ Started

VP

This will broaden the teams general knowledge with Vishal gaining some knowhow in using the backend.

For story maps, a lot of the user stories are outside the scope of the general development, but some have been changed, so for account management:

As a user I want to be able to manage my account, to log in and user the service						
Activities	Log in	Create a new account	Update contact details	Store user details	Delete account	Miscellaneous
Tasks	enter email and password Add tokenisation of Login Add Cookie Token Verification	Create a Sign-up page frontend set two-factor authentication method enter unique username (email) enter password set account recovery options	change user password change user email change 2fa options	store user username store password store other account details	Show a confirmation screen Delete account	view user account information Read FAQs Learn more about products Read legal and privacy agreements read help section for users
Sub-tasks	Receive password reset link by email register device for 2fa	ensure secure password Add Token Verification	Add Token Verification	store a hashed password encrypt user data apply a hash function password input sanitization Unit tests for User backend	Add Token Verification	Read product reviews Leave feedback Communicate with other customers

To:

As a user I want to be able to manage my account, to log in and user the service						
Activities	Log in	Create a new account	Update contact details	Store user details	Delete account	Miscellaneous
Tasks	enter email and password Add tokenisation of Login Add Cookie Token Verification	Create a Sign-up page frontend set two-factor authentication method enter unique username (email) enter password set account recovery options	change user password change user email change 2fa options	store user username store password store other account details	Show a confirmation screen Delete account	view user account information Read FAQs Learn more about products Read legal and privacy agreements read help section for users Sign out
Sub-tasks	Receive password reset link by email register device for 2fa	ensure secure password Add Token Verification	Add Token Verification	store a hashed password encrypt user data apply a hash function password input sanitization Unit tests for User backend	Add Token Verification	Read product reviews Leave feedback Communicate with other customers

These being Rodion's user stories for deletion and signing out.

And, for the AR theme:

As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website ✓ Form an icon to float above the exact location ✓ Be able to get geolocation data in longitude and latitude 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located 	<ul style="list-style-type: none"> Have the user press a button to state they have reached their location. Have a button to pickup a tag when near it When clicked, put the tag into "Picked" status Implement reminders to place tags back
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR ✓ Research GeoLocation API ✓ Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

To:

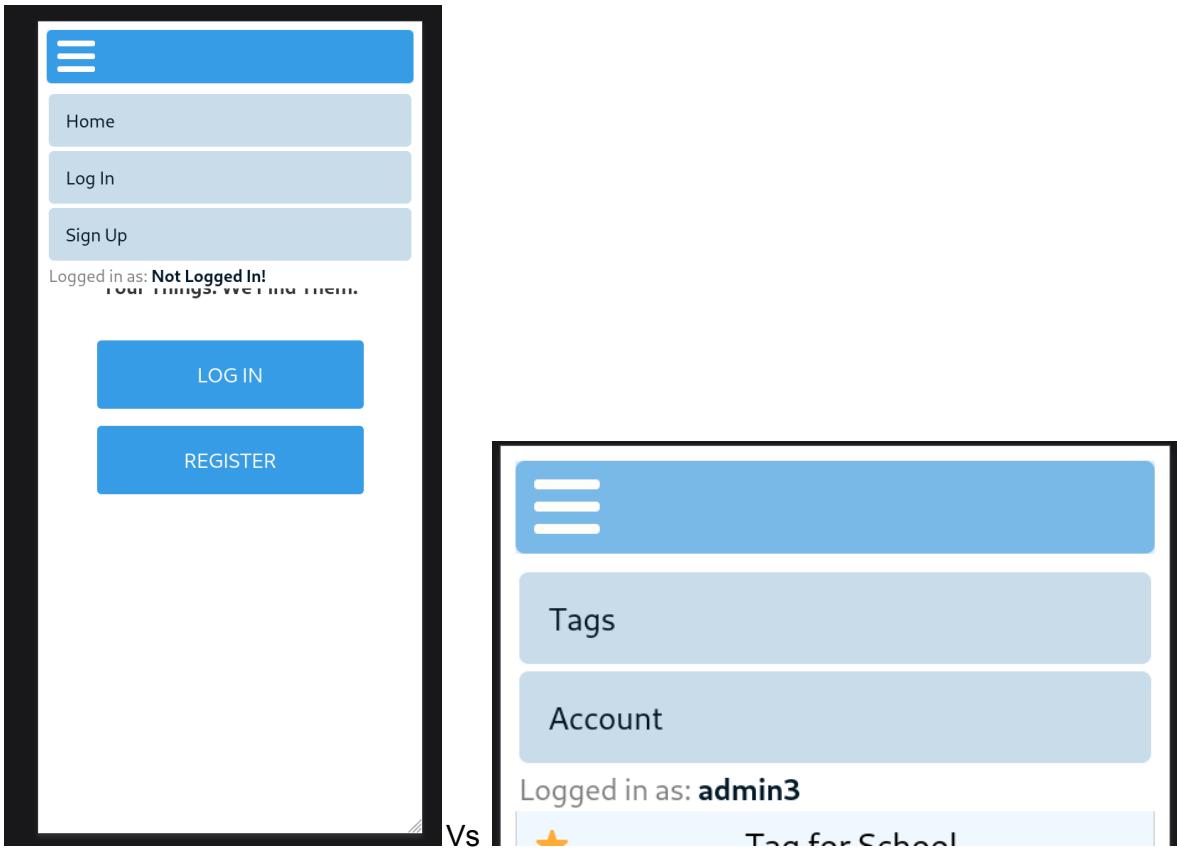
As a user I want a visual AR indication of where my tag is so I know where my items are.			
Activities	Have a AR icon on that location	Have the icon space sensitive	Have a Pickup button
Tasks	<ul style="list-style-type: none"> Research methods to integrate AR into the camera ✓ Design using relevant diagrams an overview of the AR system ✓ Have the page access the camera ✓ Integrate AR.js into the website ✓ Form an icon to float above the exact location ✓ Be able to get geolocation data in longitude and latitude 	<ul style="list-style-type: none"> Have the icon grow in size the closer you are Have some indicator for what rotation the icons are located 	<ul style="list-style-type: none"> Have the user press a button to state they have reached their location. Have a button to pickup a tag when near it ✓ When clicked, put the tag into "Picked" status ✓ Implement reminders to place tags back ✓
Sub-tasks	<ul style="list-style-type: none"> Have a page for viewing the tags in AR ✓ Research GeoLocation API ✓ Make the page automatically open the camera ✓ 	<ul style="list-style-type: none"> Perform a visual indicator of distance Congratulate via visual effects when approaching from a set distance 	

