**Galgotias College of Engineering & Technology**

**[Knowledge Park-II, Greater Noida, Uttar Pradesh, India]**



Session 2020-21

MINI-PROJECT

ON

**Share Me**

BACHELOR OF ENGINEERING

(DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING)

3RD SEMESTER

**Submitted By**: **Submitted To:**

Anuj Garg (1900970100024) Dr. Inderpreet Kaur

Anurag Chuhan (1900970100025)

**ACKNOWLEDGEMENT**

We take this opportunity to express our sincere thanks and deep gratitude to all those people who extended their wholehearted co-operation and have helped us in completing this project (“**Share Me**”) successfully.

First of all, We would love to thank to our ***H.O.D. Dr. Vishnu Sharma*** and our ***Mentor Dr. Inderpreet Kaur*** for providing us with an opportunity to undertake this project in this college and providing us with all facilities.

Lastly, We are thankful to all those friends, who have been instrumental in creating proper, healthy and conductive environment and including new and fresh innovative ideas for us during this project. Without their help, it would have been extremely difficult for us to prepare the project in a time bound framework.

**Anuj Garg and Anurag Chauhan**

**Table Of** **Contents**

**Title**  **Page No.**

ABSTRACT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_4

1. Introduction\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_5

2. Working\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_6

2.1. Development Approach\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_8

2.2. Development Methology\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_10

3. System Requirement\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 15

4. Results\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_16

5. Conclusion and Future Scope\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 19

6. References\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_21

**ABSTRACT**

Over the last few years, there has been a drastic change in information technology. This includes the various ways in which files can be shared and stored. HTTP is a relatively new protocol which has been steadily taking over more and more market share. Easy to use and easy to develop, it has picked up a following of developers who want to create content for the masses. From businesses, to non-profit organisations, to single users, there seems to be various applications which can use http to offer better, faster, and smarter sharing. This paper aims to combine the two, building a desktop based application for windows, offering users the power of easy and faster way of file sharing in the palm of their hand.

**1. Introduction**

Today the time has come that the Internet has become an integral part of daily life. Digitalised life is now dependent on digital files. Starting from documents to image files, Spread sheets to Contact files, files are now costlier than many thing and become very important.

File sharing is the practice of distributing or providing access to digitally stored information, such as computer programs, multimedia (audio, images and video), documents, or electronic books. It may be implemented through a variety of ways. Common methods of storage, transmission and dispersion include manual sharing utilizing removable media, centralized servers on computer networks, World Wide Web- based hyperlinked documents, and the use of distributed https networking.

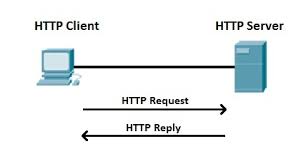
File storing and sharing is essential, and we are now having many ways to perform this but every way has its merits and demerits. For storing the use of dedicated server is used which has various demerits and also the cloud server is used which is complicated for deployment and also require more systems.

The file sharing has become very popular now a day for its higher data secure and utilisation properties. The http file sharing systems provide infrastructure for communities to share storage space. Unlike traditional distributed systems, http networks aim to aggregate large numbers of computers that join and leave the network frequently and might not have permanent network (IP) addresses. In a pure distributed http system, server-client communicate directly with each other and share resources without dedicated servers. In essence, these http systems build, at the application level, a virtual overlay network with it own routing mechanisms. Different file sharing ways as Google Docs, Dropbox, mediafire, hotfile are there, which provides various file sharing services in different ways of services

**2. Working**

File sharing applications are made up of applications called server-client. These application generally has a specific task to perform, and an application can be made of several classes, each performing part of the job of the application.

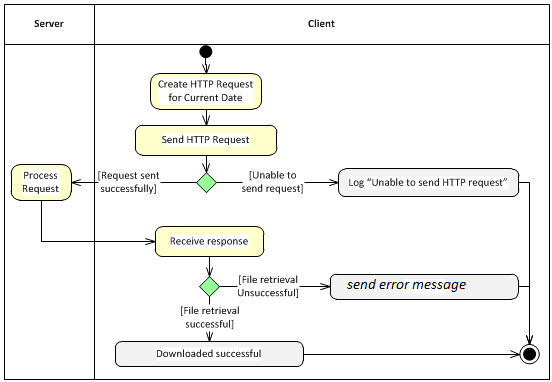
For communication between server-client, there are some protocols, Hyper Text Transfer Protocol (HTTP) is one of them which is used in this project. This protocol carry information around the system, pass along application data, or request the server to open a file.

