Department of Computer Engineering

Bilkent University

Senior Design Project

Project Short-Name: Nebula

High Level Design Report

Group Members: *Arda Atacan Ersoy, Kübra Nur Güzel, Seyfullah Yamanoğlu, Başak Şevval Ekici, Serhat Bezmez*

Supervisor: *Eray Tüzün*

Jury Members: *Cevdet Aykanat, Abdullah Ercüment Çiçek*

Innovation Expert: *Duygu Gözde Tümer*

Table of Contents

1. Introduction

Cloud storage has come to rely almost exclusively on large storage providers as trusted third parties to transfer and store data. This system suffers from the inherent weaknesses of a trust-based model. Because client-side encryption is non-standard, the traditional cloud is vulnerable to a variety of security threats, including man-in-the-middle attacks, malware, and application flaws that expose private consumer and corporate data. The main infrastructures of Nebula is not very different regarding to the current systems that are being used to provide cloud storage to users, in terms of the storage, hardware and network components. However, with the unique architecture of the Nebula, it is aimed to provide a different system with enhanced security and privacy.

Nebula is a peer to peer decentralized blockchain based cloud storage system. What does these technical terms mean and why it is an important issue at the moment? Cloud storage becomes more popular everyday but also more people want to keep their data secure and private. With the not so pleasant past events from other centralized cloud storage products, such as data losses or hacking to private photographs of celebrities, the idea of Nebula arose. Nebula offers secure and distributed storage of personal data with the use of remote nodes for requests and configurations as well as a remote data center infrastructure. Each data will be distributed in an encrypted fashion, with the use of blockchain to make the best possible security services up and running.

1.1 Purpose of the System

Nebula offers many advantages compared to data center-based cloud storage. Data security can be maintained using client-side encryption, while data integrity will be maintained via a proof of storage and retrievability that Hyperledger provides[1]. The impact of infrastructure failures and security breaches will be greatly reduced. An open market for data storage may drive down costs for various storage services by enabling more parties to compete using existing devices since with

Nebula users will provide and use the storage themselves. Data on the Nebula network will be resistant to tampering, unauthorized access, and data failures.

Decentralized cloud storage network offers many advantages compared to datacenter-based cloud storage in terms of security, privacy, redundancy in data and cost reduction due to efficiency. Nebula will provide the data security with client-side encryption which is not a standard in the systems that are widely used now since they are datacenter-based models. Also the file loss can be avoided since extra copies will be transmitted in case of errors. With the help of blockchain storage, cloud computing cost can be reduced.

1.2 Design Goals

1.2.1. Security

- Application should keep the data secure and private via a series of client-side processes before it enters the network. To do that, first the metadata required to remotely verify file integrity should be generated, then the data is split into shards, and each shard should be encrypted.
- End-to-End encryption and client-side encryption will be provided for each user from the network.
- The encrypted data should be then sent out into the network, and the integrity of the data should be checked at regular intervals. So unencrypted data should be never exposed, files in this system should be as secure as the user's key management.
- Program should manage this file storage in terms of blockchain technology and files should be separated into different users. So, no one device has an entire file and all the files are copied on numerous devices, so if a hard drives fails your files won't be lost.
- By the aim of decentralized storage, application should not be vulnerable to hacker attack since it is impossible for anyone without the private keys to decrypt the encoded files.
- With the help of blockchain inherited features (immutable ledger, consensus mechanisms), data confidentiality and security will be provided.

1.2.2. Usability

- Application needs to be user friendly and user interface should be understandable for the simplicity of the user commands.
- Applications , peculiarly Android version, must be efficient in terms of memory and CPU usage since mobile phones have limited resources on memory, battery and CPU. The optimal term “efficient” will be decided after some test on different mobile phones.

1.2.3. Cost

- Application should provide lots of hosts for decentralized file storage. Hosts should determine their own prices for renting out their hard drives (similar to the Airbnb model). This ensures that the prices will stay competitive as time goes on.
- Application should provide much less expensive services compared to major players. Application should offer from ⅓ to ⅛ cost of global centralized storage companies. (Per terabyte storage cost, Amazon: 23$, Google Cloud: 20$) [6]

1.2.4. Performance

- The speed of user’s file storage will depend on the performance of local storage who services to our user. In order to compete with centralized system, we need more people to serve as storage disk. As we increase our competition between people who provide disk storage. We can increase our performance.
- Server should response less than 500 ms while doing the peer distribution.

1.2.5. Extendibility

- Application should be suitable for changes because when users demand new features from us, we should have such infrastructure that can provide it.
- Application should provide the payment method for storage and bandwidth as user use it. User should not be restricted to buy very huge storage files unless user needs it.

1.2.6. Marketability

- In terms of other non-functional requirements, decentralized file storage system has some advantages to centralized ones(See Table 1). This is very important to use this application for the users who concerns these non-functional requirements.
- Big database companies are willing to work with decentralized file storage services such as Couchbase, MongoDb, InfluxDb. [8] Thus, there is a demand in the market for such product.

1.2.7. Scalability

- Since user and file data tied on blockchain platform, scalability of the application is tied on the underlying blockchain platform which is currently determined as Hyperledger. With Hyperledger each node has their own responsibilities and that provides an independent scale for each node. [9] Nebula system should also perform in a similar way.
- Scalability of the application should be adjusted considering the cases when limited scalability harms the availability and reliability of the system such as potential overload of the network.

1.3 Definitions, Acronyms, and Abbreviations

TCP: Transmission Control Protocol

P2P: Peer to Peer

API: Application Programming Interface

HTTP: Hypertext Transfer Protocol

GUI: Graphical User Interface is a program interface that takes advantage of the computer's graphics to make the program easier to use.

Client: User End of the Application. For Nebula it is the web application.

Server: The part of the application which responds to Client's requests. Responsible for data management and API interactions. In Nebula distributed network pretends server role.

