Front-End Development Capstone Project

Table of Contents

[Project Scenario 2](#_Toc190622192)

[Resources to Support Successful Project Completion 4](#_Toc190622193)

[Capstone Project Technical Reference 5](#_Toc190622194)

[Tasks Overview: Designing Website Layouts 7](#_Toc190622195)

[Hands-on Lab: Create Mockup Designs for Website Layouts 9](#_Toc190622196)

# Project Scenario

Estimated Time: 10 minutes

**Scenario Overview**

**StayHealthy Inc.** is a non-profit organization dedicated to improving healthcare facilities in remote areas with minimal to no medical facilities or resources. The Executive Director has started the Go digital initiative as part of StayHealthy's broader vision to maximize reach. One aspect of the initiative is building a website to help patients access doctors anytime, anywhere.

You are a **front-end developer** assigned to design and develop the website.

**Problem Statement**

StayHealthy conducted an extensive survey to assess the medical facilities available in remote areas and found that several patients have a tough time finding the right doctor at the right time. There are usually no doctors or hospitals within a 50-mile radius, or the wait time to meet a doctor is too long. Many times, this has resulted in life-threatening emergencies or lost lives.  
With the internet and broadband availability increasing in remote areas, the initiative from StayHealthy will help get patients access to general physicians and specialists at the right time easily.

**Project Requirements**

* A user-friendly, intuitive, and accessible platform that helps connect patients with doctors and specialists for consultations and medical prescriptions

**Features**

* **Medical appointments online**
  + The website must allow patients to schedule appointments with doctors anywhere, anytime.
  + Consultation must be available doctors at any time should be possible without prior appointments.
  + The patient/end-user must be able to view the different time slots when a doctor is available and select the time slot of their preference.
  + The patient must get a reminder or notification on an upcoming appointment.
  + The patient must have the ability to cancel or modify the appointment.
* **Doctor listing for patients**
  + The patient/end-user must be able to view the available doctors with their ratings.
  + The patient/end-user must be able to view the doctor's ratings and review comments.
  + The patient/end-user must be able to search for a specific doctor by name.
* **Consultation Feedback**
  + The patient/end-user must be able to rate the consultation on effectiveness in terms of diagnosis, communication, and so on to contribute to the system's transparency and help other patients make informed choices.
* **Profile Management**
  + A user must sign-up to make any appointment bookings.
  + To book an appointment, a user must input mandatory details like name, address, phone number, age, blood group, and any existing medical records.
  + The user must have the ability to update their details.
* **Access to patient records**
  + Only the doctor and patient must have access to patient records.
  + Patients must be able to download their medical records, such as test reports.
* **News feeds and Training**
  + To keep patients informed of different medical health conditions and healthy lifestyles:
    - Educational videos on various topics should be available and sorted by category.
    - There are daily tips on healthy lifestyle and other preventive measures accessible.
* **Navigation**
  + Users must be able to navigate the site with minimal clicks.
* **Website Administration**
  + Users must be able to manage patient and doctor profiles, maintaining the system's security and integrity.
  + Users must be able to generate reports and analytics on appointment bookings, patient demographics, and healthcare provider performance.

The development of the project will be phases. In the first phase of the project, the Executive Director has requested for the following features:

* Medical appointments online
* Doctor listing for patients
* Consultation Feedback
* Profile Management
* Navigation

**Expectations from you**

For the first phase of the project, you will need to:

* Design the UI/UX for the website using Figma
* Develop static and responsive pages using HTML, CSS, and Reactjs
* Deploy the website

It is also expected that you follow the application development lifecycle as depicted below. So as you begin with this scenario, make sure to analyze the requirements mentioned here thoroughly before you move to the next phase.



**Note:** You can enhance the front-end user interface for different Web pages and components throughout the development process.

# Resources to Support Successful Project Completion

This capstone project is a way to test your understanding and proficiency across various concepts and tools you learned in different courses in the [**IBM Front-End Developer Professional Certificate**](https://www.coursera.org/professional-certificates/ibm-frontend-developer) program. To ensure your success, here are some resources that are built into the course to help you complete the project.

**Exemplar Code**

In every lab, you will find exemplar code to guide you when building the website. The code block has descriptions for different snippets that explain the logic of building the function.

**Design Guidelines**

Designing website layouts is the first step. While the generic design guidelines and procedures steps for creating layouts in Figma are covered in the UI/UX course, you will find additional references to design elements available in the Figma community to help you learn more about UI/UX, interactions, and more.