****

***Figure 1. Flow of Data***

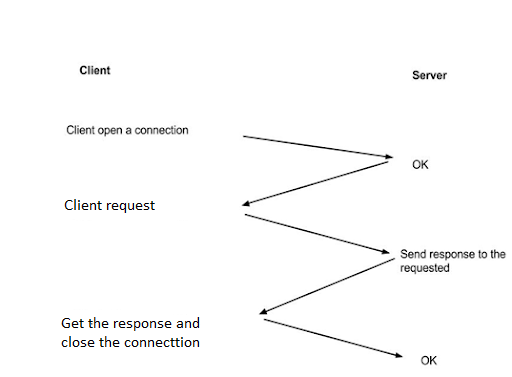
The application design diagram which shows possible paths through the application, and the activities used to act on the requested functions is shown above. As shown, should a user request as a server to send a file, the application send a request message to the client application to allow the contents to be receive using a relevant program.

Actions are broadly separated into two main functions: user authentication or file management and manipulation. These actions are separated to enable reuse of security functions by several different file management functions, to avoid recreation of code and possible security holes.



***Figure 2. A UML Diagram of a server side Architecture***

The UML diagram shows the design of the server side of the application. It demonstrates the interaction between server-client and the breakup of the interaction into the different message as seen in the preceding server side diagram. If a client denied the request message than a server received a error message and if the client accept the request then simply the file will be download and saved at the default location in the client disk.



***Figure 3. An HTTP Conversation***

The following figure is a sequence diagram of interactions between the client and the server. It details the events that occur when a client requests a certain action, and the messages sent between the client and the server.

**2.1. DEVELOPMENT APPROACH**

**Data Structures for Data Transmission**

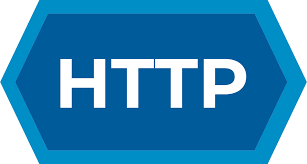
The data structure and technology used to transfer information between client and server is a very important decision. It can affect the work load put on the devices to encode and decode data, which can affect battery life on mobile handsets. It would also alter transmission times, which would bear upon the quality of service to the user.

One option of technology which could be used as a data structure for transmission is JSON. This is simply an array of key-value pairs. Data is parsed into strings, stored in the array, and transmitted as a single string. At the other end the string is parsed as an array, and then each pair must be parsed again into the required types. This is a simple structure, which is relatively easy to implement.

However, there are several issues with this format. First, all data to be sent must be parsed to strings; this is not always a suitable format for certain information. Secondly, there is a computational strain put on the devices to parse data to and from strings. This can increase the time required to act on information, and use more battery, which reduces quality of service to the user.

XML is another technology capable of storing data in a structured manner. It provides a simple, hierarchical structure to information that is human readable. Using generic tags similar to HTML, it is extensible with user created tags to define relations between obaajects and their contents. This involves the use of some software to build the XML schema, a file which lays out a definition of acceptable XML documents which could be used with an application. Again, however, XML uses text to transfer data, which as stated is not always suitable for all data types. It is also not built for efficiency, which is a major concern when using mobile devices. A third option to consider is Google's own protocol buffers. They offer a data structure built natively in C++, Java or Python. For this project Java would make the most sense.

**An HTTP Protocol**



***Figure 4. HTTP***

The **Hypertext Transfer Protocol** (**HTTP**) is an application layer protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web, where hypertext documents include hyperlinks to other resources that the user can easily access, for example by a mouse click or by tapping the screen in a web browser.

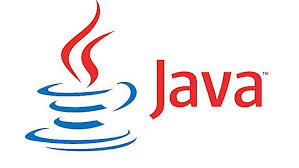
Development of HTTP was initiated by Tim Berners-Lee at CERN in 1989. Development of early HTTP Requests for Comments (RFCs) was a coordinated effort by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C), with work later moving to the IETF.

HTTP functions as a request–response protocol in the client–server computing model. A web browser, for example, may be the *client* and an application running on a computer hosting a website may be the *server*. The client submits an HTTP *request* message to the server. The server, which provides *resources* such as HTML files and other content, or performs other functions on behalf of the client, returns a *response* message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

**2.2. DEVELOPMENT METHODOLOGY**

The methodology used to develop an application can greatly affect the resulting product. A single run through of development can lead to a product which does not work correctly or does not cover requirements as expected. Conversely, too much time testing may result in an application which is not finished. The Waterfall method would provide a clear set of requirements up front, and reduce the amount of time not in implementation, so more focus could be put on development, therefore most likely leading to greater output of functionality. However, there are known weaknesses of this methodology, such as clients not having a fixed set of requirements, and issues with integration of components. For this project, an agile process seemed to be best suited to development. As the supervisor could be seen as the client, requirements maybe checked with him. Weekly meetings provide a chance to update the client on activity and progress. This weekly cycle ensures that, should the project start heading in the wrong direction, it is caught early and can be corrected. This way, the project would provide a correct piece of software, in terms of the clients requests.