1.4 Overview

The Nebula platform is a decentralized data processing architecture designed for secure, scalable management of online data storage, file sharing, and user access for individual consumers. This technology seeks to displace single source Cloud Storage Providers (CSPs) [8] and Storage Partitions. The Nebula ecosystem aims to manage security for data-at-rest (storage), and data-in-use (sharing). The ecosystem is structured to provide scalable benefits and incentives for all participants to grow the security, integrity, financial competitiveness and performance of Nebula technology.

Nebula uses Hyperledger technology, which is a blockchain technology hosted by Linux Foundation[10]. This technology will be used to decentralize Nebula's network and databases. There will be two immutable ledger distributed across to the network: One is used for auditing user accounts. The other is used for the auditing files and peers relationship. Inherited blockchain features enhanced Nebula to being immune to attacks or single point failures since the peer nodes will continue to function of keeping blockchain ledger. Moreover, uploading the data on a centralized cloud, the data is distributed across the network. With the immutable ledger, cloud is shared across the all nodes and it is highly encrypted in such manner that is impossible the interrupt. In other words, only owner can access to the file. Thus blockchain technology is useful to decentralize and secure the data. Blockchain consensus mechanism and smart contracts make this interruption nearly impossible. Instead of based on blockchain technology, Nebula combines blockchain ledger with its P2P network, which contributes enhanced security, performance file transfer and decreased cost to the end user. This combination of distributed storage and blockchain provides a verification of the network without any third party.

2. Current Software Architecture

There are many options for cloud storage systems on the market. In this report, current software architecture will be analyzed in two sections regarding to their architectural types: centralized cloud systems and decentralized cloud systems.

Nebula aims to be a decentralized cloud system but it is a fairly new topic so there are not very popular systems to analyze at the moment. With the centralized cloud systems, very popular systems such as Google Drive, Apple iCloud and Dropbox will be discussed.

2.1 Comparing Two Systems

Before intensely analyzing the current systems, a brief overview of centralized and decentralized concepts is below in the given in TABLE 1 in order to understand effectiveness of inherited features of decentralization. Regardless of the specific services offered by companies, we have theoretically compared these two concepts. TABLE 1 is a critical decision point for us to consider pros and cons of the decentralized systems.

	Centralized Cloud Systems	Decentralized Cloud Systems
Points of Failure	Single Point of Failure: If the master server machine goes down, clients are no longer able to process user requests since the machine that runs the core software to process requests is dead. Although the central systems are developed by dominating companies, theoretically single point of failure cannot be overlooked.	No single point: Client machines aren't relying on a single server (multiple nodes).
Maintenance	Easy to maintain as there is only a single point of failure.	Difficult to maintain since there are many participating nodes in the system.

Liquidity	Better market liquidity since it is a seated system in the current market.	Inadequate market liquidity to compete with centralized since it is a new technology in the market.
Scalability	Scale out the system by vertical scaling — by adding more storage, I/O bandwidth, processing power (number of CPUs, number of cores) to the server machine. (Costly solution comparing to decentralized)[17]	Scale-out the system by adding more nodes.[17]
Fault Tolerance / Stability	Theoretically unstable. Single failure can occur on master server machine.	Very stable and has an immunity to a single failure.
Speed	Fast transactions	Slower. Considering the nodes are physically separated in space and encryption, separation and distribution of the data take some finite amount of time. Faster when the node number maximizes and geographical location of nodes are close.[2]
Evolution to new technologies	Based on single framework, which brokes diversity and evolve slowly. [16]	Once the basic infrastructure is in place, evolution is much easier in the structure.[16]

Trust Factor	Trusting a 3rd party with potentially sensitive data comes with all the risks of human vulnerabilities. Mistakes can be made and data can be lost, stolen, or even sold.[17]	The system is trustless (refer to Glossary): Trades are executed peer-to-peer and the exchange relies fully on its users. Encrypted and decentralized storage methods prevent other nodes from accessing data. Even the renter of hard drive space cannot access the information stored inside it.[17]
Ease of development	Fast development. Pick up a framework and apply it [17]	More complex design:Low level details like resource sharing (trade) and communications (transport). [17]
Ease of use	Simple UX and steps.	Complex user experience and confusing steps / transactions.
Service Fee	Not cost-effective for customers. For example, Amazon S3 has a fee $23 per month 1TB storage.[2]	Less expensive for instant Siacoin charges around US $2 for 1 TB per month. [2]

TABLE 1: Comparison Table of Decentralized and Centralized Cloud Systems

How many TB?

1

Storage Provider	Monthly Storage Cost	Download Bandwidth Cost	Private	Decentralized	Included Multi Region Redundancy
Sia	$2	$1	✔	✔	✔
Amazon S3	$23	$92	✖	✖	✖
Google Cloud	$20	$110	✖	✖	✖
Microsoft Azure	$24	$87	✖	✖	✖

Table 2: Comparison Table of Cloud Service Providers [2]

As it is seen from Table 2, 1 TB storage providing cost is shown in the picture. Decentralized cloud Sia has a much less cost comparing to the other major centralized cloud services. [2]

2.2 Centralized Cloud Systems

Centralized Cloud Systems are the most traditional cloud systems currently in the market. It has a single server architecture for multiple clients. The server has more computing resources than its clients, allowing all of the major processing done by one server computer. [3] The following products in this section uses distributed client and server architecture models.

2.2.1 Apple iCloud

- iCloud is a mobile device friendly application.
- iCloud gives a 5GB free storage but for bigger storage usage, the service gets more expensive.
- Compatible with Mac and Windows PC

- It is used for file and document sharing and iCloud can also stores music, photos, videos, and documents.
- 128 bit AES encryption, end-to-end.
- Two factor authentication with an Apple ID.

2.2.2 Google Drive

Google Drive is one of the most used cloud storage services available at the moment. The advantage with Google Drive is that it works synchronized with other Google Services such as Google Documents and Gmail. User can modify documents on their Drive and also work collaboratively with other Google accounts. Google Drive can be used from the web or with the mobile applications. It does not have a desktop application.

