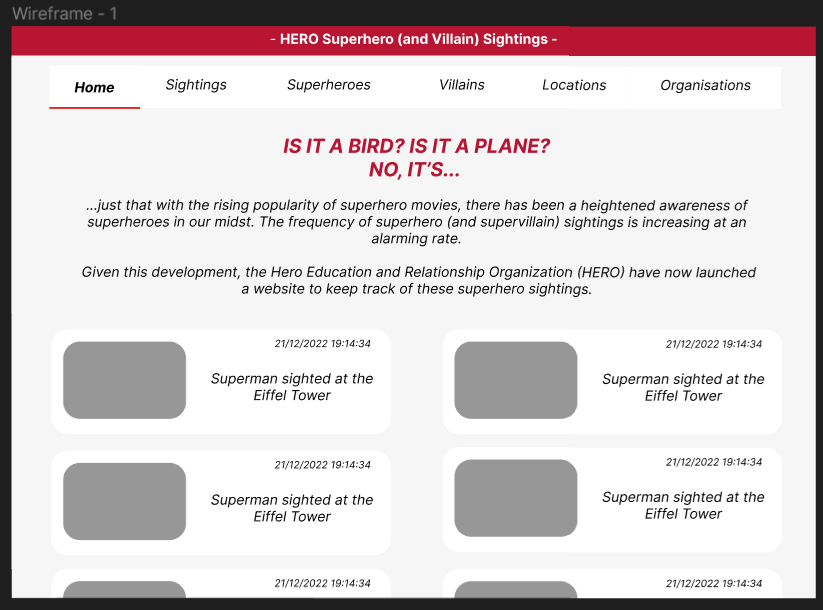
Wireframes

# Landing Page



Default landing page – no path variable beyond localhost:8080

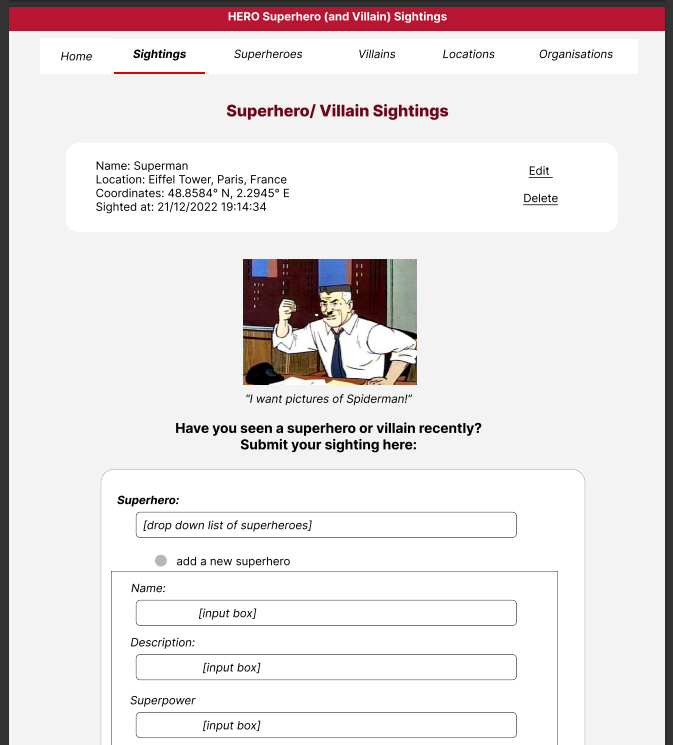
Query: get 10 most recent sightings

* Grouping: Need to order in reverse chronological order (most recent first)

Elements used: if the sightings entity consist of a superhero and location entity, and a datetime object, it will need superhero.name, location.name, and the date and time

Grey box = either map embedding or a picture (optional)

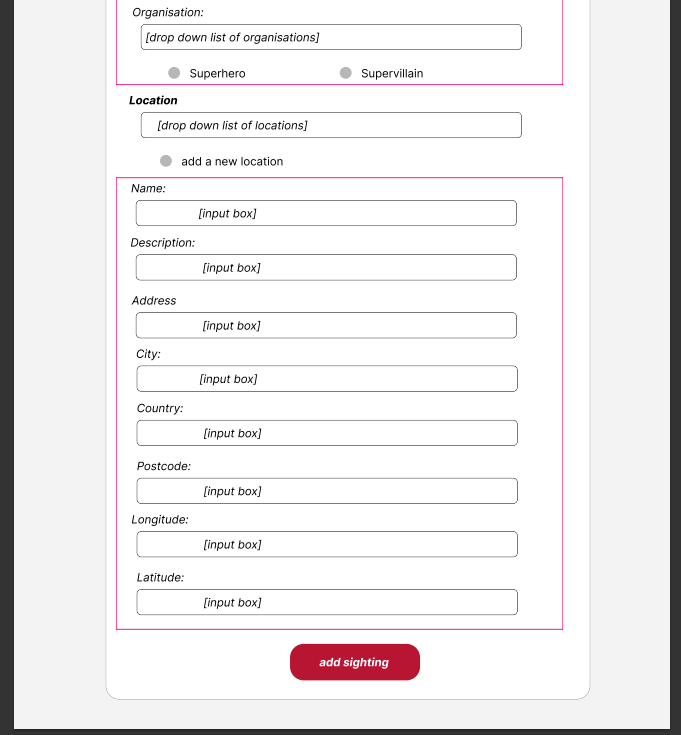
# Sightings



@getMapping(/sightings)

