What is the internet? - The Internet is a worldwide network of networks that uses the Internet protocol suite (also named TCP/IPfrom its two most important protocols).
The Internet is a way to connect computers all together and ensure that, whatever happens, they find a way to stay connected.
At its most basic, the Internet is a large network of computers which communicate all together.

What is the World Wide Web? - The *World Wide Web*—commonly referred to as **WWW**, **W3**, or **the Web**—is an interconnected system of public webpages accessible through the Internet. The Web is not the same as the Internet: the Web is one of many applications built on top of the Internet.

What are NETWORKS? - When two computers need to communicate, you have to link them, either physically (usually with an Ethernet cable) or wirelessly. Such a network is not limited to two computers. You can connect as many computers as you wish.

What are SERVERS?- Computers connected to the web are called **clients** and **servers**.

- **Clients** are the typical web user's internet-connected devices (for example, your computer connected to your Wi-Fi, or your phone connected to your mobile network) and web-accessing software available on those devices (usually a web browser like Firefox or Chrome).
- ***Servers are computers that store webpages, sites, or apps.*** When a client device wants to access a webpage, a copy of the webpage is downloaded from the server onto the client machine to be displayed in the user's web browser.

What are ROUTERS?- Each computer on a network is connected to a special tiny computer called a *router*. This *router* has only one job: like a signaler at a railway station, it makes sure that a message sent from a given computer arrives at the right destination computer. To send a message to computer B, computer A must send the message to the router, which in turn forwards the message to computer B and makes sure the message is not delivered to computer C.

By connecting computers to routers, then routers to routers, we are able to scale infinitely. Such a network comes very close to what we call the Internet.

What are PACKETS?-

the format in which the data is sent from server to client. Basically, when data is sent across the web, it is sent in thousands of small chunks. There are multiple reasons why data is sent in small packets. They are sometimes dropped or corrupted, and it's easier to replace small chunks when this happens. Additionally, the packets can be routed along different paths, making the exchange faster and allowing many different users to download the same website at the same time. If each website was sent as a single big chunk, only one user could download it at a time, which obviously would make the web very inefficient and not much fun to use.

Metaphor for the internet and web -

I would consider a ROAD system as a great metaphor.

There are high speed interstates that connect all the secondary and local roads around the country. Each house has a driveway that connects it to the vast road network.

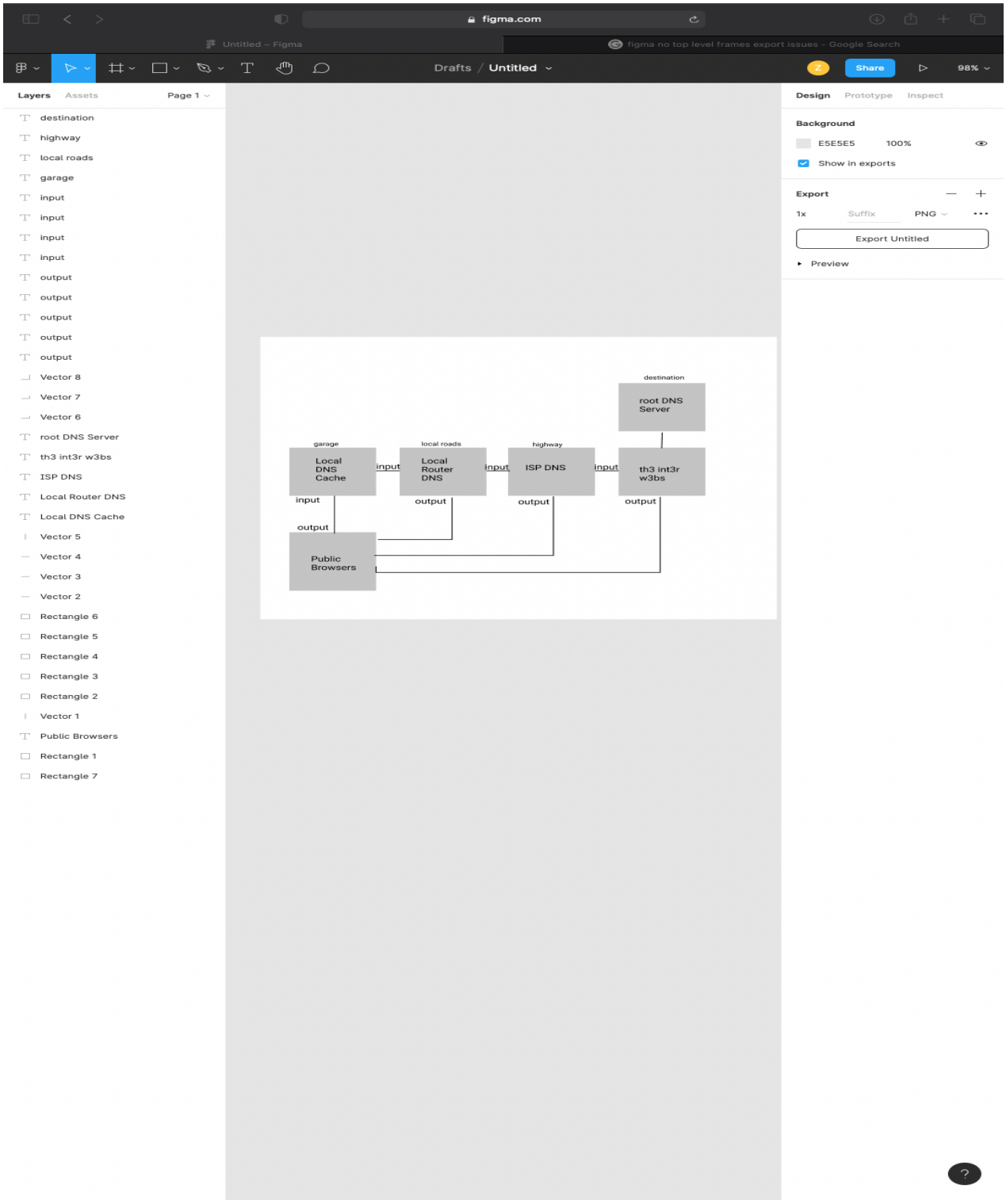
There is a consistent naming convention (N-S roads have odd numbers and E-W roads have even numbers.