- Google Drive is a mobile device friendly application.
- Drive gives a 15GB free storage but for bigger storage user needs to subscribe to a payment plan.
- Does have a desktop application.
- Allows storage for music, photos, videos, and documents.
- Google Drive also allows collaborative working on certain file types
- User can share their files directly with entering an email address
- They use their own software for encryption.

2.2.3 Dropbox

- Dropbox gives a 2GB free storage but paid subscriptions available that offer more capacity.
- Dropbox offers computer applications for Apple macOS, Windows, Linux computers.
- Dropbox has a desktop application.
- Dropbox mobile application is compatible with iOS, Android, and Windows Phone smartphones and tablets.[4]
- The Dropbox software enables users to drop any file into a designated folder.
- Dropbox uses strong cipher for encryption.

2.3 Decentralized Cloud Systems

Decentralized file storage topic is popular even it is a new system to compete with centralized file storage companies. And lots of companies are investing their fundings to the Filecoin ICO and companies that are explained below which try to develop decentralized file storage system. [5] Current up-to-date decentralized cloud storage systems are all using blockchain at the moment. [14]

2.2.1 Interplanetary File System (IPFS) with FileCoin

- It is a protocol to create a new way to server information on the web.
- It allows user to choose where to get content from and user can set privacy of peers user trust to receive his/her files.

- The files are stored in the nodes based on their content. Each node stores only content it is interested in.
- The content can be accessed offline.
- IPFS is an open source project.
- They use Proof of Spacetime and Proof of Replication techniques.
- They did their Initial Coin Offering (ICO) in September 2017.
- It is still in development stage.

2.2.2 SiaCoin

- It is a protocol to create a new way to server information on the web.
- They used Proof of Storage technique on their algorithm.
- SiaCoin has a working product.
- It provides an open source platform.
- Siacoin has a 10 min block time (same as bitcoin) producing 144 blocks a day. [4]
- The deals are secured by file contracts, Siacoin's form of smart contracts. [6]

2.2.3 Storj

- Their technology revolves around file sharing and separates parts of the files to users in the network. When the user requests for the file, Storj locates all the shards and piece them together. [3]
- Storj is a distributed cloud storage which means not all the processing of the transactions is done in the same place.[7]
- The files are encrypted before the separating process and the user who owns the file has his private key for validation.
- Storj claims that they have a working product, but it is not released yet and does not allow new user account registration.
- It provides an open source platform.
- Their paths are encrypted in a hierarchical and deterministic way.

3. Proposed Software Architecture

3.1. Overview

Architecture of Nebula can be shown into two parts as “Client” and “Network” architectures individually. Subsystem decompositions, hardware and software mappings will be described in the following sections. Also, other main components o Nebula system will also be shown and discussed in detail.

The basic architecture of the Nebula system is illustrated in Figure 2. There are basically 3 main components of the system which are described as Backend

Engine, Data Centers and Distributed File Storage network. These components are interconnected each other through backend engine. Backend Engine serves as an API to the web, mobile and desktop clients. User logs and audits are kept in blockchain ledger by using the 3rd party HyperLedger component such that each peer will be has the immutable blockchain ledger. Thus Hyperledger Fabric is an external component corresponds the data centers in the figure.Backend engine simply validates user registration and also their unique keys for their files. Moreover, data gathered from the peers are controlled by backend engine.

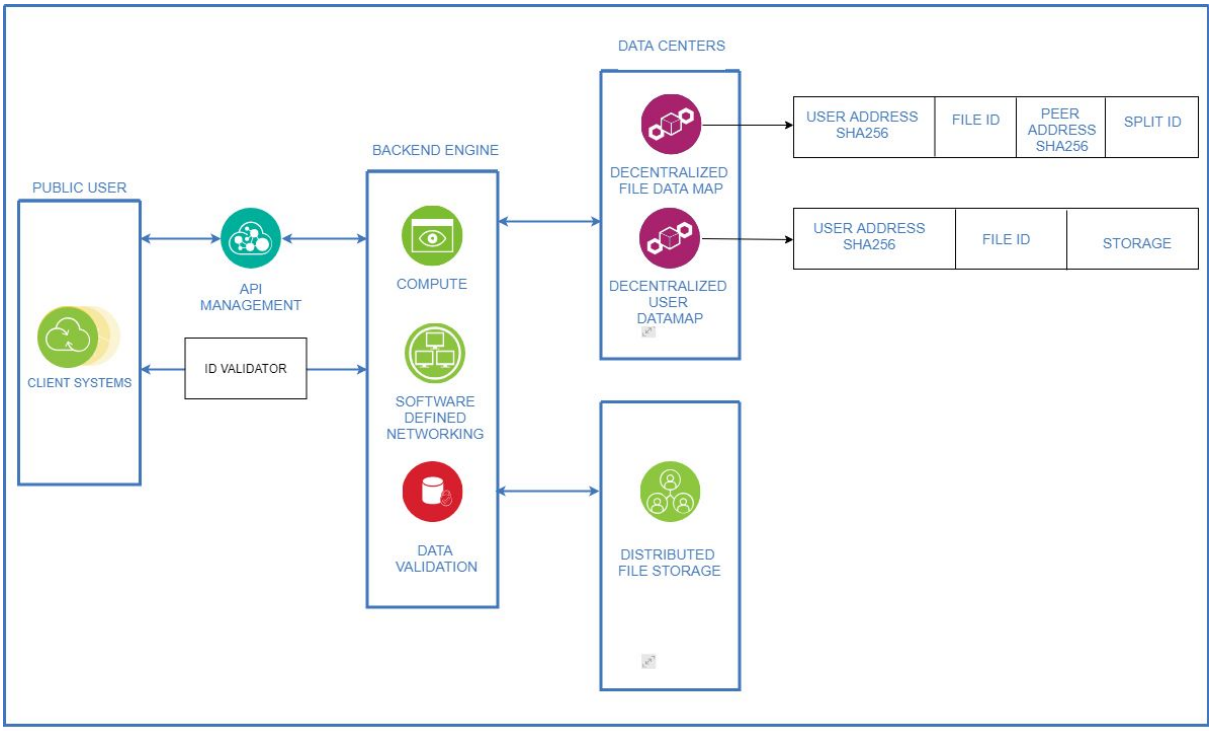

Figure 2: Nebula Drafted Architecture

The difference between blockchain based decentralized cloud:

Following table (TABLE 2) compares the key aspects of a decentralized peer to peer cloud network and a blockchain based decentralized cloud network. Overall, especially from the security perspective, blockchain usage in the network would increase the reliability of Nebula system.