Top half displays all the sightings using a get all sightings query

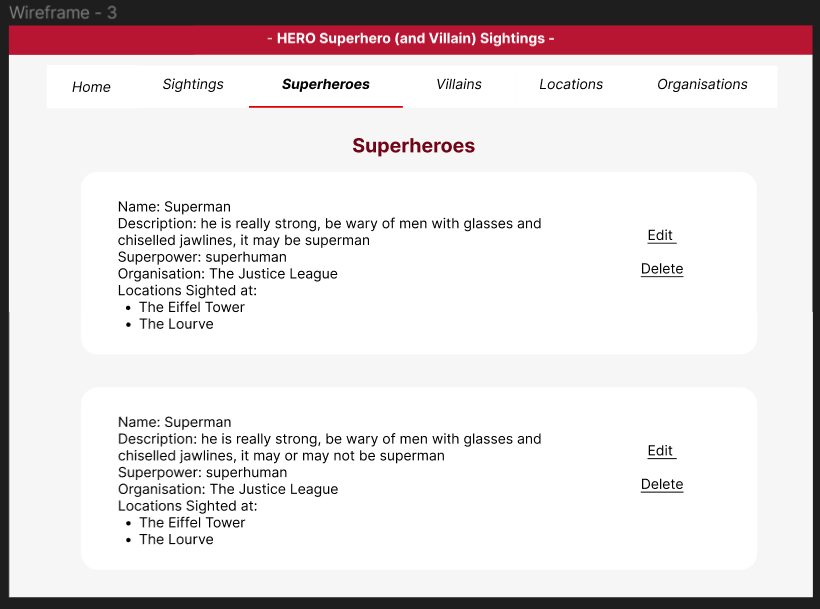
* Needs superhero name
* Location name, address, city, country, postcode, coordinates
* Date time
* Option to edit = calls @getMapping(/editSighting(id))



Form

* Box in pink, in a div that’s hidden until the user selects to add a new location/ superhero?
* Didn’t make sense to add a superhero/ location separately from a sighting, so it’s done here
  + User has option to either select all the superheroes and locations, or add a new one if they select the radio button

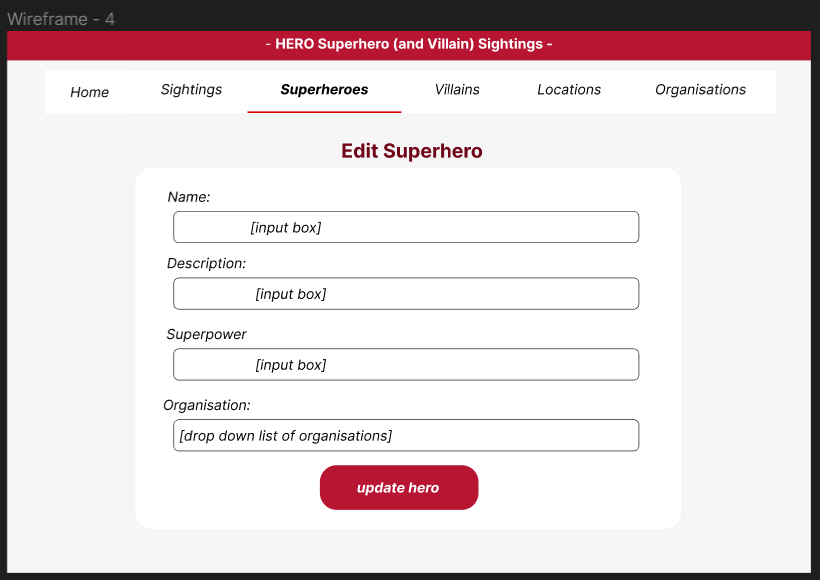
# Superheroes



@getmapping(/superheroes)