**Git Commands for the Project**

Git commands will be used to manage code for the project's various parts, which are covered in different labs. We do not expect you to memorize all the commands. You can use the [**Git Commands**](https://cf-courses-data.static.labs.skills.network/Er0P-NR6MdEQkLA3J-xqUA/github-commands-v1.md.html) reading as a reference in case you are stuck.

**Technical Reference**

Various concepts and tools were covered as part of the Front-End Professional Certificate. Most of the tools will be used in this project. You may also learn about some additional concepts and tools. To help you explore additional concepts and tools, you can use the links in the [**Technical Reference**](https://cf-courses-data.static.labs.skills.network/lx899cZNWtiXjgrx9gvXXg/reference-frontendcapstone-v1.md.html) reading.

**Additional Learning**

* Introduction to different approaches of creating websites using React including Vite and Create React App (CRA)
* Optimizing code using Webpack explained with an example
* Automated CI/CD using GitHub actions

# Capstone Project Technical Reference

**Introduction**

Throughout the project, you might encounter new concepts that have been added to ensure you are able to develop a fully functional project. This project can be added to your portfolio and showcased to potential employers. This reading is a collation of React concepts and tools that are referenced in the capstone project.

**React Router**

React Router is a client and server-side library for React. React Router enables client-side routing.

You can use React Router DOM, an npm package, to implement dynamic routing in a web app. React Router DOM enables the functionality to navigate across various components and pages in a single-page web application (app) without refreshing the page, instead dynamically fetching content based on the URL. This process is called routing.

You must install **react router dom** in the React project's root folder using npm i react-router-dom to create routes.

You can find more information on React Router and related concepts and uses at [**reactrouter.com**](https://reactrouter.com/en/main/start/tutorial).

**Which hands-on lab implements this feature?**

[**Convert Static Pages to Dynamic React Components**](https://cf-courses-data.static.labs.skills.network/IBMSkillsNetwork-CD0322EN-Coursera/labs/m3/l2/Transfer_Static_Components_to_React.md.html)

**ES7+ React/Redux/React-Native snippets**

**ES7+ React/Redux/React-Native snippets** are extensions for React, React-Native, and Redux in JavaScript with TypeScript (JS/TS) with ES7+ syntax. ES7+ is customizable and integrates with Prettier, a code formatting tool.

You can find more information about the extensions and ES7 installation instructions in the [**Visual Studio Marketplace**](https://marketplace.visualstudio.com/items?itemName=dsznajder.es7-react-js-snippets).

**Which hands-on lab implements this extension?**

[**Convert Static Pages to Dynamic React Components**](https://cf-courses-data.static.labs.skills.network/IBMSkillsNetwork-CD0322EN-Coursera/labs/m3/l2/Transfer_Static_Components_to_React.md.html)

**Event handlers**

Event handlers are functions that you can create to customize how a component responds to a user interaction, such as a click or hover. When developing the capstone project, you will use at least two event handlers in each component.

**onFocus**

The onFocus event occurs when an element receives focus. For example, when a user clicks inside a text box, it triggers the onFocus event that allows the user to see what they type. The onFocus event is commonly used to execute functions or perform actions when an element gains focus.

Refer to the **[onFocus](https://react.dev/reference/react-dom/components/common" \l "focusevent-handler" \o "onFocus" \t "_blank)** topic on the *react.dev* website for more information.

**onBlur**

React onBlur is an event that triggers when an element loses focus. You can use the onBlur event for components, such as input fields, text areas, and buttons.

Refer to the **[onBlur](https://react.dev/reference/react-dom/components/common" \l "focusevent-handler" \o "onBlur" \t "_blank)** topic on the *react.dev* website for more information.

**Which hands-on lab(s) implement event handlers?**

[**Build the Appointment Booking Component**](https://cf-courses-data.static.labs.skills.network/32Z2Nu6JkEDxwq8pka9-BA/Build%20the%20Appointment%20Booking%20Component-v1.md.html)

You can also optionally consider using one or both the event handlers in the following labs including:

* [**Build the Reviews Component**](https://cf-courses-data.static.labs.skills.network/0xF1ysHTQdlS38FvAnVVYA/Build%20the%20Reviews%20Component-v1.md.html)
* [**Build the Profile Component**](https://cf-courses-data.static.labs.skills.network/ERFpAKbwH155quvLMPpaKQ/Build%20the%20Profile%20Component-v1.md.html)

**reactjs-popup library**

reactjs-popup is a versatile library that enables you to create and manage modal dialogs, menus, and tooltips within React applications. Using the library, you can perform various actions, such as displaying information and prompting specific actions without making the user navigate away from the current page. You can design visually appealing, animated, responsive, and customizable models to provide an enriching experience to users.

