## Semantic mark-up and component deconstruction

From the screenshot below, provide an annotation of:

* 1. the semantic mark-up and page hierarchy (h1, h2, h3 etc.) of the page
  2. the high-level components that the page is comprised of

A screenshot of a computer

Description automatically generated

## Data analysis

You have received several, high-fidelity mock-ups (Step A, B and C), and are tasked with the job of identifying what parts of it might come from a data-source that will be provided to you, and what parts will be hardcoded into the application and its components.

Identify the logical entities across each of the pages and describe their structure and properties based on how they have been used in these pages. Provide your thought process in comments.

An example output, based on what could be inferred from Step A (below):

{

"search": {

"location": {

// The state is individually selected in the box on the left

// so this must be separately stored

"stateOrTerritory": "NSW",

"name": "Darlinghurst"

},

// These search terms don't seem to fit in any of the other headings

"keywords": [

"skin",

"biopsy"

],

// Assume that no specialties chosen means "all" because

// that’s more sensible than listing them all individually

"specialties": []

}

}

Step AA screenshot of a cell phone

Description automatically generated

Step B

A screenshot of a computer screen

Description automatically generated

Step C

A screenshot of a computer

Description automatically generated

## Code review, tests and comments

We have a fictitious, existing code repository, with a not-very-good application. It renders a list of Foo objects and provides the user with a way of marking a Foo as complete, as well as creating new Foos. Foo objects have some attributes which are randomly assigned on creation. Each completed Foo which is sendable will be sent by some asynchronous process.

Your task here is:

* Write some tests you think would improve the stability of the app
* Fix any incidental issues you might discover while testing/using the app
* Write good commit messages detailing your approach