This being the reminders when picking up a tag.

Sprint 14 (12/03/2023-19/04/2023) (Sprint 13 Retrospective)

This focused on finishing the styling, and adding documentation, which this won't explain.

Alex finished the account page, and also finished the protected routes. So, when you're logged out, it displays:



Alfie finished the styling of the website, which you have seen a bit. For the home page:

Your Things. We Find Them.

LOG IN

REGISTER

Please Log In

Name

Password

LOG IN

----- OR -----

SIGN UP

Please Sign Up

Name

Email

Password

SIGN UP

- ### Your Tags
- Tag for Work
 - Tag for School
 - Keys**
 - Default
 - Default5
 - checking
 - Default
 - Default
 - Default
- Create Tag +**

- ### Your Tags
- Tag for Work
 - Tag for School
 - Keys**
- Don't forget today
- Place**
- Edit**
- Delete**
- Default
 - Default5
 - checking
 - Default
 - Default
 - Default

Enter Tag Name
Default

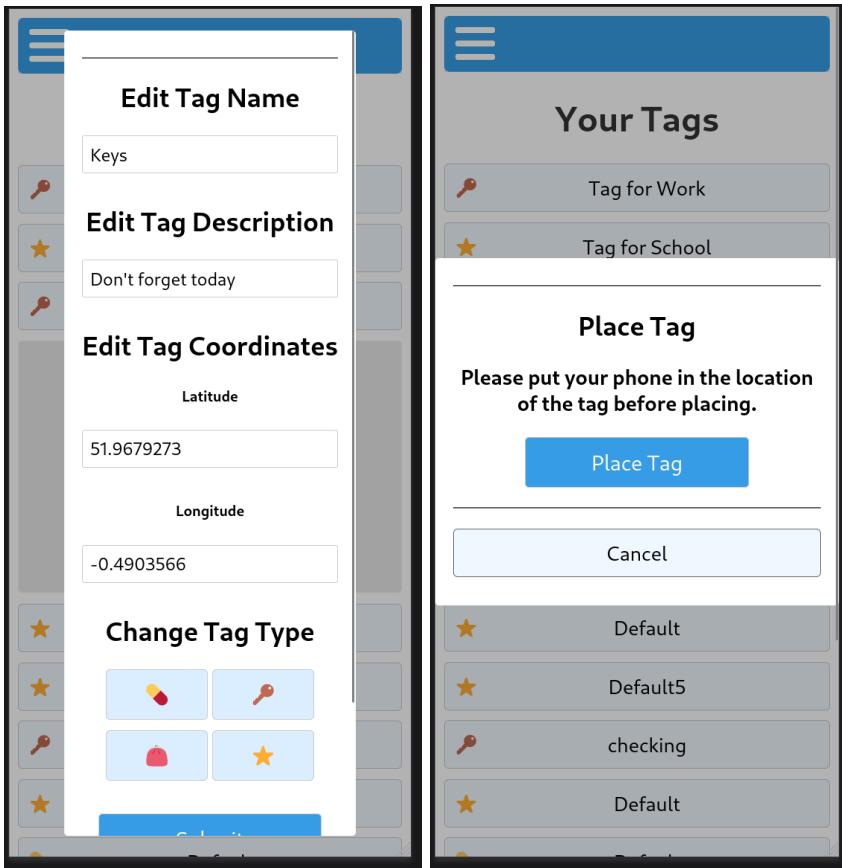
Enter Tag Description
Empty

Select Tag Type

Submit

Cancel

Create Tag +



We also had a meeting with the supervisor, who said that we should add a home page with details on the functionality of the website, and should make the tags page more user friendly by making it clear you can open a tag.