You can install reactjs-popup library using npm i reactjs-popup.

Refer to the [reactjs-popup](https://www.npmjs.com/package/reactjs-popup" \o "reactjs-popup" \t "_blank) section on the npm website for more information.

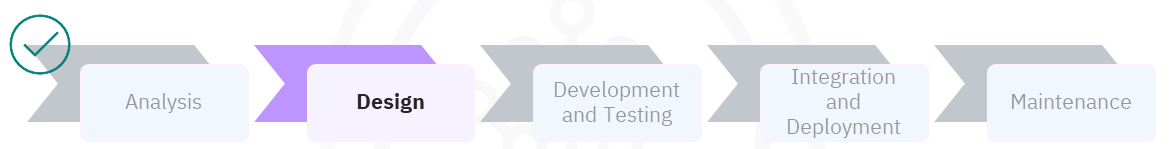
**Which hands-on lab implements the reactjs-popup library?**

[**Integrate Existing Functionality**](https://cf-courses-data.static.labs.skills.network/Qq9QYQ_gTjULqBq0-HRA3g/Integrate%20Existing%20Functionality-v1.md.html)

# Tasks Overview: Designing Website Layouts

**Estimated time needed:** 5 minutes

Congratulations on your new role as a front-end developer at StayHealthy Inc. After analyzing the business requirements documented in the [**Project Scenario**](https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMSkillsNetwork-CD0322EN-Coursera/labs/m1/l1/CapstoneProjectScenario.md.html), you can start designing the mockup layouts for the **Medical Appointment Booking** website in Figma.



In this lesson, you will perform the following tasks:

* Design mockup layouts for the following:
  + Navigation Bar using the Figma vector tools
  + Sign-up form using the Figma text and input tools
  + Login form using the Figma text and input tools
  + Application Booking features using the Figma design tool. The pages include:
    - Search for a doctor
    - Doctor Card with doctor details
    - Appointment booking
    - Appointment cancellation
    - Reviews section using the Figma design tools
* Test the layouts

Let's look at additional details for each element.

**Additional Information**

**Navigation Bar**

* The Navigation Bar must have links for the different functions, such as Appointments, Reviews, Sign-up, and Login. The StayHealthy logo should also be visible.
* The different items on the Navigation Bar must be designed considering the position, size, and style.
* The design must be responsive.

**Sign-Up Form**

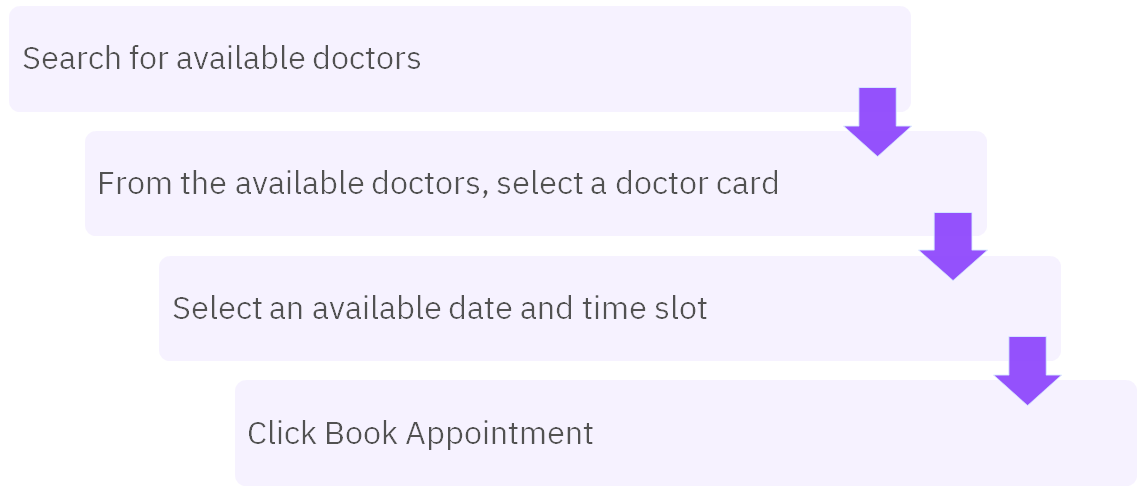
* As the name suggests, the Sign-up form must capture various fields, including user type (patient/doctor/admin), email address, name, phone number, and password.