**Java Runtime Environment:-**

****

***Figure 5. Java JRE***

The *Java Runtime Environment*, or *JRE*, is a software layer that runs on top of a computer’s operating system software and provides the *class libraries* and other resources that a specific Java program needs to run.

The JRE is one of three interrelated components for developing and running Java programs. The other two components are as follows:

* TheJava Development Kit or JDK, is a set of tools for developing Java applications. Developers choose JDKs by Java version and by package or edition—Java Enterprise Edition (Java EE), Java Special Edition (Java SE), or Java Mobile Edition (Java ME). Every JDK always includes a compatible JRE, because running a Java program is part of the process of developing a Java program.
* The *Java Virtual Machine*, or *JVM***,** executes live Java applications. Every JRE includes a default JRE, but developers are free to choose another that meets the specific resource needs of their applications.

The JRE combines Java code created using the JDK with the necessary libraries required to run it on a JVM and then creates an instance of the JVM that executes the resulting program. JVMs are available for multiple operating systems, and programs created with the JRE will run on all of them. In this way, the Java Runtime Environment is what enables a Java program to run in any operating system without modification.

**Eclipse:**

****

***Figure 6. Eclipse***

To develop a large scale application, an integrated development environment provides an easier way to manage and create software. There are many available, for free or for a fee, each offering different properties. For this project, an IDE aimed at Java would be most suitable. Google offers plugins for Eclipse to enable easier integration for development, debugging and releasing for its Android and App Engine products. This IDE is available for free on all major platforms, providing easy portability between development hardware.

**Spring Boot**

****

***Figure 7. Spring Boot***

Spring Boot is a project that is built on the top of the Spring Framework. It provides an easier and faster way to set up, configure, and run both simple and web-based applications.

It is a Spring module that provides the **RAD (Rapid Application Development)** feature to the Spring Framework. It is used to create a stand-alone Spring-based application that you can just run because it needs minimal Spring configuration.



***Figure 8.Spring Boot Components***

In short, Spring Boot is the combination of spring framework and embedded servers.

In Spring Boot, there is no requirement for XML configuration (deployment descriptor). It uses convention over configuration software design paradigm that means it decreases the effort of the developer.

We can use Spring **STS IDE** or**Spring Initializer** to develop Spring Boot Java application.

We should use Spring Boot Framework because:

* The dependency injection approach is used in Spring Boot.
* It contains powerful database transaction management capabilities.
* It simplifies integration with other Java frameworks like JPA/Hibernate ORM, Struts, etc.
* It reduces the cost and development time of the application.

**Java FX**

****

***Figure 9.JavaFx***

Java FX is a Java library used to build Rich Internet Applications. The applications written using this library can run consistently across multiple platforms. The applications developed using Java FX can run on various devices such as Desktop Computers, Mobile Phones, TVs, Tablets, etc.

To develop GUI Applications using Java programming language, the programmers rely on libraries such as Advanced WindowingToolkit and Swing. After the advent of Java FX, these Java programmers can now develop GUI applications effectively with rich content.

To develop Client Side Applications with rich features, the programmers used to depend on various libraries to add features such as Media, UI controls, Web, 2D and 3D, etc. Java FX includes all these features in a single library. In addition to these, the developers can also access the existing features of a Java library such as Swing.

Java FX provides a rich set of graphics and media API’s and it leverages the modern Graphical Processing Unit through hardware accelerated graphics. Java FX also provides interfaces using which developers can combine graphics animation and UI control.

One can use JavaFX with JVM based technologies such as Java, Groovy and JRuby. If developers opt for JavaFX, there is no need to learn additional technologies, as prior knowledge of any of the above-mentioned technologies will be good enough to develop RIA’s using JavaFX.