	Decentralized P2P Cloud Network	***Blockchain based Decentralized Cloud Network***
Need of a central server	*No*	*No*
Speed and efficiency	*Nodes which hold user's file are known by the system so retrieving the file is fast*	*Nodes which hold user's file should be mined from the chain first to retrieve the file and this process spends some extra time*
Security	*This system is all about copying the information and this weakens the security.*	*Blockchain prevents copying of information*
Reliability	*Information could be lost in case of seeders quitting.*	*Because of distributed ledger information is never lost*
Anonymity	*In order to make the anonymous P2P system usable and reliable, some system designs make tradeoffs between anonymity and performance such as reliability, latency and throughput.*	*Blockchain is verifiable by public but yet anonymous. Anonymity is one of the most important features of blockchain system. (One of our experts Tuna Orbay confirmed that in order to achieve anonymity; blockchain must be used in our project.)*

TABLE 2: Comparison Table of Blockchain Based and Regular Decentralized Cloud Systems

3.2. Subsystem Decomposition

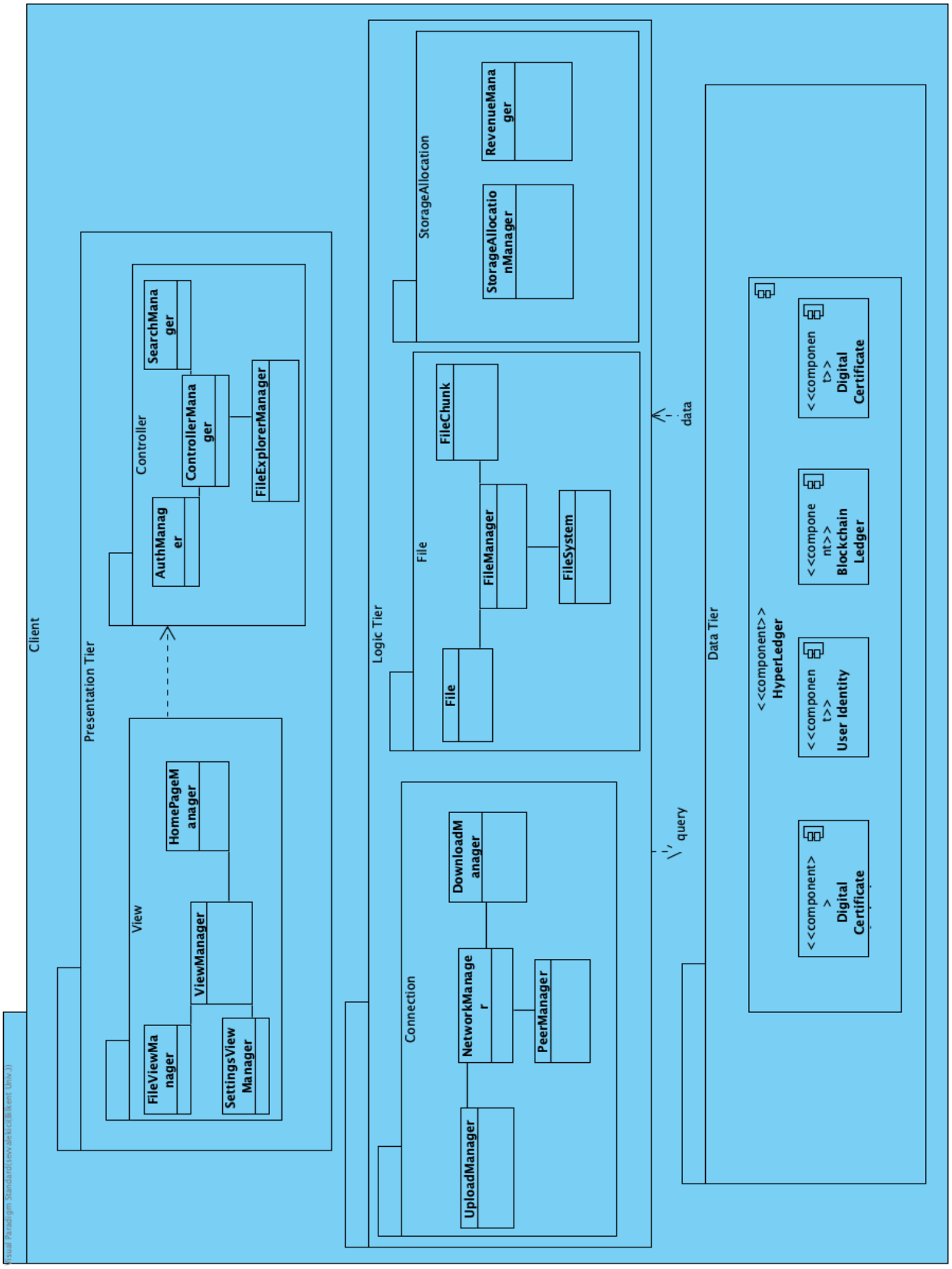

Figure 3: System Decomposition of Nebula

Our system is formed with a 3 Tier Decomposition. 3-Tier Architectures are often used with similar systems where Nebula is a P2P application. By implemented with this architecture we gained a greater flexibility to us by giving allowance to update accordingly independent to other parts.