**Login Form**

* The Login form must capture the email address and password.

**Appointment Booking**

The steps for Appointment booking are as follows:



Similarly, once a patient books an appointment, they must have the ability to cancel the booking.

Consider the above flow when creating the layout for Appointment Booking and associated elements.

**Reviews**

As the name suggests, the **Reviews** page will be about consultation feedback. The layout must reflect the following:

* List all consultations for the user with an option to provide feedback for each.
* The feedback form, must be designed to contain at least a 5-star rating mechanism for the consultation and free flow text to write a detailed review.
* If the feedback has been provided, the detailed feedback must be displayed.
* A user can only provide feedback once for a consultation.

**Key points to remember for all layouts**

* You must have a Figma account. If you have a free version of Figma, you are limited to managing a maximum of three projects simultaneously. Ensure you have two or fewer projects. If not, delete one project to make room for this project.
* Apply the UI/UX design principles as you design the mockup layouts. For example, the website:
  + Must be intuitive.
  + Must have a visually engaging interface that enhances user experience and encourages seamless navigation.
  + Must have properly labeled all elements.
* Explore various design elements to craft a platform that also serves its purpose effectively and leaves a lasting positive impression.
* Make sure to **take screenshots** wherever specified and save them!

Follow the instructional lab to complete the above tasks step-by-step.

Enjoy the creative process of bringing the website components to life! Happy designing!

# Hands-on Lab: Create Mockup Designs for Website Layouts

**Estimated time needed:** 90 minutes

**Introduction**

In this lab, you will visualize and design the layouts for a website based on a given scenario.

**Objectives**

After completing this lab, you will be able to:

* Visualize design concepts and ideas based on the given scenario
* Establish a clear information hierarchy within the interface
* Create a design for the website layouts while keeping the designs accessible for the **Medical Appointment Booking Application**, including:
  + Navigation Bar
  + Sign-up
  + Log-in
  + Appointment Booking
  + Reviews

**Prerequisites**

* You should have completed the prerequisite courses, especially the **Designing User Interfaces and Experiences (UI/UX)** course.
* You must have a Figma account. If you have a free version of Figma, you are limited to managing a maximum of three projects simultaneously. Ensure you have two or fewer projects. If not, delete one project from Figma to make room for this project.

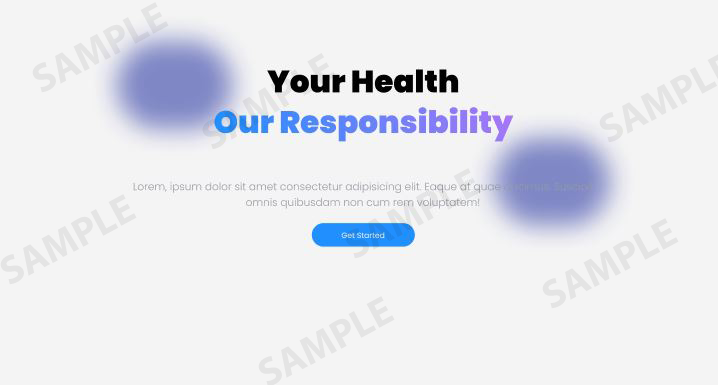
If you prefer revisiting the basics of Figma, click [**here**](https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMSkillsNetwork-WD0201EN-Coursera/labs/Module3/L1/Getting-started-with-Figma.md.html).

* Identify where you will source the images for the website.

**Project Scenario**

Click [**here**](https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMSkillsNetwork-CD0322EN-Coursera/labs/m1/l1/CapstoneProjectScenario.md.html) to review the project scenario.

StayHealthy Inc. has provided a sample landing page.

Click here to view the **sample**.

**References to help complete the lab**

Here are some Figma links that will help you design and complete the exercises:

* [Colors and Typography](https://www.figma.com/community/file/1305839990168004583/simple-design-system-typography-and-color-system)
* [UI Starter Kit](https://www.figma.com/community/file/1219594362058612280)
* [Style Guidelines](https://www.figma.com/community/file/902569748938803801/style-guidelines)

**Tips for finding and downloading images**

Images form an important part of any website. The same applies to this Capstone project. You can either find your own images for the project or use these suggested images from *[Pixabay](https://pixabay.com/" \o "Pixabay" \t "_blank)* and *[Unsplash](https://unsplash.com/" \o "Unsplash" \t "_blank)*, sites that supplies royalty-free images.

