Bachelor of Computer Science (Digital Systems Security)

Final year project – Proposed system solution

(CSCI321)

version 1.0

A location based document locking application

Table of Contents

Project team structure

| Name	Email	Role
Abhi Jay Krishnan	ajk126@uowmail.edu.au	Manager, Designer
Durrah Afshan	durrahafshan@gmail.com	Documenter
Rivaldo Erawan	rivaldo.erawan97@gmail.com	Implementer
Kim Heoncheol	effectmix@gmail.com	Tester

The above team structure is implemented to clearly distinguish the area each team member is to focus on and it does not mean he/she will be the only one involved in doing that task. This is to have a pair of eyes watching each aspect of solution development.

We have decided to use a private GitHub repository for version controlling and team collaboration. Each team member will have local repository which will then be merged with the central repository. All members will be working on different part of the project at a given point in time to prevent clashes when committing to central repository. All documents and source codes will be stored in this central repository.

Each project meeting will have a project dairy summarising the content of the meeting and what actions must be taken.

Project Gantt chart
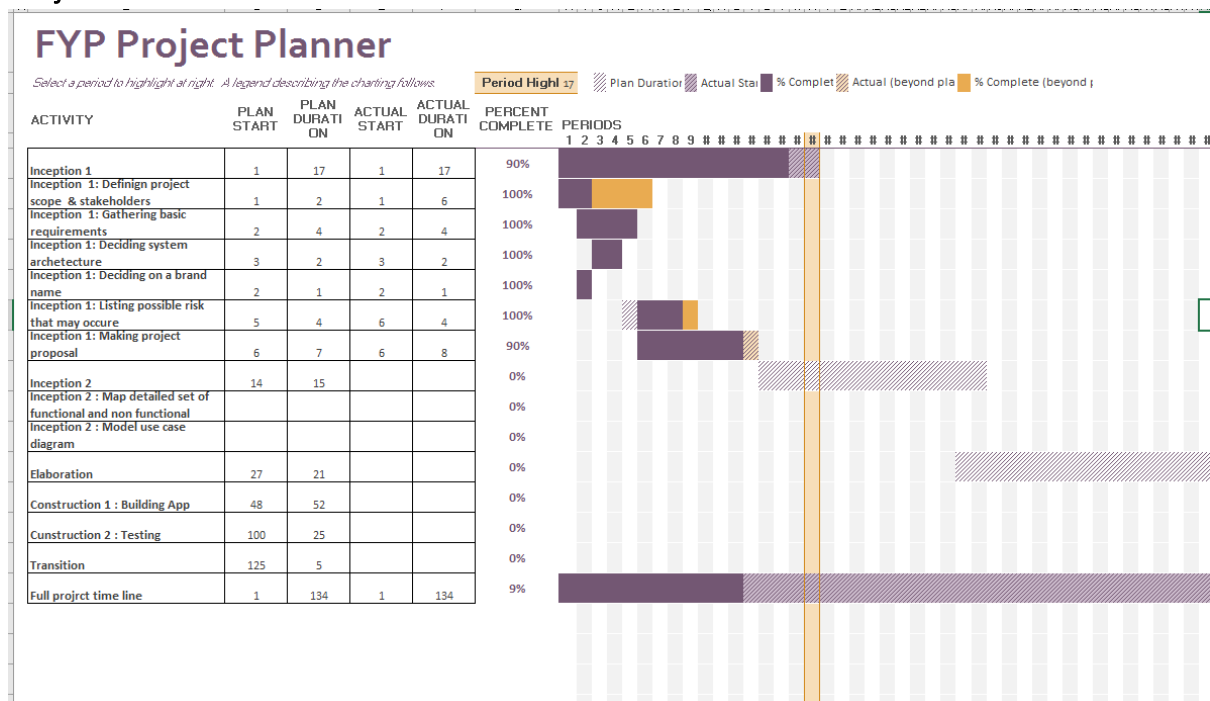
Market Survey

We have done a bit of a research on the currently available applications in the market before defining the project scope. Here are the apps, their features and somethings that they are missing out.

App	Platform	Description	Security	Features	Short-comings
Secure Safe [1]	iOS, Android, Desktop (Mac)	App for online file storage & password management. Unique service as it provides double encryption, triple data storage and zero knowledge architecture, which ensures very high level of data security and privacy protection.	➢ Automatic logout upon exiting the program ➢ Encryption is done using AES-256 and RSA 2048 ➢ Files are decryptable only by the user itself ➢ Communication between client-server is secured using https ➢ Passwords are encrypted for maximum security ➢ Support 2-factor authentication (SMS token) for premium users	➢ Login includes password and id (email) ➢ Save texts, images, documents (scans), and videos ➢ Scans (similar like pictures and convert to pdf) ➢ Email system ➢ Import/export data	➢ Connection independent (slow connection leads to slow access to data) ➢ Need internet connection, offline mode can only access passwords ➢ Allow multiple login attempts with wrong password/id ➢ No password recovery
Passible [2]	iOS	A password management app which is redesigned for fast and simple experience while entering website's logins and credit cards.	➢ Automatic logout after exit from program ➢ AES-256 encryption ➢ Does not allow multiple login attempts (timeout 5 minutes for 3 consecutive wrong attempts; after that each wrong attempt will get a 5-minute timeout) ➢ Encryption with key and random string	➢ Login using 4-digit pins ➢ Support Touch-ID (fingerprint) ➢ Save account credentials and credit card details ➢ Analyzing password strength features ➢ Support offline mode ➢ Private web browser	➢ Does not support import/export data ➢ No password recovery
File Locker [3]	Android	App to protect your content against unauthorized	➢ Fast way to lock huge files (like movies) by hashing just both beginning	➢ Involves Android Media Scanner automatically	➢ No password recovery as password is not stored anywhere.