Presentation Tier is the frontend tier of our system and consists of User Interfaces. View and Controller Packages are in this tier where they have different classes under each of them. Further explanation for each class will be given in *Section 4 Subsystem Services.*

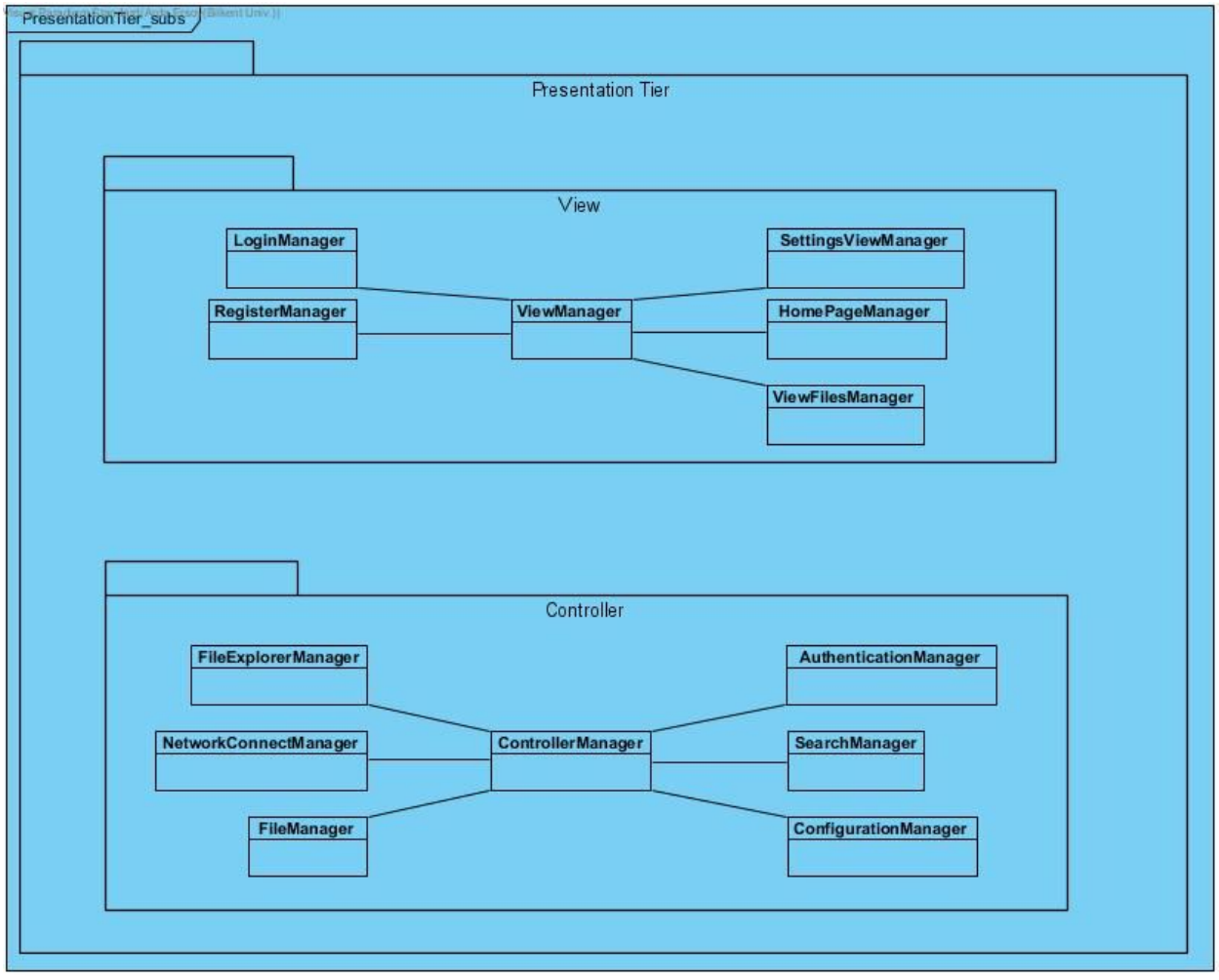

Figure 4: Presentation Tier Decomposition

Logic Tier performs detailed processing and controls functionality of our system. Nebula's Logic Tier consists of Connection, File and Storage Allocation Packages. Detailed explanation for each class under these packages will be explained further in the document.

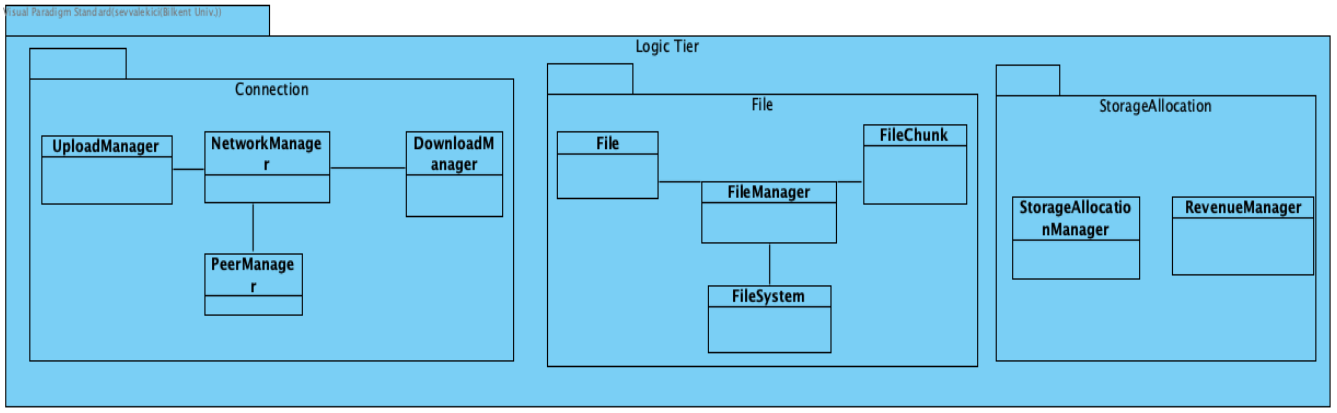

Figure 5: Logic Tier of Nebula

Data Tier includes data persistence mechanism, in our case this is done by HyperLedger Fabric structure. Hyperledger Component in our system has 4 other components underneath which will also be explained in detail in section 4.

