# **Capstone Introduction:**

**Title Page should include:**

What is the title of your project? What is the problem you are trying to solve? What needs are

you trying to address? How is your proposed approach/product/technology/study?

* The app Is created for the Users to search about covid- 19 testing centers near them.  
  Walk-in appointments or drive through appointments can be booked through the app.  
  Depending on the age and availability of the dates they can go for testing.  
  The app should have introduction to covid symptoms and related Information.
* Enough research should be done before publishing any results related to the article and data from multiple resources.
* A map should be shown at the bottom of the page.  
  Daily published news about the topic should be updated on the app.

**Goal :**

What is the goal of your capstone project?

The goal of my capstone project is to launch an app to locate the centers for the covid -19 testing and book an appointment for any kind of testing such as walkins or drive through and choose a time for the appointments according to the age and availability .

**Learning Goals:**

What would you like to learn during the execution of your capstone project? Do you plan to use

your capstone to create a business? What skills are you planning to develop? What new

approaches or technologies are you planning to learn?

• Learn how to design an online course curriculum

• Learn how to select mixed media to effective most effectively

Develop an app which is useful for the general population to find the information about the covid testing centers.

**Target Audience, Personas & Empathy Maps:**

My target audience is anyone who can access the internet and get the info needed any years old who have access to smartphone, internet and have knowledge of the english language. I will use technology which will be accessible to everyone with a smartphone and a Printer.  
  
**Capstone UI/UX Checklist II: Design**

**As a developer I won’t be**

* Leaving the fonts and styles completely untouched from default (text, forms, links).
* Placing UI elements or content up against other elements or the edge of the window.
* Overusing neon or overly bright, glaring colors for blocks of text or large backgrounds.
* Using font sizes or color combinations that make text hard to read (e.g. pale font on a light background - always consider [contrast readability](http://colorsafe.co/)).
* Overusing fonts from the handwriting or display categories from Google Fonts.
* Overusing jQuery animation or CSS transition effects that can become sluggish (note that not all users are on high-end devices that can execute these effects smoothly).
* Overusing images, specifically ones that haven’t been optimized for the web (compress jpeg and png images [online](http://tinypng.com/)).
* Using a CSS framework (e.g. Bootstrap, Materialize) if previously used on another capstone (CSS frameworks limited to use on only one capstone).

## **Part 2: In this capstone, I am using…**

* Using less than three fonts and a pairing that doesn’t clash (check out recommended pairings by [ReliablePSD](https://www.reliablepsd.com/ultimate-google-font-pairings/) and [Femmebot](http://femmebot.github.io/google-type/) or go to directly to Google Fonts, click on “see specimen” under a specific font and scroll down to “Popular Pairings” to preview Google’s recommendations).
* Using moderate spacing – margin, padding, and line-height – to avoid elements from being positioned right up against another (read about the [importance of white space](http://uxmyths.com/post/2059998441/myth-28-white-space-is-wasted-space)).
* Using Fontawesome icons (sparingly and with text labels in most cases) for menu items or section titles.
* Being consistent with styling by writing CSS rules for either elements or classes.
* Using a complementary color combination for content and UI elements (explore good [color combinations](http://color.adobe.com/)).

# **Capstone UI/UX Checklist: Responsiveness**

## **In this project, I have used..**

* Mobile-first design - all main features and content are present on mobile view.
* Used the [viewport meta tag](https://developers.google.com/speed/docs/insights/ConfigureViewport#overview) in the <head> section: <meta name="viewport" content="width=device-width, initial-scale=1">
* Added [media queries for all standard devices](https://responsivedesign.is/develop/browser-feature-support/media-queries-for-common-device-breakpoints/):
  + Smart phone portrait
  + Smart phone landscape
  + Tablet portrait
  + Tablet landscape
* Written CSS rules for each media query (screen size) that optimize:
  + Text readability (e.g. font sizes easy to read without having to zoom in/out).
  + Image to screen ratio (e.g. image width/height defined by percentage of screen size).
  + Usability of UI (e.g. size and vertical spacing for clickable elements).
* Used a [responsive CSS grid](https://github.com/Thinkful-Ed/responsive-grid-example-and-challenge/blob/solution/css/float-grid.css) to vertically reorganize any horizontal columns or boxes when user views on a mobile device.
* Avoided positioning any content or elements in a way that requires user to scroll horizontally or zoom in/out to view (excluding map data).
* Converted desktop horizontal nav menus to a drop down vertical menu on mobile view.
* Used Chrome DevTools to simulate [common mobile device viewports](https://developers.google.com/web/tools/chrome-devtools/device-mode/emulate-mobile-viewports#viewport-controls) and confirmed that everything behaves as expected.
* Tested the project on as many devices as possible to account for [the limitations of DevTools simulations](https://developers.google.com/web/tools/chrome-devtools/device-mode/emulate-mobile-viewports#limitations).