		reading, playing, watching, etc. The application encodes the file and makes it unreadable.	and ending of file (optional). ➢ Tracks temporary unlocked files by notification, to keep in mind you left unprotected docs. ➢ Encodes the content and file names as well.	after change to make file visible by other applications. ➢ Smart looking through directories for documents, locked and unlocked files. ➢ Opens documents directly from app. ➢ Notifies about forgotten unlocked files, which were left by user.	➢ Only one-way encryption.
File Locker – Lock any file [4]	Android	App that keeps all files secure and private with file locker. File Lock lets you password-protect your personal files (ex: photos, videos, documents, etc) in android phones.	➢ Encrypts file and save in secret location in SD Card so that file is completely secure. ➢ Import files from SD Card / Phone Memory ➢ Password protected app entry with a numeric code or Pattern lock.	➢ Unlimited files can be locked. ➢ Optimized for HD tablets. ➢ Fastest lock process with multi-select feature to import hundreds of files quickly. ➢ Intuitive interface for a great experience. ➢ Hide sensitive videos and pictures. ➢ Opens documents directly from app. ➢ Password recovery option is available	➢ Encryption and decryption is not location based.
Private Photo Vault [5]	iOS, Android	A photo safe that keeps all private pictures and videos hidden	➢ Automatic logout after exit from program. ➢ Login using 4-digit pin/pattern lock. ➢ Import/export data	➢ Save photo and videos ➢ Support offline mode ➢ Private browser	➢ Allow multiple login attempts with wrong pins

		behind a password.	➢ Pass recovery (email)		

Conclusion

The table above shows a comparison between 5 software applications available in the market. This survey shows the different types of platform the application can run on. It provides information on the features that have been used. The most common feature in all these app is that they use encrypted password for securing files such as images, videos, documents, etc. Major drawback noted from the table is the lack of password recovery and the encryption/decryption of files is not location based.

Project Scope

Project purpose

This project aim to provide user a way to store confidential documents in mobile devices and access it only in the area he/she find it is safe. By including two factor protections, one being password (what the user knows) and second being the location (where the user is currently), we will be able to provide a better solution compared to the applications currently in the market (based on market survey).

These are few ways a document in a mobile device may be compromised: -

- The documents stored in mobile device may end up in the wrong hands if the device itself is stolen.
- The user may lend the device to someone who intern may wish to gain access to these documents.
- The documents may be accessed remotely by penetrating device through network.

Our solution aim to provide a secure vault for document storage so the it does not get into wrong hands even if the device is compromised. The solution also provide a secure backup cloud storage with double layer encryption one by the app itself and one by Amazon server.

Target Users

There are several applications for this solution in the market, here are few of them:-

1. Employees working in defense industry may have to handle highly secretive documents that should not be taken of the secure premises.
2. Most of the large firms these days have documents that are their intellectual property and wish to keep them from getting into wrong hands.
3. General public may want to store their personal information and keep it within their safe zone such as their home.
4. Governmental authorities may wish to keep their confidential documents within the country or within the restricted area.
5. A hospital may wish to keep the patient's document within the campus but at the same time giving staff the freedom to view it while moving around.
6. A school may want to let the authorized staff to review an exam paper on the move while keeping the document within the restricted zone.

Proposed features

The proposed solution is an android based app with following features: -

Functional Features

PDF Viewer

A PDF viewer for user to view his documents in the vault. The pdf viewer will only be accessible after the user has been authenticated and if the user is within the radius of the location stored in the database. The pdf viewer will close when the user moves out of this zone. The file that the user wishes to see will be the only one that will be decrypted. The rest will remain as cipher text even when the user is in authorised area. This pdf viewer will help the user to be more productive by having the ability to access sensitive document while moving within the secure location.

Deleting files

The user has the option to delete the files that are not needed and these files will also be deleted from the backup.

Setting preferred locational radius

Once the user adds a new file he can set the radius he wishes with small radius being more secure and larger radius being more convenient. The files will be grouped based on the location and the user can choose the area if there is an overlap.

Less clustered interface

The user will be only able to view the files that was saved to a location making file accessing, pleasant and less tedious.

Import files

The user will be able to import a new file from the local file directory and secure it through encryption. The original file can be deleted to prevent adversary from viewing it.

Secure cloud storage

The data will be safely stored in the central database with additional layer of encryption by the cloud infrastructure provider.

Non-Functional features

Changing Password

The user after logging into the app can change the password by providing the old password. The password will be saved to the central database. The password forms a part of the key used to secure files and tables in the database. Once the password is changed the database will be secured using new password when the user logout.

Password recovery

If the user forgets the password he will receive a password recovery code through his email or notification. Upon entering the code, he will be redirected to the changing password procedure.

Dealing with stolen device

The user can migrate to a new device using the backup data and change password so that the data in the old phone can never be decrypted as the sever will never authenticate the user even if he knows the old password.

Dealing with missing folder

The user will be promoted to download the backup folder and will be able to access files as usual or the user can choose to create a new one.

Protection against GPS spoofing

The location will be checked regularly (every 10 sec) and will be compared with the previous location, if there is a significant difference in the coordinates (100m) the app will encrypt the files and logout.

Secure cryptographic algorithms

Files are encrypted using secure cryptographic algorithm such as Advanced Encryption standard (AES256). The file will not have the same name as the original file after encryption, the original name will be stored in the database along with the new name. The password will be hashed using strong

hash functions such as the SHA256. The central database will have an additional layer of encryption provided by the cloud service provider (Amazon Web Service).

Migration

The user may wish to move to a new device, and the app allows user to download the backup file from the cloud and continue using as usual. The user credentials are stored in a central database to ease migration.