* [License information for Pixabay images](https://pixabay.com/service/license-summary/)
* [License information for Unsplash images](https://unsplash.com/license)

Download relevant images to a local folder.

**Exercise 1: Design the Navigation Bar**

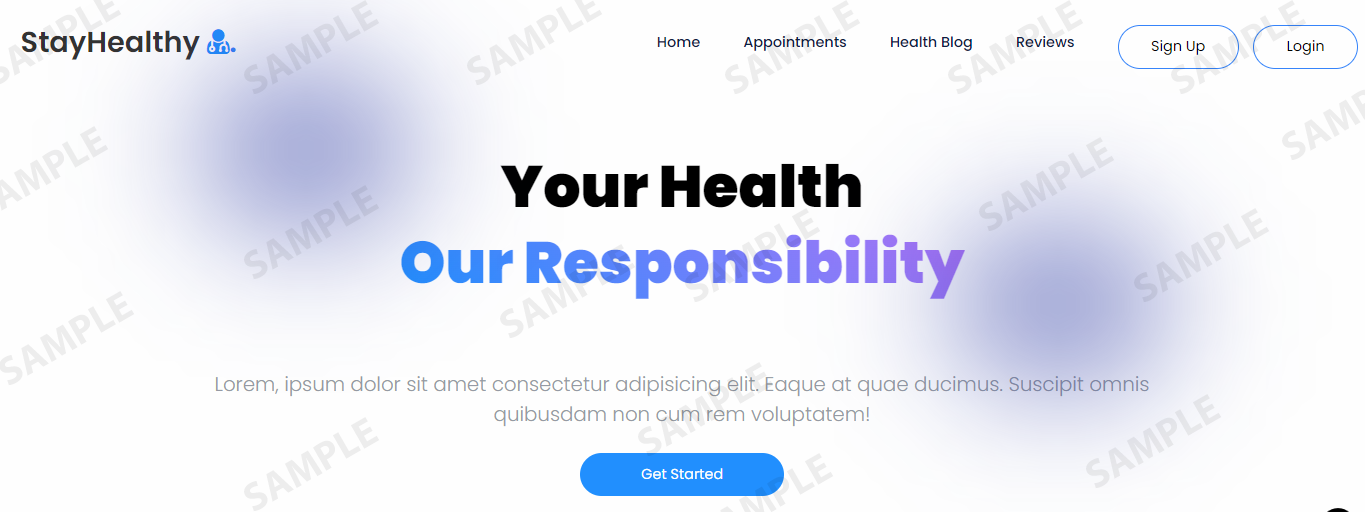
1. Create a frame for the navigation bar to be visible on every route.
2. The navigation bar must have links for appointments, reviews, sign-up, and login.
3. Design the layout of the Navigation Bar, considering its position, size, and style.

**Tips:**

* + You can design different types of navigation bars, such as the top horizontal navigation menu, sidebar static navigation, main menu, or breadcrumbs.
  + The recommended height for a navigation bar is 40-80 pixels to accommodate the readability of text.
  + To ensure sufficient space between components in the navigation bar, keep a gap of at least 10-20 pixels between components.
  + You can use a list and a table to display a collection of items, such as the different buttons on the Navigation Bar.
  + Define the margins.

1. Use the vector tools in Figma to create navigation icons or buttons.
2. Explore different color schemes and typography to make the navigation bar visually appealing and easy to use and then implement it in the design.
3. Ensure the navigation elements are clearly labeled and intuitive for users.
4. **Take a screenshot** of the navigation bar layout created in Figma and save it as **navbar\_design.png**.