**3. System Requirements**

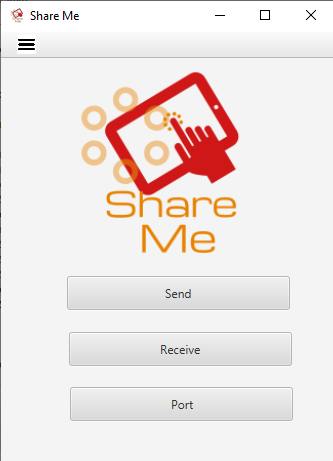
**Software Requirements**

* Operating System : Windows (7 or above)
* Drivers : Java Runtime Environment
* Eclipse
* Spring Boot
* JavaFx

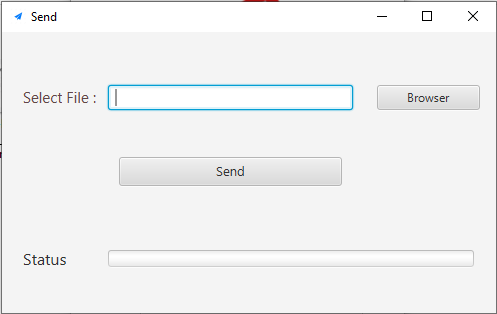
**Hardware Requirements**

* i3 Processor
* 5MB of hard drive space
* 500MB of RAM

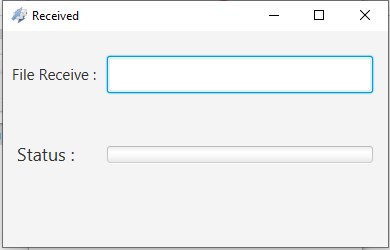
**4. Results**

****

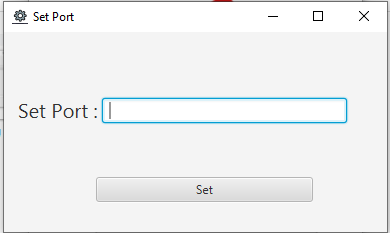
***Figure 10. Home Screen of the application***



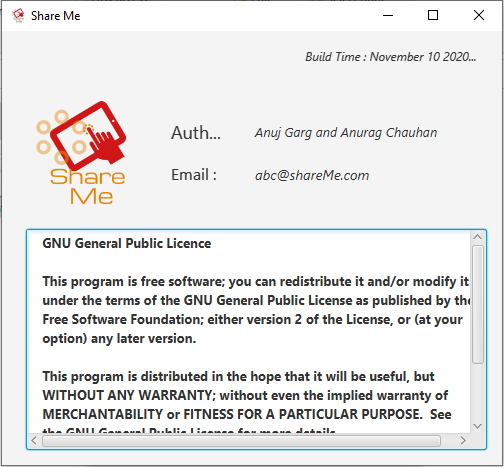
***Figure 11. Send View***



***Figure 12. Receive View***

******

***Figure 13. Set IP address view***

******

***Figure 14. About view***

**5. Conclusion and Future Scope**

Our project has reached the milestone as we designed to meet the requirements. The primary goal of this project is to give an idea about file sharing among desktop with http server. This project has given us an depth information about java technology, spring boot and java FX and its applications in day today life. This project offers several benefits due to the use of simple and easy to understand interface. This project is developed using java and desktop application platform ,which is easy to read, learn, scales well, performs well for being interpreted, and is incredibly well- documented. As the final result meet to our expectation of File sharing. The file sharing application and the platform is ready.

**Future Scope**

There are several expansions to the application which, while unnecessary under the requirements outlined previously, would offer greater functionality to the user and more depth in the application. Some of these are changes to existing code to increase functionality, whilst others are simply adding new components.

* First, implementing a file splitting algorithm when uploading and downloading files. Currently, the application takes a whole file and sends it to the server for storage. This only allows for small files to be sent and stored. By splitting larger files, these could also be stored, allowing the potential for large images and video to be stored and shared.
* Virus scanning of files being uploaded would be a useful extension to add. As so many files belonging to many users would be sending, scanning them would provide security to all users, as well as the application, by preventing viruses from being shared or stored on the server.
* We would add filter offer for searching and filtering of files. As users begin to own and gain access to several boxes, it is quite feasible they may lose track of where a given file is, or wish to see a list of all files of a given file type that they have access to. By offering some form of searching and filtering, they would be able to access this information.
* The final idea could be to offer a chatting and screen sharing extension. Users can chat with each others as well as they can share screen an show then directly the data without sending them.

Of course these are only a few ideas of extensions, there are many more ways to broaden the application and offer more functionality, depending on what is deemed desirable.

**6. Reference**

* [**www.wikipedia.com**](http://www.wikipedia.com)
* [**www.javatpoint.com**](http://www.javatpoint.com)
* [**www.geeksforgeeks.com**](http://www.geeksforgeeks.com)
* [**www.googleimage.com**](http://www.googleimage.com)