Forest H

Weekly report 1

NodeJS notes

About NodeJS: often used to build API, back end services, data intensive and scalable, fast efficient coding , good response time, low file sizes, largest opensource libraries.

A runtime environment for executing JS code outside a browser, JavaScript code is usually handled by a browser’s engine, each has their own. Ryan dahl took the v8 engine(from google) made node.exe with v8 in c++, basically enhancing it to work with file systems and networks. Synchronous architecture/blocking gets a server request and a thread is allocated to that request (that thread becomes busy). If out of threads new requests have to wait. Node applications are asynchronous by default, uses event queues which is good for I/O intensive apps, however node is single thread based and so is not good for cpu intensive applications. Mostly the first week’s work is just exposure tied to write some notes in the code too to show I’m looking at things not just writing a couple lines, writing lots of the hello world type programs would sort of be a waste, just need to learn language content quickly so I can do some real things.

Udacity notes

TCP/IP basics

Communications functions can be divided into subset actiiviites , each activity can be looked at as a layer. Layers provide a specific service(function) to other layers, a layer has a specific protocol(control messages) to connect other systems. There are two standerds SNA(IBM’s proprietary) and ISO’s OSI model. OSI has upper and lower layers, lower consisting of Networks, data links, and physical links/mediums. Respectively they are for choosing the next node and linking to it, controlling the flow of messages on a link, the physical connecting providing the link. Upper layer consists of Applications, presentation, sessions, and transport. Again, respectiviely they do: providing services to an app, present info to a user that they will understand, controls user to user dialogue, raises quality of service provided by network to the level required by the user.

Nice guide I added for my own reference really below, very little coding done in first weeks work it seems, could probably write another 5 pages at least but as this is supposed to be a summary ill try to quickly mention some of the lecture topics, hopefully in the future work shown will be more and writing about the same or less.

Topics: Ip addresses dynamic and static, ipv4 vs ipv6. Ips are connectionless protocols and unreliable. ARP & ICMP protocols assist IP. TCP connection oriented and reliable, implements flow control, sends data using IP. Ports and IPs used together is a socket, ports used to map TP data for a process, such as port 443 for HTTPS. TCP communication steps, server listens, client request connection, SYN flag set, server responds, clients acknowledge connection via ACK flag being set. TCP is sliding window and has some self-error correction. TCP needs to be closed on both ends. UDP is a simple protocol, unreliable and connectionless, specifies upper layer protocols. Unicast 1-1 multicast 1-n broadcast 1- all.