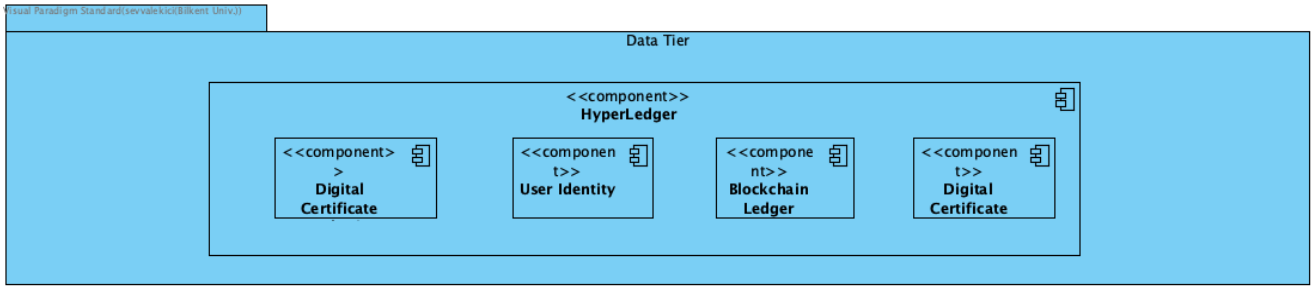

Figure 6: Data Tier of Nebula

3.3. Hardware and Software Mapping

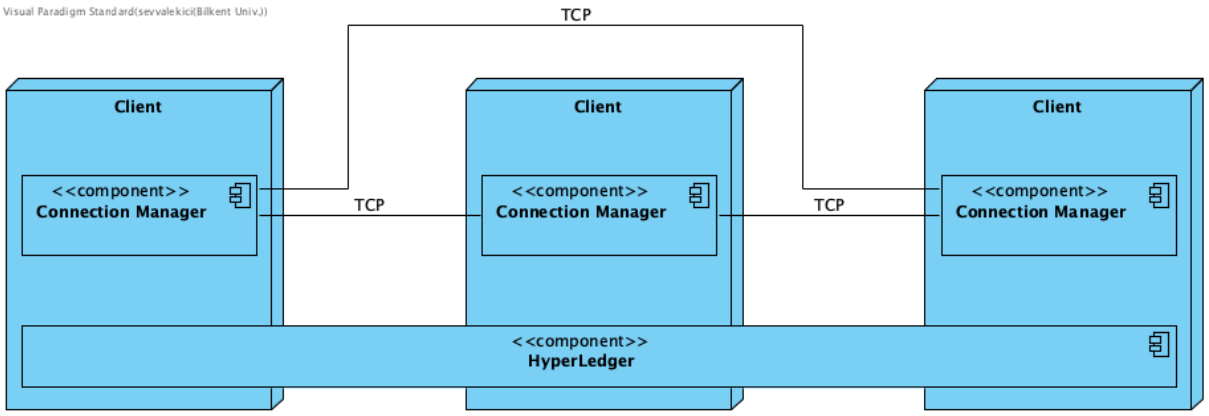

Figure: Deployment Diagram.

Hardware and software mapping of Nebula will include three clients and another component that is for the decentralized peers of the system. There is a three client architecture here in order to have a better understanding for the network of the system. The middle and the right client machines can be understood as distributed networks whereas the left one is the user client machine. Client machine is taking a role that user is interacting with the network via tasks by uploading, downloading or joining network in a navigated GUI schema. Decentralized network is responsible for the file distribution. As it is seen in the diagram, HyperLedger component is not centralized in one machine and so our system achieves decentralization with this model. Since Nebula is P2P application, three client machines that will hold the file chunks are also included in the decentralized network which means its roles are also inherited.

3.4. Persistent Data Management

Hyperledger Fabric is the key component for our data management system since Nebula is a P2P cloud network using blockchain database. User registry information such as, username, password and email will be kept in the Fabric Certificate Authority (CA). With this registry information, users will be assigned a public and private key pairs for the authorization of their identity. Moreover, relation between user and a file (owner of a file or holder of a file chunk (segment)) and the

registered peers with their address in the system will be kept in a distributed ledger on every node. Keeping the data distributed is the vital part of the Nebula system since consistency of the system is required. Hyperledger fabric ensures this distribution of the data via its consensus mechanism. (See the glossary for detailed explanation)

3.5. Access Control and Security

Each user will be able to edit his/her credentials and have access to their personal data. Anonymity(inherited from blockchain infrastructure) will be served for identity of the file owner in the system. Since Hyperledger Fabric implements a Elliptic Curve Digital Signature Algorithm(ECDSA) with a signature SHA-256, block information of an distributed ledger is almost impossible. Also ECDSA provides avalanche effect that means whenever a single bit changes in a block in the blockchain, then its corresponding hash is unpredictably changed. These features ensures the personal data can only be accessed by owner.

Nebula requires that a user registers with a valid email address and a password.In order to secure and ensure blockchain consensus, an email can only be used for a single user. This is intended to limit the number of accounts a person can create. In the case that the user forgets his/her password, an email will be sent to their respective emails. Without signing in, users will not be allowed to the system.

For a non-peer users, who can store their files in the Nebula but does not want to allocate a space for the network, payment information will not be stored or cached in any database.

3.6 Global Software Control

Each peer node will be event-driven in this case it can be considered as client requests in order to avoid race conditions since two users have the potential of requesting data simultaneously.

Internal Control:

- Asynchronous callbacks will be used for communication.

- Services will notify other subsystems via their methods about their status and needs.

