Name: Ismail Lakhani

Q1. Few techniques which you can use to make responsive websites as a frontend developer?

Ans:

With the increasing request from mobile users, it is essential to have a responsive design for the website. However, A responsive design is different from a mobile-friendly design. In later, the website looks good on mobile devices but have glitches when viewed on higher resolution. A responsive website is a website design that allows your website to adapt to the size of any screen it is being viewed on.

Although developer can tweak UI with proper CSS design, the contents mustn't overlap and bleed off. Here are few techniques which can be used to make a responsive website:

1. Work in Phases:

- Have a clear idea about the need and the end-user of the website. Then divide the crucial components. Like Content is the higher priority as we don't want our buttons to hide the information on the page when viewed on lower resolution devices.

2. Content Design

- As a Frontend developer, we need to make sure the navigation looks good on desktop and mobile devices. Sometimes that means writing a completely new mobile navigation or adjusting current navigation to look good on mobile and tablet devices.

- This also means adjusting the content blocks keeping the following cases in mind:
    a. Written Information and content visibility
    b. Buttons/Navbar not overlapping and blocking another content.
    c. User able to use touchscreen and getting the correct response

3. Mobile View first

- Ideally, it is necessary to design mobile UI first, considering how mobile users hit the website these days.
- However, we can also adjust the desktop view to look good and adapt to the device based on the screen size.

4. Image and Media services

- After the blueprint is ready and the website has a structure, we need to optimise the website's media.

- This means the image is loaded as big as it is supposed to and optimised for the particular device accessing the website.
- The background images and videos are not overlapping and restricting the User to use other components.

5. Using Template

- Bootstrap is the most popular CSS framework for developing responsive and mobile-first websites. Its open-source and free yet features numerous HTML and CSS templates for UI interface elements such as buttons, forms. It also supports JavaScript extensions.
- In React, we don't use traditional pure CSS; instead, we use CSS in Javascript.
- Either adopt a hybrid approach using a global CSS specific to your project (which causes limitations in reusability for other components) or start adding styles to each of your Components in their JSX templates.
- Styling the website is also essential> keeping the styling to a minimum

6. Additional Libraries

- "This library provides a wrapper component with a familiar interface to media queries."
- Using additional libraries like Mixing, Decorator and JResponsive enable the developer to use existing components without modifying in any way, keeping their state and properties the same.

Q2. Testing website's responsiveness

Ans:

Tools:

1. Chrome > inspect > Elements > 'Devices' icon top left corner or Keyboard shortcut ctrl+Shift+M (for Windows) and cmd+ Shift+ M (Mac)

    a. This shows the website on different devices from the list and also allows to test the website in landscape and portrait view
2. Websites
    a. http://ami.responsivedesign.is > This website is free to use and allows the developer to test the UI in Big Screen Monitor, Laptop, Tablet and mobile view. The only thing developer needs to do is add the URL of their website and it gives the responsiveness report.
    b. Other websites are ScreenFly.com, Responsinator and Pixeltuner.

3. The easiest way to test the responsiveness is by adjusting the window size of the browser and analyse if the contents are overlapping or adjusting as per screen size.

Q3. Favourite Text Editor

Ans:

1. My favourite Text Editor in Visual Studio Code. It is developed by Microsoft and it has everything I need for development.
2. It offers a lot of themes and tools to make developing fun and easy.
3. IntelliSense: It is an extension feature that is for auto-completing and giving hints about the code. It not only gives me suggestion but also gives more info about the functions.
4. Apart from that I also use Debugging, Build-in-Git, Live server, and Sass Extensions.
5. It is a simple and easy layout. As I use an extended monitor I can view my code side by side for comparison and debugging.
6. Easy to customise
7. There's also Search, create file and folder and other basic features that make it all rounded ideal text editor.
8. It also has a built-in terminal so I don't need to open my main terminal, change the directory and then run commands.
9. This terminal can be open for each active workspace which makes it easy to manage open applications in OS.
10. The next thing I like to VS code is to build in shortcuts to speed up coding. For example '!' to get the structure of the HTML document ready. Similarly, I can use such shortcuts from wrapping lists, image source and more.
11. It is available in all the OS and hence I can have a familiar developing interface in either Windows or Mac.

Q4. Keyboard Shortcut

Ans:

1. For MacOS:
   Option + shift + k (⌥⇧K) for Apple Logo

2. For Windows OS:
   windows logo key and 'L' key to lock the computer screen

Q5. Last Article I read related to IT

Ans:

I am very enthusiastic about Artificial Intelligence and Automation. Recently, I read the article 'Is Your Face the New Airline Boarding Pass?' published on Aviation Today.