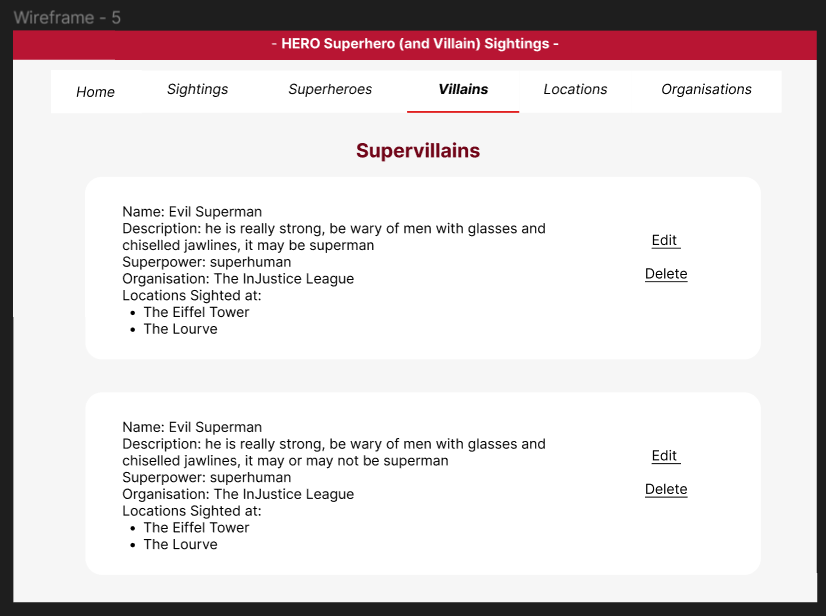
* Uses a query to display all superherovillains where ishero = true to only show the superheroes
* Also uses a query to find all the locations where the specific superhero has been sighted at
* Selecting delete calls postmapping(@deleteSuperhero)
  + Needs to delete any sightings where the superhero was involved with as well

# Edit Superhero



* Getmapping(/editSuperhero)
* Should have existing values inside the input boxes and drop down box, that the user can edit from this page to update the database

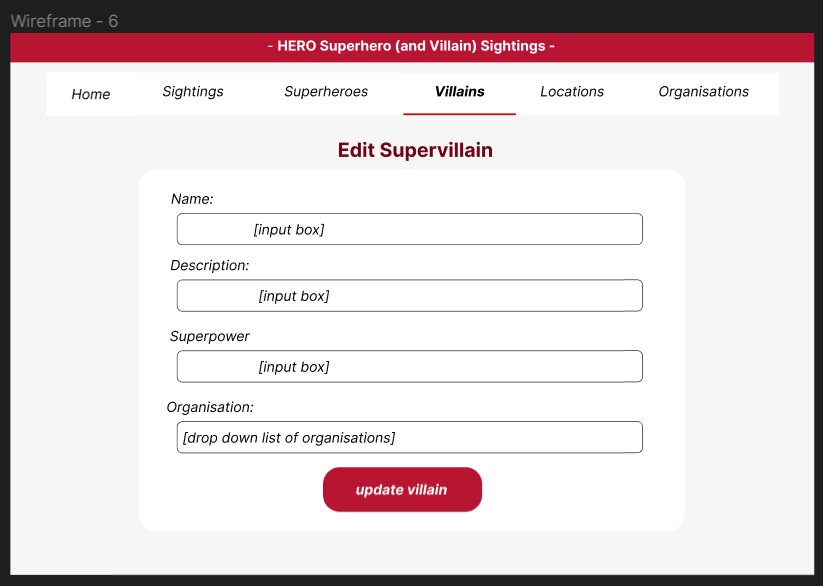
# Supervillains



@getmapping(/supervillains)

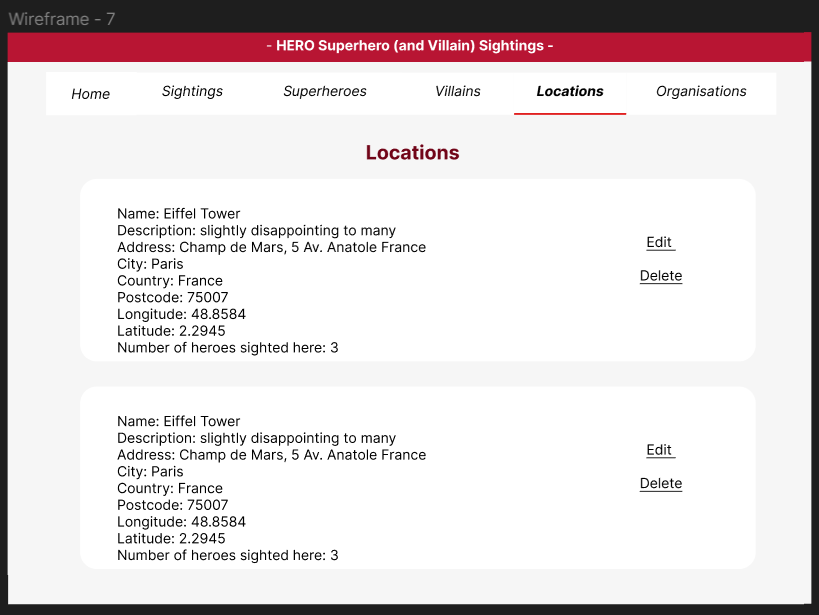
* Uses a query to display all superherovillains where ishero = false to only show the supervillains
* Also uses a query to find all the locations where the specific supervillain has been sighted at
* Selecting delete calls postmapping(@deleteSupervillain)
  + Needs to delete any sightings where the supervillain was involved with as well