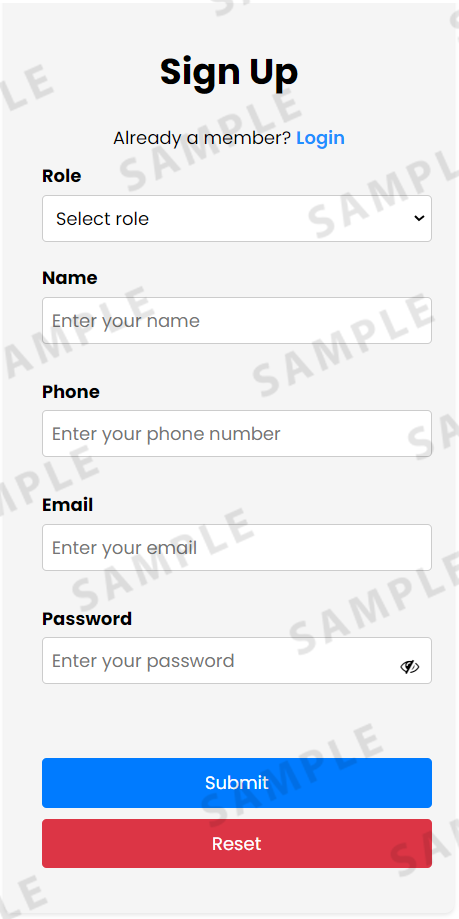
Click here to view a sample Navigation Bar layout.



**Exercise 2: Design the Sign Up form**

1. Create a frame for the **Sign Up** form.
2. Define the main layout of the frame (for example, a rectangle containing all the sign-up components).
3. Design the **Sign Up** form layout, including role, name, email, and password fields.
   1. Add the field names as labels. Keep the titles short and simple.
   2. Design a drop-down menu for the role field. You can provide two to three roles: Doctor, Patient, and Admin.
   3. For name and email fields, use the Text option.
4. Use the text and input tools in Figma to create the form elements.
5. Ensure the spacing, alignment, and visual cues to guide users through the sign-up process. In addition, it provides a high contrast between the text color and the background color of the navigation bar. When designing the navigation bar, make sure there are clear visual indicators for hover and active states—for example, changing a menu item’s background or text color when the user hovers over or clicks it.
6. Add form validation or error messages to provide a seamless user experience.
7. **Take a screenshot** of the **Sign Up** form created in Figma and save it as **signup\_form\_design.png**.

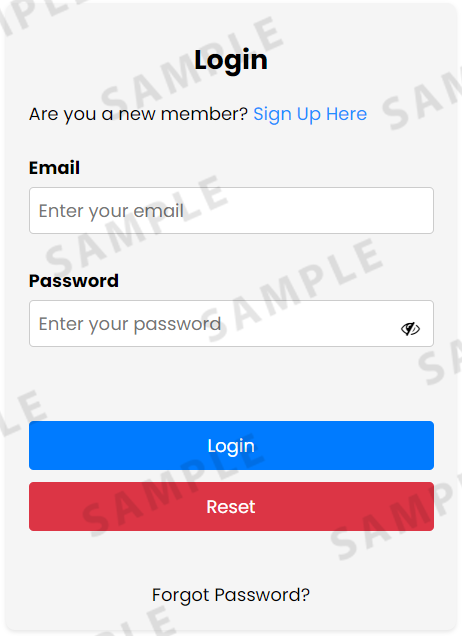
Click here to view a sample layout for the **Sign Up** form.



**Exercise 3: Design the Login form**

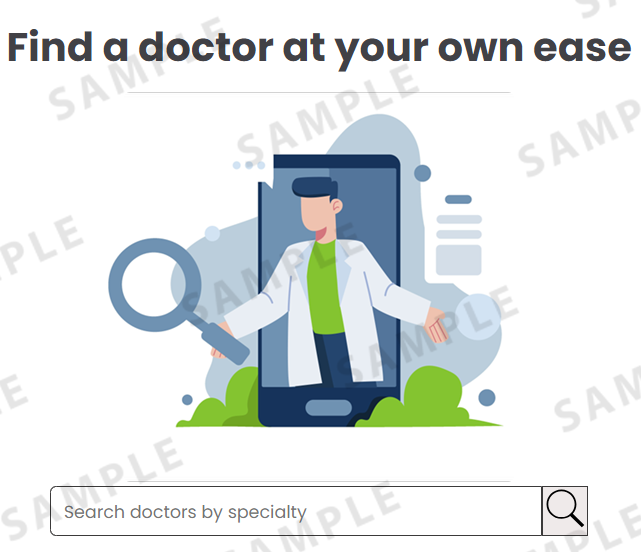
1. Create a frame for the **Login** form.
2. Design the **Login** form layout containing the email and password fields.
3. Use the text and input tools in Figma to create the form elements.
4. Add form validation or error messages to provide a seamless user experience.  
   **Tips:**  
   - Show password requirements when the user sets the password.  
   - Provide user with the option to view the password.  
   - Add validation after all fields have been filled out.  
   - Define the margins.
5. **Take a screenshot** of the **Login** form layout created in Figma and save it as **login\_form\_design.png**.

