Web Application Architecture

Let’s start with making sense of a web application. A web application is typically accessed using a web browser. A web app end user starts by launching a web browser such as chrome, edge or firefox, inputs a web address, called a URL, in the address bar of the web browser and hits the enter key. A web page then loads onto the web browser. What exactly is going on in between hitting the enter key and the loading of the web page?

### Where is the web page stored?

Web pages are stored on a remote computer, named the web server, connected to the Internet. Internet is the largest network in the world. There are several web servers connected the Internet such as google.com, facebook.com, etc. The web servers host several web pages stored on it and each web page has a unique address. This address is called Uniform Resource Locator, URL in short. Here is an example URL <http://www.google.com/search?q=web+app>. Let’s break this URL into its components.

1. **http** is the protocol
2. **://** is the protocol separator
3. [**www.google.com**](http://www.google.com) is the domain name
4. **/search?q=web+app** is the URI

### How does the web browser fetch the web page?

Using the URL, the web browser connects to the remote web server using TCP/IP. Once the connection is established it makes a request for the web page. The web server then reads the web page from its storage and sends it as a response to the browser. Browser receives the response, processes the web page data and renders it in the browser.

### How does the web browser connect to the web server?

The web browser connects to the remote web server using TCP/IP. The browser creates a TCP socket connection to the web server. To create a socket connection the browser would need to know the IP address and the port number of the remote web server. Upon a successful TCP connection with the web server, the browser is able to send a request for a web page.

### How does the browser know the IP address of the web server?

A Domain Name System, aka, DNS is a database that stores all the domain names registered on the Internet. It is a hierarchical database and each domain name has several records. The record which provides the IP address for the domain name is the A record.

The web browser takes the URL given in the address bar and extracts the domain name from the URL. Let’s take the URL <http://www.google.com/search?q=web+app> as an example. The domain name in the example URL is [www.google.com](http://www.google.com). The browser then connects to the DNS sends a IP address lookup request for the domain name [www.google.com](http://www.google.com). The DNS responds to the browser’s lookup request with IP address mapped to the domain name.

### How does the browser know the port number of the web server?

To make a TCP socket connection to the web server, the browser needs both the IP address and the port number. The standard HTTP port number for web servers is 80. The browser uses port 80 to connect to the web server. It is possible that the web server is running on a non-standard port such as 8080 or 8888. In such cases, the port number should be part of the URL. For example, if google web server is running on the non-standard port 8080, the URL would looks like this, [www.google.com:8080](http://www.google.com:8080). In such cases, the browser will use the specified port number instead of the standard HTTP port 80.

### Is what format does the browser send the request to the web server?

The user provides a URL to the web browser. Using the URL the web browser has to construct a request string which should be sent to the web server.

The format of the request string is specified in the Hypertext Transfer Protocol, aka HTTP. The browser is required to follow the HTTP specification. Otherwise the web server receives the request but it won’t understand it and treats it as a bad request.

Similarly, the web server’s response string should also follow the specification given in the HTTP. Otherwise the browser won’t understand the web server’s response.

### How does the browser render the web server’s response?

Web server sends a response header called Content Type. The Content Type indicates if the response has a plain text, HTML text, JavaScript, CSS, Images, PDFs, Videos and so on. Based on the content type the browser uses its built in capabilities to parse the content, draw the graphics and render the content to the screen. If there is no built in capability to render the content then it opens an appropriate software to render it. For example, for a long time most browsers couldn’t open PDF files and when the content type is PDF it would open it in Acrobat Reader.

### What type of content can the web server host?

The web server can broadly host static content, dynamic content and active content. Static content is files that don’t change over time. It can be HTML pages, Images, PDFs, Office files, etc. Dynamic content is generated on the fly by software programs. For example, a report with data retrieved from a database. Active content is something that has real time user interaction. For example, network game and video chat.