External Control Flow (between subsystems):

- Control flow is distributed in the blockchain and p2p framework system. Thus there is no central control instance.
- Each peer node has its own control flow.
- Peer requests needs, waits the system for the gathering information than continue its control flow.
- Peers use asynchronous callbacks for the communication

Concurrency Control:

- Since multiple subsystems and multiple peers are working simultaneously, concurrency is the significant issue the handle. Thus, services should be controlled in a way that they eliminates the race conditions which may bring consequence inconsistencies within the system**.**

3.7 Boundary Conditions

3.7.1 Initialization of the System

In order to use Nebula, user needs to have access to internet connection and a browser to go to the website. The user needs to create an account if its his/her first time using the Nebula Cloud System. User needs to give his/her email and a password for signup information. Also, user needs to be state if he/she will contribute to the system by giving up some storage space or not.

3.7.2 Termination of the System

User can logout from Nebula by simply clicking logout button provided. In such case the user wants to use Nebula Cloud again he/she will login again. If user does not logout from the account they will be automatically considered logged in even if they close the website since for the browser client there will be a session manager which manages each sessions of the user with determined expiration time.

The most significant termination condition is termination requires a signal to send to the system such that the peer is not available for the connection. There also stands several implemented termination cases like closing the socket connection,

which are considered trivial with respect to the termination signal and not mentioned in details here.

3.7.3 Failure of the System

If there is no internet connection, user will fail to download any file from the system or upload any file to the Nebula Cloud. In this case application waits for a network connection to apply any change. Only cached data will be available during that time.

Due to inherited feature of the blockchain technology, if the system is not a robust network with a wide distributed nodes, it is difficult to get the full benefit from the technology. This failure might be occur in the first release of the application.

Another failure may occur due to read and write permissions of the located operating system. Since Nebula uses the located drive for system allocation, operating systems may not allow this allocation due to their security policies. This failure will resolved by user action to give permission to the application.

In addition, peer connection can be interrupted due to the network infrastructure of the client machines. In this failure, system will act to connect backup nodes in order to recovery.

Validation of the distributed ledger can create a failure in the system. For any bad-intentious change in the distributed ledger, hyperledger ensures the most distributed chain is the original system chain. Thus, hyperledger will solve the issue in itself.

4. Subsystem Services

4.1. Presentation Tier

Presentation Tier is only responsible for the UI and the interaction between the application and the user.

4.1.1. View Subsystem

View Subsystem takes care of visual interaction between user and the system. Individual explanation of components of the module are listed below:

View Manager: It controls the task distribution between other components of the view subsystem.

Home Page View Manager: The View Manager initially directing user to the home page. In this page download option is featured alongside with upload option. User will decide which one to do and Home Page View Manager will redirect the user to one of those.

File View Manager: User can view the files he/she uploaded to Nebula Cloud and download a prefered file if desired.

Settings View Manager: User can view the settings of his/her account and can change these options if desired.

4.1.2. Controller

Controller Manager: It controls and combines the task distribution between other controllers in this subsystem.

Authentication Manager: This manager controls the login and registry of the user interface. It is a control label between the client UI and the data tier which makes the data validation and sanitization.

File Explorer Manager: File Explorer Manager handles the actions of the user towards their files. These actions are downloading, uploading, deleting or viewing the files.

Account Manager: This manager is used to handle users actions for accounts such as signing up a user, checking sign up informations, logging in, changing password, deleting account and changing the option of being a peer.

Search Manager: It’s a sub component of the file explorer manager. It searches the file in the system directory that client application uses.

4.2. Logic Tier

Logic tier subsystem consists of mission critical operations. In this subsystem, application connection operations, file separation and distribution operations, and storage allocation in the located machine is handled. With respect to the operation types, it can also query to the blockchain database and retrieve the necessary information from data tier.

4.2.1. Connection

Download Manager: Download Manager handles the download operation from the Nebula system to the users own storage.

Upload Manager: Upload Manager handles uploading a file to the storage system of Nebula from the local storage unit of user.

Network Manager: This is the central class of Connection Package. Network Manager is responsible for the functions of taking input elements and identifies the event that occurs.

Peer Manager: Peer manager is the connection control mechanism for the client application. It finds the peers from blockchain database and establishes a connection with those peers.

4.2.2. File

File Manager: This is the central class for the File Package. It is responsible for the communication between other classes in the package.

File System: This is the corresponding file manager agent for Nebula application. Nebula will traverse the file directory in the located operating system.

File Chunk: This is a Data-Transfer-Object(DTO) for the file chunks. The related operations regarding to file chunks will be handled here.

File: This is a Data-Transfer-Object(DTO) for the whole file. The related operations regarding to file will be handled here.

4.2.3. Storage Allocation

Storage Allocation Manager: This manager handles the allocated space by the user to the system. It calculates the usage of the bandwidth and the storage and decides how much storage space should be provided.

Revenue Manager: Revenue manager calculates the cost of the user to the system. Also it calculates the revenue space for a user who allocated space by the user.

4.3. Data Tier

4.3.1. Hyperledger

These are the inherited components of the Hyperledger Fabric. These components will be used for our blockchain infrastructure. We will configure these components in order to fit Nebula network.

Blockchain Ledger: Blockchain ledger is responsible for the file and user audits in a distributed network.

User Identity: User identity corresponds the different actors in the blockchain network including peers, applications, administrators etc. These identities determines the permissions over the resources and the information access**.**

Digital Certificate: This is a digital certificate which hold a set of attributes relating to the holder of the certificate. [19]

Digital Certificate Authority: The certificate authority is a validation components which permits users to take undesired roles or do a job which is not allowed by a certain type of user identity.

5. Glossary

Hyperledger: Hyperledger is mainly an open source blockchains and tools collection. Main collection is backed by Linux Foundation but also various companies (including IBM) have their product in hyperledger. The reason hyperledger is used in the project Nebula is instead of putting too much effort on building blockchain infrastructure again, hyperledger collection can be re-usable for the boost innovation

speed. Since the terms related to blockchain such as smart contracts and consensus mechanisms etc. are difficult to implement, we will use hyperledger components.

For more architectural information please see the reference 11 and 12.

