**Note Application:** Our application will be similar to discord in which users will be able to communicate over long distances and share files. There will be multiple chat rooms and even direct messaging in the application. Users will have the option to engage in voice calls with one or many other recipients.

The concepts I will be using to explain the application are listed below:

* States
* Networks
* Notational Machines
* Encryption
* Compression
* Machine Learning

**States:** An example of a state would be a simple on and off state (could be 1 for on and zero for off in binary). A state is just a temporary configuration that is normally based on user interaction. If you were to scroll through a website like eBay for example, then the homepage would be considered as a particular state of that website and if you were to click on the checkout basket then the website would enter a different state known as the checkout state. There is a new emerging technology known as Quantum computing which uses superposition states instead of the simple on and off states that you would find in normal computers.

Superposition allows a Qubit (Basic unit of quantum information) to appear in both a 1 and 0 zero state while being unobserved. This is quite powerful because this means that a Quantum computer can hold far more values than a regular computer in turn giving it the potential to decrypt any encryption algorithm in the near future which is a big security risk.

**States of the Note App:** Our Application will have many different states such as:

**Message sent**: This is a state in which the user has sent data to another recipient and the message has been successfully delivered. The user will be greeted with a blue tick icon on their screen to show that the message has been sent.

**Message received:** The user enters a new state in which a message has been received by another recipient. The application will generate a message read icon for both the sender and receiver to let both parties know that the message has been received.

**Potential Error states:** Since our application will require access to the internet then the potential error states when sending a message to another user could be caused primarily by the sender losing his or her internet connection. This would probably lead to the message not being sent as the data would’ve been lost while the application was changing states. The sender will be given an error message to let them know that the message hasn’t been sent to minimize the damage caused by this error state.

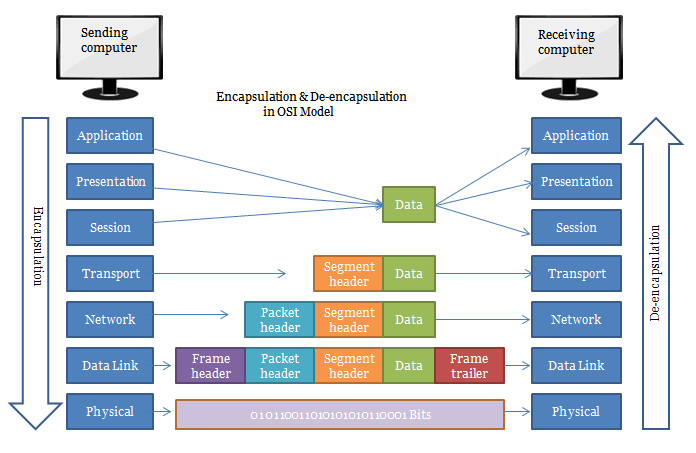
**Networks:** A basic definition of networks is two computers connected together. One example of a network is called a wide area network which means that computers could be connected together from continents away. Networks help us communicate over the internet using network protocols which are standardized rules created to make sure two computers can communicate no matter the brand name. The seven-layer OSI model was originally introduced to help simplify the complex protocols that were used to communicate over the internet but has since been replaced by TCP/IP model which is only condescend into 5 layers. The 5 layers are application, transport, network, datalink and the physical layer.

**Data encapsulation of a message being sent on Note:**

**Application Layer**: Our application when sending data to another user will most likely use the HTTPS protocol which is a more secure version of the original HTTP protocol. The data will be encapsulated meaning it will be compressed and encrypted before being sent to the transport layer.

**Transport Layer**: The Data will be cut into pieces and become a segment on this layer as different layers have a different PDUs (Protocol Data Units) which are just another way of labelling information. A TCP header (contains a source protocol and destination protocol) will be attached to it. The TCP protocol will be used to keep track of data and is more reliable than UDP so the application will use that to keep track of any loss of data.

**Network Layer**: The Segment is sent to the network layer and where an IP header is added (contains source and destination IP address). An IP protocol is used for communication between two nodes. The Segment is converted into a packet PDU before being sent through the datalink and physical layer which will both add their own headers and PDUs.

The encapsulated data with all the headers will be sent through the clients own internal network until it reaches the router. The router will then forward the data using routing protocols like OSPF or EIGRP. The ISP (Internet service provider) will then forward the data to the recipients’ computer which will begin the process of decapsulating the data. Deencapsulation just removes all the headers, encryption, trailers and decompresses the file into a useable format once received by our application on the receiver end.

**Notational machines:**  First created by Benedict Du Bouley in the 1980s. Notional machines are an abstraction designed to help people understand a language or maybe to take complex actions done by a programme and simplify them down to make the actions easier to understand. Notional machines don’t necessarily have to be accurate as the idea is to help you understand what the actual machine does instead of overwhelming you with all the complexities that go into a particular programme.