Backup

The user can back up his data to cloud (AWS S3) to retrieve it later when the disaster strikes or for migration. The system will track the changes and only save the newly added file. The data stored in cloud is send over after encryption.

Speed

A temporary SQLite database will be created to store the user credentials locally to costs (time) that is incurred through constant communication with the central database. This database will be dropped when the user logout.

These are the key features that will be included in the application. Further enhancements such as support for more file types will be added if these basic requirements are met.
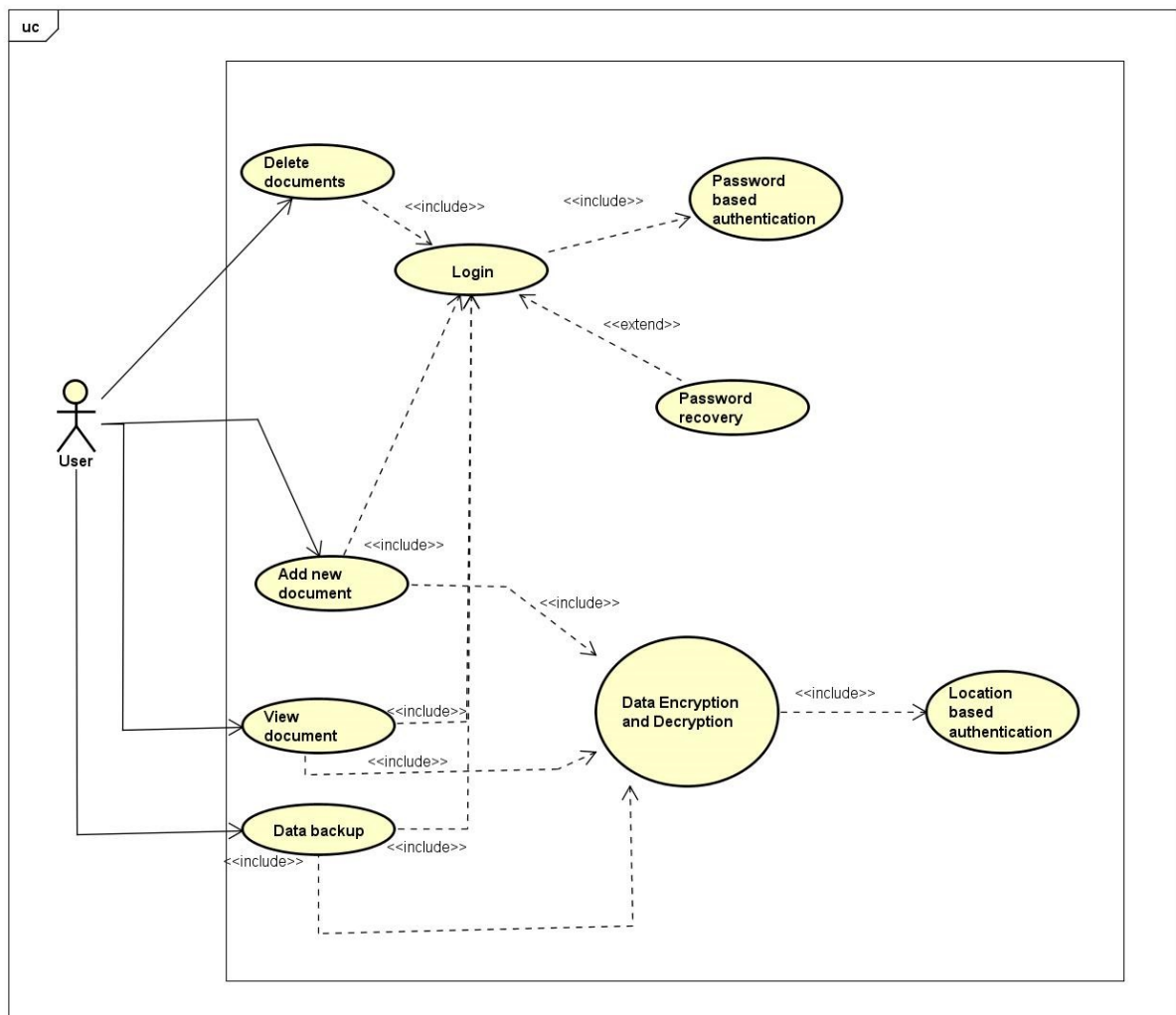
Highest level use case diagram

Figure 1: highest level use case diagram

Programming development methodologies

Waterfall

This is a legacy model for software development projects. The development cycle is linear and not capable of supporting challenges faced by modern software development domain such as changing user requirements and adaptations. There is high amount of risk and uncertainty using [6].
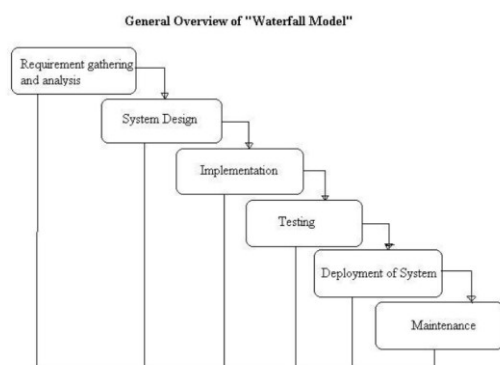

Figure 2: water fall model

Prototyping model

Under this model the requirements are gathered from user and a prototype is build based on it for evaluation before refining requirements. This iteration repeats until final design is confirmed before final construction of the product itself. This model may be inappropriate for this project as there are no rapid changes that may need iterative prototyping. Prototyping also require active involvement of end user which is not possible for this project [7].
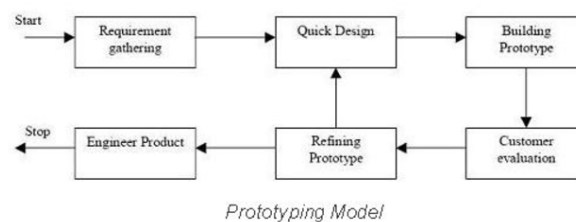
Figure 3:prototyping model