has standards for how wide the roads are and speed restrictions.

No one entity is responsible for managing all the road networks. All types of vehicles can use the road. Trucks (streaming music), cars (email), buses (video) can all use the same medium yet contain different data.

When a link is closed (accident or construction), traffic can be routed to use an alternative path to reach its destination.

Also, new roads can be added and easily integrated into the existing system.


figma.com
Untitled – Figma
figma no top level frames export issues - Google Search
Drafts / Untitled
Share
98%
Layers
Assets
Page 1
destination
highway
local roads
garage
input
input
input
input
output
output
output
output
output
Vector 8
Vector 7
Vector 6
root DNS Server
th3 int3r w3bs
ISP DNS
Local Router DNS
Local DNS Cache
Vector 5
Vector 4
Vector 3
Vector 2
Rectangle 6
Rectangle 5
Rectangle 4
Rectangle 3
Rectangle 2
Vector 1
Public Browsers
Rectangle 1
Rectangle 7
Design
Prototype
Inspect
Background
E5E5E5
100%
Show in exports
Export
1x
Suffix
PNG
Export Untitled
Preview
destination
root DNS Server
garage
Local DNS Cache
input
local roads
Local Router DNS
input
highway
ISP DNS
input
th3 int3r w3bs
input
output
output
output
output
Public Browsers

Addresses and Domains

1. **What is the difference between IP address and Domain name?** - IP address is the number, or numerical instructions to a website or page, while a domain name is the simpler and easier to understand LINK to the website.
2. **What is Devmountain.com 's IP address?** - 172.66.43.107
3. **Try to access devmountain.com by its IP address. It shouldn't work because we have our sites protected by a service called CloudFlare. Why might it be important to not let users access your site directly at the IP address?** - Malware attempts, or if the IP address changes, the Domain name will stay the same saving confusion for users.
4. **How do browsers know the IP address of a website when we type in its domain name?** -It checks local DNS Cache to see if we've visited the site before, then our local router DNS to see if any other devices have been to the site, ISP DNS if both those fail, and then all the way to the root server if it still can't find what it's looking for.

How a web Page loads Into a Browser

1. Initial request (link clicked, URL visited) - A page load begins when a user selects a hyperlink, submits a form, or types a URL in a browser. This is also referred to as the initial request or the navigation start. The user's action sends a request across the network to the web application server.

2. Request Reaches App Server - The request reaches the application for processing. (The request may take some time to start being processed. This could be the result of request queuing or it could be other factors.)

3. App Code Finishes Execution - The app finishes processing and sends an HTML response back across the network to the user's browser. This is sometimes referred to as response start or first byte.

4. Browser receives HTML, begins processing - The user's browser receives the HTML response and starts to process the Document Object Model, or DOM.

5. HTML processing finishes - The DOM finishes loading; this point is known as DOM ready. Using the DOM, the user's browser starts to render the page.

6. Page Rendered in Browser - The page finishes rendering in the user's browser and the window load event fires. (For pages that use asynchronous loading, some elements may continue to load after the window load event occurs.)

PART A - GET /

1. The h1 and h2 tag
2. html
3. yes, looked at server.js and saw a msg with in an <h1>Jurrni</h1> <h2>Journaling your journies</h2>
4. Because it was written in the server.js

PART B - entries

1. You see the entries
2. The object including entries
3. Yes
4. Looking at the object in the server

PART C -

1. Pushing an entry into the entries “object” or “class”
2. id , date, content
3. Curl -i -X POST -H ‘content-type: application/json’ -d ‘{“date”:”September 27”, ”content”:”Hello”}’ http://localhost:4500/entries
4. Curl -i -X POST -H ‘content-type: application/json’ -d ‘{“date”:”September 27”, ”content”:”Hello”}’ http://localhost:4500/entry
5. A new entry
6. Strings of entries
7. yes
8. yes