# Edit Supervillain



* Getmapping(/editSupervillain)
* Should have existing values inside the input boxes and drop down box, that the user can edit from this page to update the database

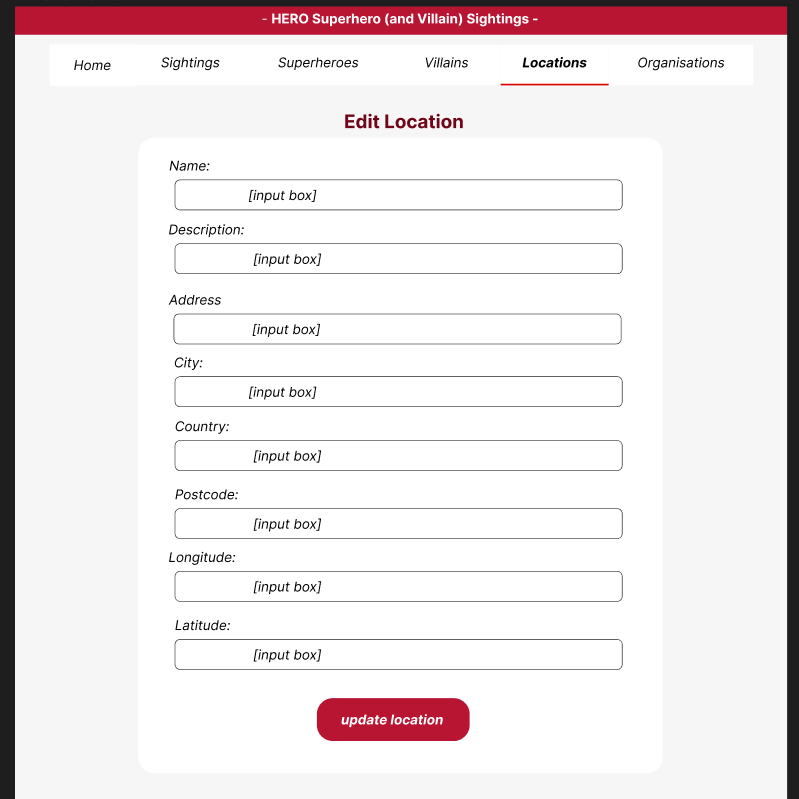
# Locations



@getmapping(locations)

* Uses a query to get all of the locations stored within the database
* Uses a query to count the number of heroes that have been sighted at that location
* Selecting delete calls postmapping(@deleteLocation)
  + Needs to delete any sightings where the location was involved with as well

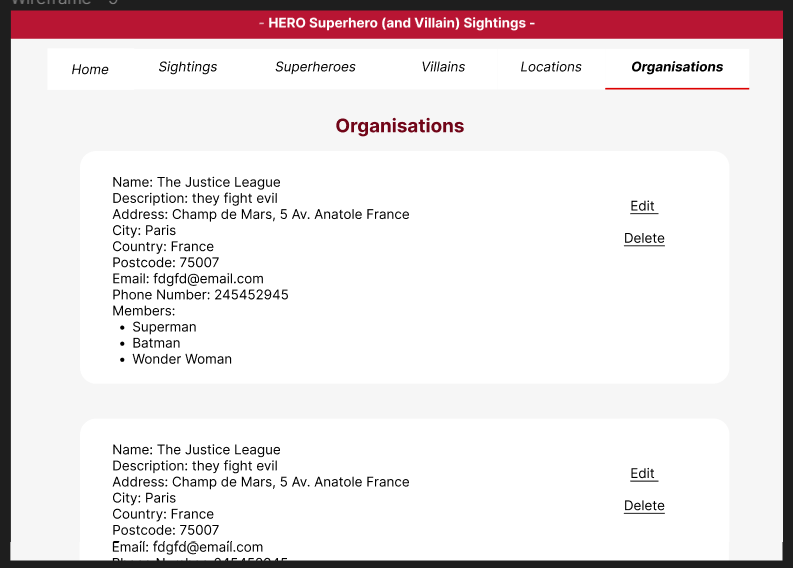
# Edit Locations



* Getmapping(/editLocation)

Should have existing values inside the input boxes and drop down box, that the user can edit from this page to update the database

# Organisations



Getmapping(/organisations)

* Uses a query to display all of the organisations
* Also has a query to display all the superheroes associated with the organisation