Agile model

Agile methodologies include Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature-Driven Development (FDD). They all are based on a common principle of iteration and continues feedback that it provides to successively refine and deliver a software system. This method also requires active involvement of end-users and may track off the user is not sure about the

Figure 4:agile model

outcome they want. There is also lack of emphasis on necessary designing and documentation. Hence, agile model may not suitable for this project [8].
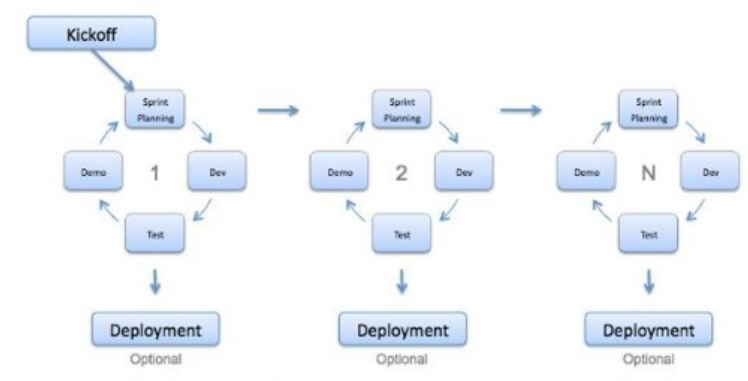
Rational Unified processing

Rational unified processing model provide a disciplined approach to assigning task and responsibilities within the development organisation. The development is divided into 4 sections:-

Inception – The project scope is defined along with requirements gathering and Risk analysis.

Elaboration – Coming up with detailed design of the system.

Construction – The actual development of the system.

Transition – Testing and deployment of the system.
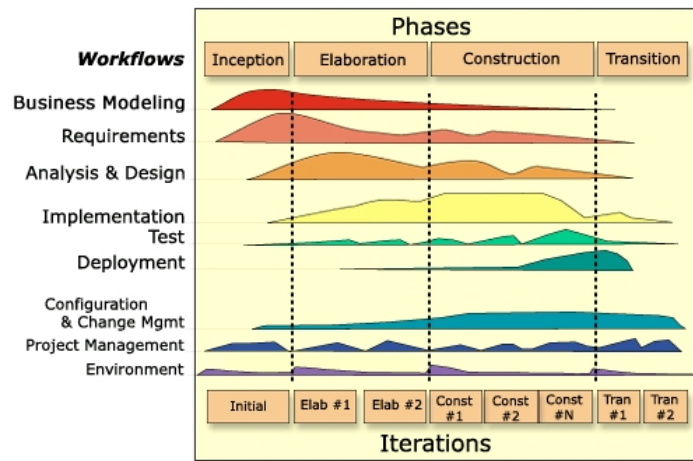
Figure 5: Rational Unified Processing

There may be more than one iterations for each of the above stages and the also provide a flexibility to make changes to requirement deep into the development cycle [9].

After considering above different models we have decided RUP is the most suitable for this project as it allow us to decide the business case, need less involvement of end user and gives flexibility to make changes during later part of the development process.

Table 1: Proposed application development language

Language	IDE	Developer	Description
Java	Android SDK, Eclipse + Android ADT	Android	Java is used to develop android native app and java is the official language for development android application using android studio.
C++	QT	The QT company	An android application can be developed in C++ using Qt libraries. It archives the same speed as natively developed app but the app package size is significantly large. Using C++ will also require more development time.
Kotlin	Android SDK	JetBrains and opensource developers	Kotlin relatively new programming language that runs on java virtual machine.
HTML JavaScript	Apache Cordava	Adobe Systems	HTML and Java script can be used to develop web apps that run in android OS and cordava.js help in connecting with systems features such as camera, accelerometer and GPS. This method normally produces apps which are slower that natively developed apps.

There are other methods used to develop an android application but none have the same flexibility and speed as the natively developed apps using java. Hence, we have considered in using java and android SDK to develop this solution. Android SDK also uses XML to define the user interface elements.

Table 2: Platform comparison

	Desktop platform	Mobile devices
Portability	A Desktop is restricted to a local area while the while the laptop is normally less mobile than a mobile phone.	Due the small form factor, mobile devices such as smart phones and tablets are more portable compared to desktop platforms.
Connectivity	A desktop environment normally has built in WIFI and Ethernet connectivity.	A mobile device comes with a WIFI connectivity but lacs Ethernet connectivity. One the positive side mobile devices comes with cellular data connectivity.
Operating System	Desktop environment may have large verity of hardware and there are compatibility issues when OS tries to interact with hardware.	Mobile OS is mostly Android, IOS and Windows Mobile normally have fixed hardware.
Cost	Desktops and laptops have larger capacity and better processing capabilities compared to mobile devises hence they are generally more expensive.	Mobile devices are generally less expensive.

From the above comparison, we can see that mobile devises have better portability and low-cost factor which makes it an ideal platform for user to reference document on the go. The user will also find mobile devices a cheaper option compared to desktop.