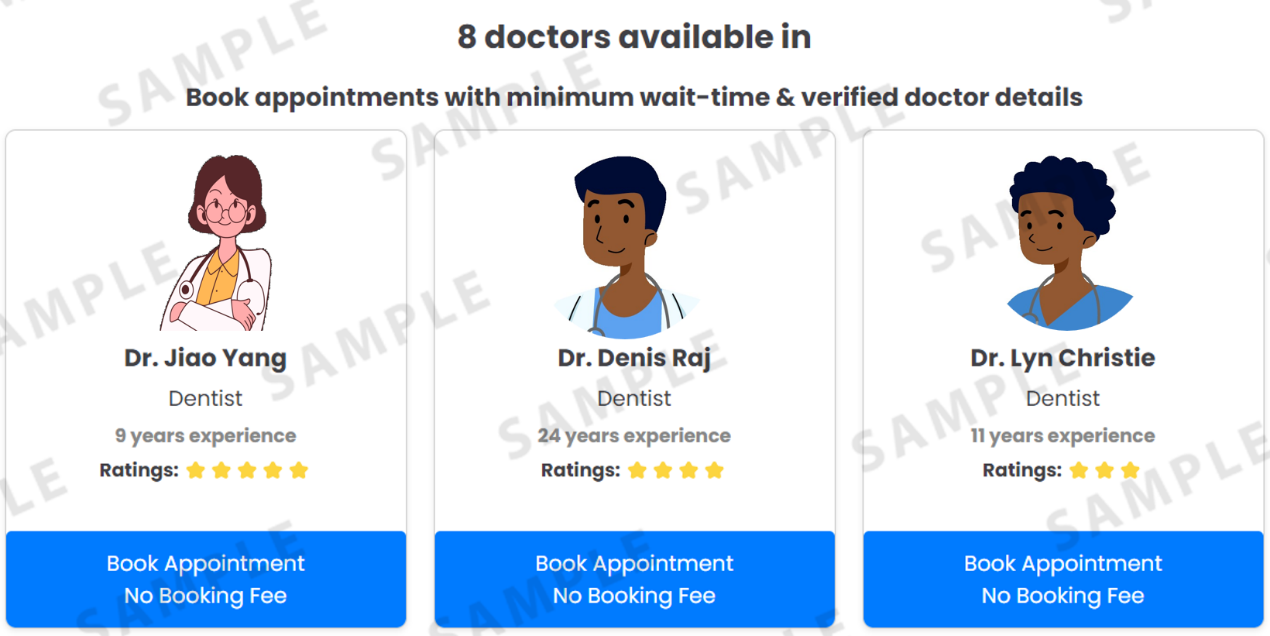
Click here to view a sample layout for the **Login** form.



**Exercise 4: Design the Appointment Booking section**

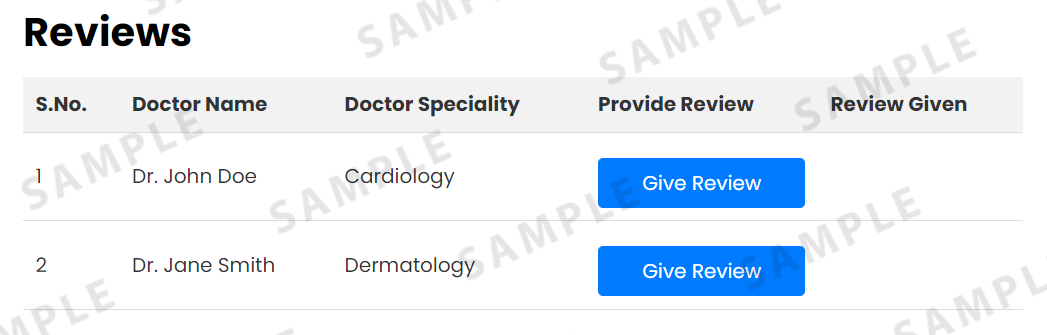
1. Create a frame for the **Appointment Booking** section.
2. Design the **Appointment Booking** layout, which will contain two layouts, one for **Search Doctor** and another for **Doctor Card**.
3. Use the Figma search bar to create the layout for **Search Doctor** to enable users to search for doctors by specific specialty.
4. Design the **Doctor Card** by adding text and other elements for the doctor's name, specialty, experience, and rating.
5. Consider adding interactive elements like buttons to ensure booking submission in the **Doctor Card**.
6. **Take a screenshot** of the **Search Doctor** layout created in Figma and save it as **appt\_search\_design.png**.
7. **Take a screenshot** of the **Doctor Card** layout created in Figma and save it as **appt\_doccard\_design.png**.

