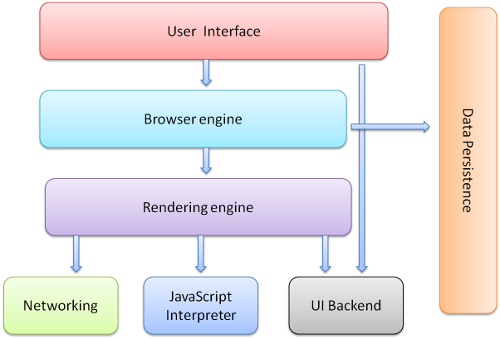
**Exercise 1.1**

When a user enters an URL in the browser, how does the browser fetch the desired result ? Explain this with the below in mind and Demonstrate this by drawing a diagram for the same.(2-3hours)

1. What is the main functionality of the browser?
2. High Level Components of a browser.
3. Rendering engine and its use.
4. Parsers (HTML, CSS, etc)
5. Script Processors
6. Tree construction
7. Order of script processing
8. Layout and Painting

**Solution**

1. **What is the main functionality of the browser?**The main functions of web browser is to fetch or retrieve informative resources from World Wide Web to the client/ user on demand, translate those files received from web server and display those content to the user and allow the client /user to access all other relevant resources & information via hyperlinks.
2. **What are the High-Level Components of a browser?**  
   These are the High-level components of a browser:  
     
   User Interface  
   It is an environment allowing users to use certain features like search bar, refresh button, menu, bookmarks, etc.  
     
   Browser Engine  
   The bridge connects the interface and the engine. It monitors the rendition engine while manipulating the inputs coming from multiple user interfaces.  
     
   Networking  
   The protocol provides an URL and manages all sorts of safety, privacy and communication. In addition, the store network traffic gets saved in retrieved documents.  
     
   Data Storage  
   The cookies store information as the data store is an uniform layer that the browsers use. Storage processes like IndexedDB, WebSQL, localStorage, etc works well on browsers.  
     
   JavaScript Interpreter  
   It allows conversion of JavaScript code in a document and the executes it. Then the engine shows the translation on the screen to the users.
3. **Define Rendering Engine and explain it’s use.**This component is responsible for rendering a specific web page requested by the user on their screen. It interprets HTML and XML documents along with images that are styled or formatted using CSS, and a final layout is generated, which is displayed on the user interface.  
   Graphical user interface, diagram

   Description automatically generated
4. **Explain Parsers (HTML, CSS, etc).**In the syntax analysis phase, a compiler verifies whether or not the tokens generated by the lexical analyzer are grouped according to the syntactic rules of the language. This is done by a parser. The parser obtains a string of tokens from the lexical analyzer and verifies that the string can be the grammar for the source language. It detects and reports any syntax errors and produces a parse tree from which intermediate code can be generated.  
   Diagram

   Description automatically generated
5. **What are Script Processors?**The script processor executes Javascript code to process an event. The processor uses a pure Go implementation of ECMAScript 5.1 and has no external dependencies. This can be useful in situations where one of the other processors doesn't provide the functionality you need to filter events.
6. **What do you mean by tree construction in HTML ?**  
   The input to the tree construction stage is a sequence of tokens from the [tokenization](https://www.w3.org/TR/2011/WD-html5-20110525/tokenization.html#tokenization) stage. The tree construction stage is associated with a DOM [Document](https://www.w3.org/TR/2011/WD-html5-20110525/infrastructure.html#document) object when a parser is created. The "output" of this stage consists of dynamically modifying or extending that document's DOM tree.  
   Diagram

   Description automatically generated
7. **What is the Order of script processing in browsers ?**The browser will execute the scripts in the order it finds them. If you call an external script, it will block the page until the script has been loaded and executed.  
   Also, Dynamically added scripts are executed as soon as they are appended to the document.
8. **Discuss Layout and Painting in browsers?**The layout (also called reflow) peace will be in charge to calculate the positions and dimensions of each node on the screen. For instance, if you rotate your phone, or if you resize your browser, the layout peace will be executed.

Finally, now that we know which nodes are visible, and their computed styles and geometry, we can pass this information to the final stage, which converts each node in the render tree to actual pixels on the screen. This step is often referred to as “painting”, “rasterizing.” or “repainting”.

**When a user enters an URL in the browser, how does the browser fetch the desired result?**These are the steps taken after a User hits enter on a URL in the browser:

>> You type maps.google.com into the address bar of your browser.

>> The browser checks the cache for a DNS record to find the corresponding IP address of maps.google.com.

>> If the requested URL is not in the cache, ISP’s DNS server initiates a DNS query to find the IP address of the server that hosts maps.google.com.

>> The browser initiates a TCP connection with the server.

>> The browser sends an HTTP request to the webserver.

>> The server handles the request and sends back a response.

>> The server sends out an HTTP response.

>> The browser displays the HTML content (for HTML responses, which is the most common).