Revenue: Nebula offers a revenue in return for the reserved storage on the devices. It will be a storage privilege for the user, which will be defined as a constant factor of reserved storage**.** Since Nebula will have non-peer customers, who is using the Nebula network for cloud storage but not allocated a storage for the network, it encourages customers to contribute to the network by returning storage revenue. The system will use the storage, cpu and bandwidth of the peers for the sake of Nebula network. We will reward the peers in the return of their system usage i.e free storage space. This given free space and its calculation mechanism is called Revenue.

Proof of storage / Proof of replication: Proof of storage is a scheme used by Filecoin, to prove to the client that the data is still stored. This can be done multiple times. In addition to that, Filecoin implements proof of retrievability, which proves to the client that the data can be retrieved. This is slightly different from the proof of storage, as it could have been stored correctly, but retrieving the data is not possible, as some chunks cannot be accessed. [13]

Proof of space-time: Proof of space-time is the scheme that combines time and space. It proves to the client that the data is stored at the time of the challenge. As this proof gives the time-lapse of storage it proves to the client that the data has been stored during all this period. [13]

Trustless System: Blockchains don’t actually eliminate trust. What they do is minimize the amount of trust required from any single actor in the system. They do this by distributing trust among different actors in the system via an economic game that incentivizes actors to cooperate with the rules defined by the protocol. [15]

Consensus: The process of keeping the ledger transactions synchronized across the network — to ensure that ledgers update only when transactions are approved by the appropriate participants, and that when ledgers do update, they update with the same transactions in the same order. [18]

7. Architectural Changes From Previous Reports

After the feedbacks from our supervisor Dr.Eray Tüzün, and architectural consultants Asst. Prof. Pelin Angın (From METU) and Ali Işık (Software Architect of Menapay which is a blockchain startup), we have decided some architectural changes in our system. As it is stated in analysis report, we moved our blockchain infrastructure to the hyperledger fabric which provides all requirements to our application we want. Hyperledger fabric has a strong community and it is backed by Linux Foundation and IBM company which let us choose to depend our blockchain infrastructure.Moreover, android mobile clients are postponed as a feature plan due to their lack of battery capacity and storage accessibility issues. On the client side, we are planning to release a desktop client for now. Last but not least, we are planning to use an open source p2p framework. We are analysing libp2p-go framework, after analysis we will combine the hyperledger and the framework.

8. Potential Risks at the Development Cycle

In this section, we will state the potential risks which we can encounter while we are in the development process. We consider this chapter would be helpful in order to address stated risks.

- **Debugging and Testing**: Since Nebula consists of a peer-to-peer & blockchain based network, most probably we will have challenges while debugging and testing this complex network. As opposed the convenient blockchain infrastructure that Hyperledger Fabric provides, it is a complex system uses Docker Containers and different type of consensus mechanism. Customization and configuration of the Fabric might be challenging as well.
- **Compatibility of Hyperledger and P2P Framework:** As it is stated in section 7, we are planning to combine one open source P2P framework with hyperledger fabric in an efficient way. However, it will create an enormous risk since we are novice on these two technologies. Moreover, it might be hard to figure the problem out if one of the system fails. Proficiency of using these systems are significantly required.

9. References

[1]H. G. Do and W. K. Ng, "Blockchain-Based System for Secure Data Storage with Private Keyword Search," 2017 IEEE World Congress on Services (SERVICES), Honolulu, HI, 2017, pp. 90-93.

[2] https://www.investinblockchain.com/what-is-siacoin/

[3] https://www.forbes.com/sites/forbestechcouncil/2017/12/05/todays-centralized-cloud-and-the-emerging-decentralized-edge/#4de8d2376b3c

[4] https://en.wikipedia.org/wiki/Dropbox_(service) Accessed: 31.10.2018

[5] What is Decentralized Storage? (IPFS, FileCoin, Sia, Storj & Swarm) https://medium.com/bitfwd/what-is-decentralised-storage-ipfs-filecoin-sia-storj-swarm-5509e476995f, Accessed: 31.10.2018

[6] Decentralized storage wars. Storj v Sia v Filecoin v Maidsafe https://decentralize.today/decentralized-storage-wars-storj-v-sia-v-filecoin-v-maidsafe-27dc3d37434f, Accessed: 31.10.2018

[7] Battle of decentralized storages: SiaCoin (SC) vs Storj (STORJ) vs Filecoin (FIL) https://captainaltcoin.com/filecoin-vs-siacoin-vs-storj/, Accessed: 31.10.2018

[8] Decentralized Cloud Storage — Storj. (2018). Available: https://storj.io/. [Accessed: 31- Oct- 2018].

[9] https://umu.diva-portal.org/smash/get/diva2:1111497/FULLTEXT01.pdf

[10]The Meaning of Decentralization – Medium. (2017) Available: https://medium.com/@VitalikButerin/the-meaning-of-decentralization-a0c92b76a274 [Accessed: 31- Oct- 2018].

[11] “Hyperledger Architecture, Volume 1”,2018. https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf. Accessed 10 Oct 2018.

[12] "About – Hyperledger". Hyperledger, 2018, https://www.hyperledger.org/about. Accessed 3 Nov 2018.

[13] https://courses.cs.ut.ee/MTAT.07.022/2018_spring/uploads/Main/bruno-report-s17-18.pdf

[14] 4 of the Best Decentralized Cloud Storage Solutions to use in 2018 https://windowsreport.com/decentralized-cloud-storage/

[15] “What do we mean by “blockchains are trustless”?”03.02.2018 https://medium.com/@preethikasireddy/eli5-what-do-we-mean-by-blockchains-are-trustless-aa420635d5f6

[16] https://www.softwareadvice.com/resources/it-org-structure-centralize-vs-decentralize/

[17] https://www.quora.com/How-does-centralized-and-decentralized-computing-differ

[18] “Introduction -- hyperledger-fabric”,2018. https://hyperledger-fabric.readthedocs.io/en/release-1.3/blockchain.html. [Accesssed: 19.12.2018].

[19] https://hyperledger-fabric.readthedocs.io/en/release-1.3/identity/identity.html#digital-certificates