Click here for samples from the **Appointment Booking** page.



**Exercise 5: Design the Reviews section**

1. Create a frame for the **Reviews** section, where patients can write reviews about their consultation experience with the doctor.
2. Design the **Reviews** section layout to display user reviews and ratings.
3. Use the Figma design tools to create visually appealing elements for each review.
   1. The **Reviews** section must have doctor name, doctor specialty, a button to allows the patient to provide feedback, and a column to indicate whether or not the patient has written a review.
   2. The review form must have the patient's name and a text block where the patient can write their feedback. If applicable, consider incorporating star ratings, user testimonials, and images.
4. Ensure spacing and alignment for a visually appealing display of the reviews page.
5. **Take a screenshot** of the **Reviews** section created in Figma and save it as **reviews\_design.png**.

Click here for a sample of the **Reviews** section layout.

**Screenshot checklist**

You should have taken the following screenshots as part of this lab:

* *navbar\_design.png*
* *signup\_form\_design.png*
* *login\_form\_design.png*
* *appt\_search\_design.png*
* *appt\_doccard\_design.png*
* *reviews\_design.png*

**Problem Statement**

StayHealthy conducted an extensive survey on medical facilities in remote areas and discovered that:

* Patients struggle to find the right doctor at the right time.
* There are usually **no doctors or hospitals within a 50-mile radius**, or the wait time to meet a doctor is too long.
* These challenges have led to **life-threatening emergencies and even loss of lives**.

With increasing **internet and broadband availability** in remote areas, this initiative will help connect patients with general physicians and specialists **at the right time**.

**Project Requirements**

The website must be a **user-friendly, intuitive, and accessible platform** that connects patients with doctors and specialists for consultations and medical prescriptions.

**Key Features**

**1️⃣ Medical Appointments Online**

* Patients can **schedule appointments** with doctors anytime, anywhere.
* **On-demand consultation** with doctors should be available **without prior appointments**.
* Patients must be able to **view doctor availability** and **select a preferred time slot**.
* Patients must receive **reminders/notifications** for upcoming appointments.
* Patients should have the ability to **cancel or modify** an appointment.

**2️⃣ Doctor Listing for Patients**

* Patients can **view available doctors** with their ratings.
* Patients can **search for doctors** by name.
* Patients can **view doctor ratings and reviews**.

**3️⃣ Consultation Feedback**

* Patients should be able to **rate consultations** based on effectiveness, diagnosis, and communication.
* This feedback system will contribute to transparency and help other patients make informed choices.

**4️⃣ Profile Management**

* **User registration** is required to book appointments.
* Users must provide **mandatory details**:
  + Name, address, phone number, age, blood group, and existing medical records.
* Users should be able to **update their details** at any time.

**5️⃣ Access to Patient Records**

* Only **doctors and patients** should have access to medical records.
* Patients should be able to **download test reports and medical records**.

**6️⃣ News Feeds and Training**

* **Educational videos** on various health topics, sorted by category.
* **Daily tips** on healthy lifestyles and preventive measures.

**7️⃣ Navigation**

* The website must be **easy to navigate** with minimal clicks.

**8️⃣ Website Administration**

* Admin users should be able to:
  + **Manage patient and doctor profiles**, ensuring system security and integrity.
  + **Generate reports & analytics** on appointment bookings, patient demographics, and healthcare provider performance.

**Phase 1 Development**

The **Executive Director** has requested that the first phase focus on the following features:

✔ **Medical Appointments Online**  
✔ **Doctor Listing for Patients**  
✔ **Consultation Feedback**  
✔ **Profile Management**  
✔ **Navigation**

**Your Role & Expectations**

For Phase 1, you are expected to:

✅ **Design the UI/UX** for the website using **Figma**.  
✅ **Develop static and responsive pages** using **HTML, CSS, and React.js**.  
✅ **Deploy the website**.

**Development Approach**

You must **follow the application development lifecycle** as outlined in this scenario. Start by **thoroughly analyzing** the requirements before moving to the next phase.

🔹 *Note: You can enhance the front-end UI throughout the development process.*

This version improves **clarity, readability, and organization**, making it easier to follow. Let me know if you need any tweaks! 🚀