Table 3: Mobile OS Comparison

	Android	IOS	Windows
Downloadable Apps and Apps Market	- **Google Play** - Nowadays, Google Play have dominated in the application market due to convenience of App approval process - Google Play is well integrated with Google's Applications such as Gmail, **Google Maps**, Google Drive and so on.	- **App Store** - In the Initial Period of App Market, App Store dominated in the application market.	- **Window Store** - Most of popular Applications are already in the market. But the update speed is slower than Android and IOS. -It is available to unify the application between Desktop such as Windows 10 and Mobile Devices.
OS Platform's Features	- **Samsung Flow**: version is still Beta version - Hangout - Google Fit - Have Wide Range of available devices (Mobile phone, tablets and wearable devices such as smart watches). Not only Samsung but also google and so on.	- Devices based on IOS can be synchronized between Apple Devices. - iMessage - Apple Health - IOS is only limited into Apple own devices such as MAC. - Most of the devices based on IOS have **expensive cost** among 3 OS.	**Continuum**: plug mobile device into the Monitor with Keyboard and Mouse then get the interface such as Desktop. Thus use the phone like PC. - Skype - Microsoft Health
Virtual Assistance	- **Google Now** - Google Now opens its API to developers who can use it for operating or referencing other apps - **Picture Recognition**: It is available to offer information by input screenshot or picture	- **Siri** - Siri has **accurate understanding** compared as Google now - But the information area by Siri is limited For example, Playing music, setting timer or alarm and so on.	- **Cortana** - It is the latest virtual Assistance in Windows at 2015 - Bing Search, Music Recognition - Still need to be uploaded about features compared as the Android and Siri

Security	- Android is based on **Open Source Code** - Easy to submit the Application into Google Play with cheap submission fee compared as the App Store - it has reinforced security in Google Play store after stage fright attack in 2015 - Direct booting which allows application to begin with the lower layer in mobile devices - File Encryption which allows protecting the personal data in devices. - Because the wide range of devices is available in Android OS, compared as the other OS platforms, it is vulnerable from malicious attack and Not whole Android OS platform is updated at the same time. -	- requires only use Apple's own devices. - App Store requires signature and checking from Apple Before submit Application into App store. - using secure encrypted channel when upload/update apps - Like Android, IOS is one of the OS that a lot of users use in nowadays. Thus There are probabilities attacked from malicious third party.	- Like IOS, Window store has strict app submission process - Device Encryption based on the local contents - Easy to integrated apps between PC and mobile devices - **Microsoft passport**: strong authentication process to access to resources. - **Device Guard**: protect data from malicious programs - **Microsoft Enterprise Mobility**: focus on Security Session with ATA (Advanced Threat Analytics) - One of the Problems in Windows store is small market compared as Android and IOS. Thus, it has a probability to being attacked in the future and also lack of features in the markets.
Biometric Security	- **Fingerprint sensor technology** after IOS - lyrics pattern recognition technology from Galaxy Note 7 in Samsung	- IOS is the **FIRST** OS which released **Fingerprint sensor technology** in the devices.	- **Windows Hello**: Authentication to sign in Windows 10 devices securely. (Surface Pro 4, Surface Book, most PCs) with fingerprint readers or Face recognition always work.

After considering the above comparison between different OS platforms we have decided to go ahead Android as it has the largest market share [10] and cross platform option does not achieve the same performance level as natively build apps [11].

Table 4: Comparison between Relational and Non-Relational Database

	Relational Database	Non-Relational Database
DB Schema [12]	➢ Structured Query Language ➢ Stores items in tables to minimize duplicate values ➢ Fixed schema ➢ Example: MySQL, Oracle	➢ NoSQL database ➢ Unstructured data ➢ No fixed schema ➢ Document-oriented ➢ Example: MongoDB, HBase
Scalability [13]	➢ The tables relationship makes scaling more resource-intensive	➢ Stores each item as single document for high scalability ➢ Includes sharding or partitioning ➢ Uses elastic scalability
Flexibility [14]	➢ Provides flexible structure to meet changing requirements and increasing amounts of data ➢ This model permits changes to a database structure to be implemented easily without impacting the data or the rest of the database ➢ In reality, growth and change are limited by the relational database management system and physical computing hardware	➢ Flexible data model ➢ Easy to store and combine data of any structure ➢ Defining types of data in advance is not required ➢ Allows dynamic modification of schema without performance impact
Cost [15]	➢ Rely on expensive proprietary servers and storage systems ➢ Licenses for this system can be quite expensive	➢ Uses clusters of cheap commodity servers ➢ Databases are open source and therefore free
Consistency [12]	➢ Tight consistency	➢ Eventual consistency rather than ACID property
Transaction [16]	➢ Transaction with ACID property (Atomicity, Consistency, Isolation & Durability)	➢ Does not support transactions
Performance [14]	➢ High performance speed	➢ High read and write performance ➢ Unlimited growth with higher throughput ➢ Lower latency than relational database ➢ Faster development life cycles for developers
Reliability [17]	➢ Changes committed in a transaction are stored and available in the database even if	➢ Automatic back up of data in separate facilities (E.g. In DynamoDB)

	there is power failure or the database goes offline suddenly.	
Reporting tools [17]	➢ Wide array of reporting tools helps to prove application's validity	➢ Lack of reporting tools for analysis and performance testing
Security [18]	➢ By splitting data into tables, certain tables can be made confidential ➢ The system can then limit access only to those tables whose records they are authorized to view	➢ Has weak password storage ➢ Lack of encryption support for the data files ➢ Weak authentication both between client and the servers ➢ Vulnerability to SQL injection ➢ Denial of service attacks. ➢

After considering the above factors we have decided to use non-rational database (NoSQL) as a central database as it is scalable and prevent a lot of administrative complications. DynamoDB by Amazon Web Service is such database and well established. AWS also provide a library to interact with the database through the Android application.