**Examples of Notional machines in the Note Application:**

**Upload:** This feature allows users to upload music files within chatrooms or directly to other users by simply clicking upload and selecting the mp4 file you wish to upload within a certain file limit.

**Download:** This feature lets users be able to download music files displayed in the chatrooms by clicking a downward arrow button.

**Voice call:** Users will be able to contact and converse with people in real-time over long distances simply by clicking another user’s profile picture and selecting the call icon.

**Potential issues**: Placing a file limit on sizes would be necessary as mp4 files are very large even in their compressed format which could potentially cause a lot of problems for mobile users who have smaller memory spaces than normal sized computers. There would also be a limit to how many files one user can upload at a time to prevent congestion.

**Encryption:** Encryption is a means of securing data as it travels across the internet. In order to secure the data, there is a mathematical algorithm used to scramble what would normally be plain text into an unreadable format known as a cipher text. A cipher text is just a long and hard to guess number that can’t be decrypted easily. To decrypt the data, you will need what is known as a key and this is used to decrypt the data into a readable format. Encryption falls under the science of Cryptography which has been around for thousands of years.

One of the newer ways encryption is being used is in crypto currencies. A virtual currency like Bitcoin allows clients to have multiple public addresses to receive transactions on. These public addresses can be shared online without running the risk of having your wallet being hijacked because as long as you keep the private address that you use to send transactions on hidden then you are safe.

**Encryption in Note App**:

Our application will need encryption for the messages being sent over the internet from one user to another to prevent a man in the middle attack. A MITM attack is where attackers interrupt an existing conversation or data transfer and pretend to be legitimate participants to both users in turn giving them access to the messages being sent by the users. End to end encryption will help us prevent this from happening. This involves sender and the receiver using a shared key to encrypt and decrypt messages that they send and receive. Since the attacker doesn’t have access to the shared key then he or she will not be able to read the messages.

**Compression**: The reason we compress data is because files can get quite big so sending it over the internet can be slower in its non-compressed format. Compression allows us to make those files smaller in turn making data transfer faster. There are two main types of file compression known as lossy and lossless compression. Lossy compression is used to make files such as images smaller by losing redundant information by taking parts of the images that can’t be noticed by most people and removing it in turn making the compression easier. Lossless compression is a technique that does not lose any data in the compression process. It packs data into a smaller file size by using a kind of internal shorthand to signify redundant data.

Using lossy compression can be quite risky because decompressing and compressing files multiple times can be lead to less overall quality each time. Lossless compression is normally used with video files. For example, it’s easy to find sound frequencies that won’t be picked up by the average human or sounds that are being masked by other sounds and remove them.

**Compression in Note App:**

Our application will be using lossy compression since users will be restricted on the types of files they can upload because we don’t want users to be able to upload malicious files. One of the few file types users will be able to upload is the Ogg file type. Ogg was originally developed by the Xiph.Org Foundation to make it easier to share audio files on the internet. This is a free open container format similar to MP3 and was designed to provide efficient streaming and manipulation of high-quality digital multimedia. Mobile users will benefit from the Ogg format since the files will be greatly reduced in size compared to an uncompressed audio file format.

**Machine learning:** Machine learning falls under the study of artificial intelligence and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. If you’ve ever had to do an annoying captcha where you select certain images, then you can think of that as way of training a machine learning algorithm to be able better differentiate between two things or more. One of the ways you can train a machine learning algorithm is through the supervised learning model. This is when you train a model on examples of input and outputs. For example, training a model to take the input of an image of cat and giving it an output of the word cat mapped to that image.

Machine learning is being used in many fields such as advertised. For example, as you shop through ecommerce sites like amazon you’ve probably noticed the targeted ads that are shown to you based on your own specific shopping history. This is done through machine learning as amazon collects massive amounts of data which is pushed into machine learning algorithms to find the items that best suit your own personal needs.

**Machine learning in Notes App**: In our application we will have recommended tabs that will give users music choices based on their own download history from the application. To accomplish this, we will be using the machine learning model known as reinforcement. Reinforcement learning is the training of machine learning models to make a sequence of decisions so much like training a dog you will reward the dog for good behaviour and punish it for bad behaviour. Reinforcement learning was used to beat the world chess champion Kasparov using the software known as Deep blue in 1997. One of the disadvantages to reinforcement learning is that it requires a lot of data and computations.

The reason we are using the download history from the Note application is because this will provide a lot of data based on user activities which will make our model better accurately determine which types of music to recommended to users. For example, if you recently downloaded a lot of rap music then most likely you would be recommended other types of rap music.

Sources:

<https://pythonistaplanet.com/pros-and-cons-of-reinforcement-learning/>

<https://bdtechtalks.com/2021/02/22/reinforcement-learning-ad-optimization/>

<https://www.scientificamerican.com/article/20-years-after-deep-blue-how-ai-has-advanced-since-conquering-chess/>