It explains how Biometric technologies, anything from fingerprints to face scans for identifications, are being floated as a solution at the airport existing the life under COVID-19 restrictions. After reading the entire article, I believe we humans will not give up on the 'Survival of the Fittest' race. Using the knowledge of Artificial Intelligence and today's computational power, we eliminate the traditional boarding pass at Airports.

This advancement helps in the safety of passengers and is also helpful in reducing paper waste, understanding that climate change is real. Various airports in the US and China deployed contactless biometric technology known as SITA Smart Path. The SITA Smart Path technology allows passengers to move throughout the airport using just their faces as their boarding pass or passport. This made the onboarding process easy and quick and boarded 400 passengers in just 20 minutes.

With such progress and the use of modern technologies, we are moving towards an acceptance of digital credentials, whether it be on your smartphone or another type of device. Eventually, it will allow us to analyse health status to determine if passengers are safe to travel.

Q6. API experience and what is it for?

Ans:

APIs are Application Programmable Interface and its way for a different program to interact and work in a certain way. It can be used to access the data from third parties. For example, in a restaurant we interact with the waiter to order food, drinks, request to pay bills and more. The waiter acts as an interface and shields from all the complicated information that happens behind the scenes. As a customer, we don't worry about the kitchen and its stocks. Here, the waiter can be assumed as an API for the restaurant. APIs are also used to hide complexity, extend functionalities of the existing system, and act as a gatekeeper to add security and protect data.

Yes, I have experience working with APIs and making HTTP requests. I have used various APIs like Google Maps API, NUIG API for an academic project and also Apple's Internal Fiscal API throughout my projects.

The Apple Fiscal is an internal API that I used to develop the web application to convert Gregorian calendar dates to Apple's fiscal dates and display quarter name, earnings, employee benefits and financial reports. The objective of this application was to help employees evaluate Apple's stock benefits by generating a financial report of the desired fiscal period.

Q7. what other technology or approach you can use to store data on client side?

Ans:

Client-side storage is necessary for quicker web response as the website information is stored close to the user/client. Thus, it leads to a good user experience for example GitHub may store the project list to serve the list faster similarly almost every application stores user's preference.

There are several ways to store data into client-side storage. We can use Web Storage as it provides a way to store key/value pairs in the user's browser. However, the data will not be erased until it is removed by the application or the user i.e. it is persistent.

There are two different types of web storage: Local Storage and Session Storage. Session storage is limited to a particular session and the data will only be persistent until the user is on that window. Thus, if the user closes the tab or the window. Unlike cookies, the session storage data is not sent to the server while making network request calls. Also, it has a larger capacity as compared to cookies.

Local storage is similar to session storage but the advantage is it doesn't erase itself. It is stored in the browser and it has the memory capacity depends on the browser and the specific device being used.

These data stored in local storage is sometimes used for AB testing and it can also be used to optimise the performance of the web app. Getting the data from local storage is faster to fetch than making a network call and getting the data.

Apart from the above-mentioned storage types, there are other client-side data storage APIs like IndexedDB and Web SQL. All the storage APIs follows the same-origin policy due to security concerns. This consist of protocol (HTTP or HTTPS), the host/ domain, and the last thing is the port where the app is hosted.

Q8. Pick a site you are familiar with and list 3-5 things you would do to improve its UI/UX.

Ans:

The one website that I would to improve UI and make it more user friendly is primevideo.com. Although it is a responsive website, few things can be improved for a better user experience.

Firstly, The website has no categorisation of the content as per the language and region. It is essential to get all the demographic data and know the users inside out. Since it's a media streaming service, it is possible that users can be from any region speaking any language. I would try to have a controller which allows user to view content from the desired language and region.

Secondly, there is no definition of how people can use the Prime interface. Before a developer designs a UI, it is necessary to define how people will use it. People use website and apps directly by interacting with the interface and indirectly by interacting with UI elements external to the product. Here, there is no easy method to move around different seasons and episodes of a TV series. I would add a dropdown component that can allow the user to browse through different seasons and episodes of the series or TV shows.

Lastly, the Prime website and Prime app on smart TVs and devices are scattered and disjointed. The Prime app mixes everything with no set categories section and mixing paid content with Prime content. This can be improved by the addition of simple progress

information specific to the device. This will enrich different users of a single subscription about different device sessions.