Table 5: Table Relational Database Comparison

	Oracle SQL	MsSQL	SQLite
Developers	Oracle	Microsoft	Dwayne Richard Hipp
License	Commercial	Commercial	Open-Source
Cost	Total (per Processor): Around $5,999 - $21,350	Free (Limited entry level database) - $14,256 ++	Free
Supported Programming Languages in Database	C++, Visual Basic, Python, R, PHP, JavaScript(Node.js), Ruby, Go, Java	C/C++/C#/Objective C, Visual Basic, Python, R, PHP, javaScript (Node.js), Ruby, Java, Perl, Scala, Clojure, Cobol, Delphi, Erlang, Eiffel, Fortran, Haskell, Tcl, OCaml, Lisp, Groovy	C/C++/C#/Objective C, Visual Basic, Python, R, PHP, javaScript (Node.js), Actionscript, Ruby, Java, Perl, Scala, Clojure, Cobol, Delphi, Erlang, Eiffel, Fortran, Haskell, Tcl, OCaml, Lisp, forth, D, Ada, Basic, MatLab, Lua, PL/SQL, Smalltalk, Scheme
Server-sides and stored procedures	exchanges SQL and .NET languages	PL/SQL	Nothing. SQLite is not for server sides database software. Usually used to store data into internal devices
Supporting XML	Support XML format or data structures	Like MsSql, Support XML format or structures	No XML support
Common Features	Support the features of Foreign keys, Durability, Concurrency, SQL standard, Data Scheme (In the case of SQLite, it also supports dynamic data scheme) and so on.		

SQLite being a server less database makes it ideal for developing a standalone application. We have chosen SQLite to be our temporary database in the local device.

Table 6: Comparison between cloud service platforms

	Amazon Web Services [19]	**Microsoft Azure [20]**	**Google Cloud Platform [21]**
Years in operation	11 years	6 years	5 years
Managed By	Amazon	Microsoft	Google
Free Tier features	• AWS Lambda – 1 Million Request/Month • Amazon S3 – 5GB of data storage • Amazon RDS - 750 Hours per month of db.t2.micro database usage (applicable DB engines) 20 GB of General Purpose (SSD) database storage 20 GB of storage for database backups and DB Snapshots. • DynamoDB (NoSQL) storage : 25 GB AND more…	• 14 virtual machines for 1 month • 40 SQL database for a month • 8TB of storage for a month • $200 credit • Up to 10 web and mobile apps.	• 28 instance hours per day. • 5 GB Cloud Storage. • Shared mem cache. • 1000 search operations per day, 10 MB search indexing. • 100 emails per day.
DDOS protection	Yes	No	No
Mobile App Software Development Toolkit	Yes	Yes	No
Mobile App Testing Service	Yes	Yes	Yes
Mobile App Development Services	Yes	Yes	Yes
NoSQL database Systems	Yes	Yes	Yes

We choose to go ahead with Amazon Web Services as it provides a lot of features in free tier and its highly scalable. Amazon is also the is provide a mobile API along with backup for database. Security wise Amazon shield provide protection against DDOS attacks. Aws also has very large customer profile and many large successful firms trust their services [22].

After considering all the above comparisons in the next section we are going to talk about the proposed system architecture.

System Architecture

The architecture adopted by the application will be a new and sophisticated *cloud-client architecture*. This architecture works by making use of cloud computing resources to manage user's data and credentials as well as authenticating user. The proposed application will be using AWS (Amazon Web Services) as its cloud service provider and AWS Mobile SDK [23] for the development of the app. The main advantage this architecture has is that we don't need to manage the resources in the cloud including the security of it.

Overview of the architecture:
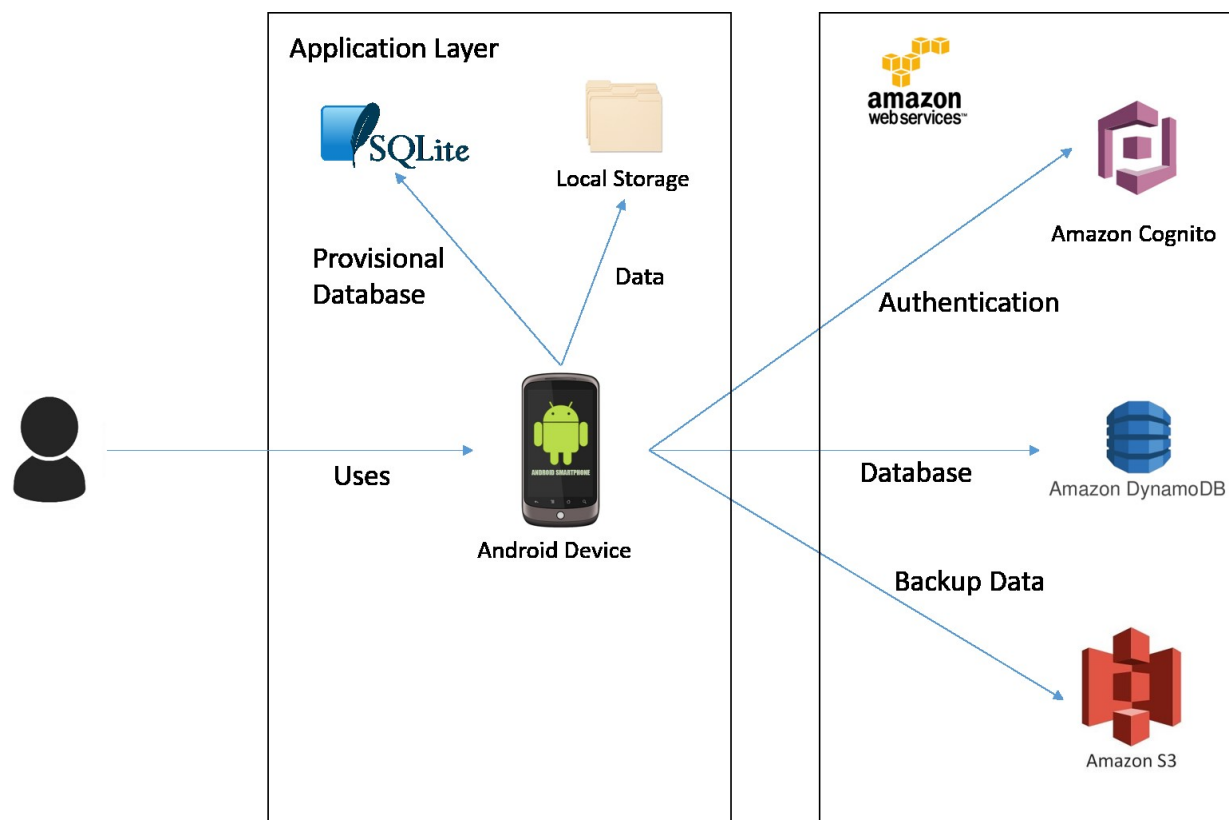
Components of the architecture:

Amazon Cognito

Amazon cognito makes it easy to add user sign-up and sign-in and manage permissions for mobile apps. We can create our own user directory within Amazon Cognito, or we can authenticate users through social identity providers such as Facebook, Twitter, or Amazon; with SAML identity solutions; or by using our own identity system. With Amazon Cognito, we can focus on creating great app experiences instead of worrying about building, securing, and scaling a solution to handle user management and authentication [24].

Amazon DynamoDB

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed cloud database and supports both document and key-value store models. Its flexible data model, reliable performance, and automatic scaling of throughput capacity, makes it a great fit for mobile, web, gaming, ad tech, IoT, and many other applications [25].

Amazon S3

Amazon Simple Storage Service (Amazon S3) makes it simple and practical to collect, store, and analyze data - regardless of format – all at massive scale. S3 is object storage built to store and retrieve any amount of data from anywhere – web sites and mobile apps, corporate applications, and data from IoT sensors or devices. It is designed to deliver 99.999999999% durability, and has many customers each storing billion of object and exabytes of data. You can use it for media storage and distribution, as the "data lake" for big data analytics, as a backup target, and as the storage tier for server-less computing applications. It is ideal for capturing data like mobile device photos and videos, mobile and other device backups, machine backups, machine-generated log files, IoT sensor streams, and high-resolution images, and making it available for machine learning to other AWS services and third-party applications for analysis, trending, visualization, and other processing [26].

SQLite

SQLite is an embedded SQL database engine that can act as a database in android local system. Unlike most other SQL databases, SQLite does not have a separate server process. SQLite reads and writes directly to ordinary disk files. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format.

Other architecture that have been considered are standalone, client-server, and web-based. Standalone is simpler and more robust, it mostly doesn't need any internet connection to use all the features in the app, but the number of feature it can have is limited. Client-server opens more possibilities for the application to have more interesting features, however there would be a problem of securing the communication, the cost to set up the server, and the maintenance of the server itself. All of this can be applied to web-based architecture too since it uses a server. Additionally, web-based is less desirable due to its performance issue because there is no local storage for the application, therefore the performance is very dependent on network connection to access the data.

21

Design principles of the application:

Modularity: separating the functionality of a program into independent, interchangeable modules, such that each contains everything necessary to execute only one aspect of the desired functionality.

High Cohesion: each module has functions and elements that are strongly related, only to fulfil one particular purpose or task

Low Coupling: modules are loosely coupled and independent so that a change in one module do not affect the other modules

Standardization: implementation will conform to standard that has been established and agreed by different parties, this is crucial for things like security

The proposed application will have the following quality:

Extensibility

The proposed application can add additional functionality without changing or damaging much of the current system. New data types can be added if it is supported by the android.

Maintainability

Following the design principles of high cohesion and low coupling, small modifications will not be a problem. Changing one module will not affect other modules significantly.

Performance

The response time will be in acceptable manner even with the huge amount of data that are processed. Efficient encryption algorithm is used as well as other processing algorithm

Usability

Adapting KISS (Keep It Simple Stupid) principle in designing interfaces will give user easier times in learning and figuring out the proposed application. It lets user to take less time to perform a certain task.

Compatibility

The proposed application will be able to run in various type of android devices as well as different version of android.

Security

Data are kept safe by encryption and login is required to have access. Security measures like protection against SQL injection or encryption algorithm will follow standard.

Risk List

Risks	Actions
Sudden growth in requirement as the project progress.	Readjust project time line to accommodate the changes.
Team members unable to contribute to the project due to health or other valid reasons.	Distribute workload among remaining members. Making use of GitHub repository will help the team member who is not able to come for meeting to contribute remotely. Also using other means such as skype calls and TeamViewer.
Estimation and scheduling of development time is done on initial stage and there may be glitches along the way that will set back the project timeline.	Project plan can be revisited and adjusted to fit within the given deadline.
Usage of new and ever-changing products in the market will lead to bugs in software that is being developed.	Proper research and training is critical when using a new tools, techniques, protocol or systems.
System performance may be compromised when having substantial number of features.	Usage of good programming practices and threading.
Selecting an unsuitable design architecture	Doing a good research and developing a good picture of the end goal based on past experiences (market survey).
Loosing support for API (Application programming interface) used while making the solution.	Usage of good object oriented or modular programming architecture will help in transiting from one API to another with minimal work.
Amazon service outage	There may be a need to migrate to new service provider who provide similar services like Azure or Google cloud platform.

References

[1]   dSwizz, "SecureSafe," dSwizz, [Online]. Available: https://www.securesafe.com/fr/. [Accessed 2017 July 23].

[2]   Passible, "Home page," Passible, 10 Oct 2016. [Online]. Available: http://www.passible.com/#security. [Accessed 23 July 2017].

[3]   M. Wojas, "File Locker," Marcin Wojas, 6 May 2014. [Online]. Available: https://play.google.com/store/apps/details?id=com.mwgo.filelocker. [Accessed 23 July 2017].

[4]   Innorriors, "File locker - Lock any File," Innorriors, 15 June 2017. [Online]. Available: https://play.google.com/store/apps/details?id=inno.filelocker. [Accessed 23 July 2017].

[5]   Legendary Software Labs LLC, "Private Photo Vault," Legendary Software Labs LLC, 11 July 2017. [Online]. Available: https://play.google.com/store/apps/details?id=com.enchantedcloud.photovault. [Accessed 23 July 2017].

[6]   "What is Waterfall model- advantages, disadvantages and when to use it?," ISTQB EXAM CERTIFICATION, [Online]. Available: http://istqbexamcertification.com/what-is-waterfall-model-advantages-disadvantages-and-when-to-use-it/.

[7]   ISTQB EXAM CERTIFICATION, "What is Prototype model- advantages, disadvantages and when to use it?," ISTQB, [Online]. Available: http://istqbexamcertification.com/what-is-prototype-model-advantages-disadvantages-and-when-to-use-it/.

[8]   I. E. CERTIFICATION, "What is Agile model – advantages, disadvantages and when to use it?," ISQTB, [Online]. Available: http://istqbexamcertification.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/.

[9]   P. Christensson, "RUP," TechTerms, 2006. [Online]. Available: https://techterms.com/definition/rup. [Accessed 25 July 2017].

[10]   J. Vincent, "99.6 percent of new smartphones run Android or iOS," 16 Feb 2016. [Online]. Available: https://www.theverge.com/2017/2/16/14634656/android-ios-market-share-blackberry-2016.

[11]   P. Khatri, "Native vs Cross-Platform App Development: Pros and Cons of PhoneGap, Titanium, and Xamarin," Mobile Zone, 08 March 2017. [Online]. Available: https://dzone.com/articles/native-vs-cross-platform-app-development-pros-and.

[12]   J. Serra, "Relational databases vs Non-relational databases," James Serra's Blog, 27 August 2015. [Online]. Available: http://www.jamesserra.com/archive/2015/08/relational-databases-vs-non-relational-databases/. [Accessed 30 July 2017].

[13]   R. babu, "NoSql vs Relational database," stackoverflow, [Online]. Available:
        https://stackoverflow.com/questions/4160732/nosql-vs-relational-database. [Accessed 30 July
        2017].

[14]   D. L. Soltesz, "What are the Advantages of a Relational Database Model?," Techwalla, [Online].
        Available: https://www.techwalla.com/articles/what-are-the-advantages-of-a-relational-
        database-model. [Accessed 30 July 2017].

[15]   A. Gajani, "The key differences between SQL and NoSQL DBs.," monitis, [Online]. Available:
        http://www.monitis.com/blog/cc-in-review-the-key-differences-between-sql-and-nosql-dbs/.
        [Accessed 30 July 2017].

[16]   "Relational Vs Non Relational Database," mongoDB, [Online]. Available:
        https://www.mongodb.com/scale/relational-vs-non-relational-database. [Accessed 30 July
        2017].

[17]   M. Ramachandran, "Relational Vs Non-Relational databases – Part 1," BIG DATA MADE SIMPLE,
        23 April 2014. [Online]. Available: http://bigdata-madesimple.com/relational-vs-non-relational-
        databases-part-1/. [Accessed 30 July 2017].

[18]   M. Obijaju, "NoSQL NoSecurity – Security issues with NoSQL Database," Perficient, 22 June
        2015. [Online]. Available: http://blogs.perficient.com/dataanalytics/2015/06/22/nosql-
        nosecuity-security-issues-with-nosql-database/. [Accessed 30 July 2017].

[19]   "AWS Free Tier," Amazone Web servises, [Online]. Available:
        https://aws.amazon.com/?nc2=h_lg.

[20]   "Azure free account," Microsoft Azure, [Online]. Available: https://azure.microsoft.com/en-
        us/free/?v=17.15.

[21]   "Google Cloud Platform Free Tier," Google Cloud Platform , [Online]. Available:
        https://cloud.google.com/free/.

[22]   "All Customer Success Stories," Amazon Web Sevices, [Online]. Available:
        https://aws.amazon.com/solutions/case-studies/all/.

[23]   Amazon, "AWS Mobile SDK," Amazon, [Online]. Available:
        https://aws.amazon.com/mobile/sdk/.

[24]   Amazon, "What is Amazon Cognito?," Amazon, [Online]. Available:
        http://docs.aws.amazon.com/cognito/latest/developerguide/what-is-amazon-cognito.html.

[25]   Amazon, "Amazon DynamoDB," Amazon, [Online]. Available:
        https://aws.amazon.com/dynamodb/.

[26]   Amazon, "Amazon S3," Amazon, [Online]. Available:
        https://aws.amazon.com/s3/?sc_channel=PS&sc_campaign=acquisition_SG&sc_publisher=goog
        le&sc_medium=s3_b&sc_content=s3_e&sc_detail=amazon%20s3&sc_category=s3&sc_segment

=82556503449&sc_matchtype=e&sc_country=SG&s_kwcid=AL!4422!3!82556503449!e!!g!!ama
zon%20s3.

[27]   C. Hope, "Computer vs. Smartphone," 27 6 2017. [Online]. Available:
https://www.computerhope.com/issues/ch001398.htm. [Accessed 27 7 2017].

[28]   M. Parker, "Android vs iOS vs Windows 10 Mobile: Which mobile operating system is best?," 16
2 2016. [Online]. Available: http://www.trustedreviews.com/opinion/which-mobile-operating-
system-is-best-2928049. [Accessed 27 7 2017].

[29]   DB-Engines, "System Properties Comparison Microsoft SQL Server vs. Oracle vs. SQLite,"
[Online]. Available: https://db-
engines.com/en/system/Microsoft+SQL+Server%3BOracle%3BSQLite. [Accessed 27 7 2017].