1.Introduction

The aim of this project report is to provide an overview of FileZilla, a popular open-source FTP client used for transferring files from a local computer to a remote server. In today's world, where the transfer of files and data is essential for businesses and individuals alike, FileZilla provides a simple, effective, and secure method for transferring files across multiple operating systems.

The report will provide a detailed description of the software, including its key features and functions, and how it can be used to transfer files using various protocols such as FTP, FTPS, and SFTP. The report will also discuss some of the advantages of using FileZilla, such as its open-source nature, its availability across multiple operating systems, and its ability to increase productivity for web developers and system administrators.

In addition, the report will address some of the limitations of FileZilla, such as its lack of support for some file transfer protocols and potential security vulnerabilities. The report will provide some tips for mitigating these limitations and enhancing the security of FileZilla.

Overall, this project report will be useful for anyone looking to understand the basics of file transfer using an open-source FTP client such as FileZilla. It will provide insights into the functionality, features, and limitations of FileZilla, as well as suggestions for how to best use this software to maximize its potential for efficient and secure file transfer.

* 1. Objectives of the project

The main objectives of this project report are as follows:

1.To provide an overview of FileZilla, a popular open-source FTP client used for transferring files from a local computer to a remote server.

2.To describe the key features and functions of FileZilla and how it can be used to transfer files using various protocols such as FTP, FTPS, and SFTP.

3.To discuss the advantages of using FileZilla, such as its open-source nature, its availability across multiple operating systems, and its ability to increase productivity for web developers and system administrators.

4.To address the limitations of FileZilla, such as its lack of support for some file transfer protocols and potential security vulnerabilities, and provide tips for mitigating these limitations and enhancing the security of FileZilla.

5.To provide a comprehensive understanding of FileZilla that will be useful for individuals or businesses looking for a simple, effective, and secure method for transferring files across multiple operating systems.

* 1. Description of the project

This project report aims to provide a detailed overview of FileZilla, an open-source FTP client used for transferring files from a local computer to a remote server. The report will cover the key features and functions of FileZilla, including its user interface, supported protocols, site manager, and transfer logs. Additionally, the report will discuss the advantages of using FileZilla, such as its open-source nature, multi-platform support, and potential for increasing productivity for web developers and system administrators. The report will also address the limitations of FileZilla, such as its lack of support for certain protocols and potential security vulnerabilities, and provide tips for mitigating these issues. Overall, this project report will be a comprehensive guide for individuals or businesses looking to use FileZilla for efficient and secure file transfer.

* 1. Scope of the project

The scope of this project report is to provide a comprehensive understanding of FileZilla as an open-source FTP client used for transferring files from a local computer to a remote server. The report will cover various aspects of FileZilla, including its user interface, supported protocols, key features, advantages, and limitations.

In terms of the user interface, the report will provide a detailed description of how FileZilla looks and functions, including its various buttons, menus, and tools. This will help users navigate the software and perform file transfers more efficiently.

The report will also cover the different protocols supported by FileZilla, including FTP, FTPS, and SFTP, and explain how each protocol works. This will enable users to select the appropriate protocol for their specific needs and requirements.

Furthermore, the report will discuss the key features of FileZilla, such as the site manager, transfer speed limit, and transfer logs. This will help users understand the capabilities of FileZilla and how to use them effectively for file transfer tasks.

In addition, the report will discuss the advantages of using FileZilla, such as its open-source nature, multi-platform support, and potential for increasing productivity for web developers and system administrators. This will provide users with a better understanding of the benefits of using FileZilla for their file transfer needs.

The report will also address the limitations of FileZilla, such as its lack of support for certain protocols and potential security vulnerabilities, and provide tips for mitigating these issues. This will help users understand the potential risks associated with using FileZilla and how to use the software safely and securely.

Overall, the scope of this project report is to provide a comprehensive guide to FileZilla, which will help users understand the software, its capabilities, and limitations, and how to use it effectively and safely for efficient and secure file transfer.

2.System description

2.1 Target system description

FileZilla is a cross-platform software, which means it is compatible with multiple operating systems. The software can run on Windows, macOS, and Linux operating systems. Therefore, the target system for this project is any computer running one of these operating systems.

For Windows users, FileZilla can run on Windows 7, 8, 8.1, and 10. The software requires an Intel or AMD processor with a clock speed of 1 GHz or higher, 512 MB of RAM, and 100 MB of free disk space. Additionally, FileZilla requires an active internet connection to transfer files.

For macOS users, FileZilla can run on macOS 10.9 or later. The software requires an Intel processor with a clock speed of 1 GHz or higher, 512 MB of RAM, and 100 MB of free disk space. Additionally, FileZilla requires an active internet connection to transfer files.

For Linux users, FileZilla can run on various distributions, such as Ubuntu, Debian, Fedora, CentOS, and more. The software requires a modern kernel with glibc 2.12 or higher, 512 MB of RAM, and 100 MB of free disk space. Additionally, FileZilla requires an active internet connection to transfer files.

In terms of network requirements, FileZilla supports various protocols for transferring files, including FTP, FTPS, and SFTP. Therefore, the target system should have access to a network that supports these protocols.

Overall, the target system for this project is any computer running Windows, macOS, or Linux operating systems, with the required system specifications and an active internet connection, capable of connecting to a network that supports the required file transfer protocols.

2.2 Assumption and dependencies (if applicable)

Assumptions:

1.Users have basic knowledge of file transfer protocols and how to connect to a remote server.

2.Users have the necessary permissions to access the remote server and transfer files.

3.Users are familiar with the operating system they are using and know how to install and run software.

4.Users have a stable and reliable internet connection to transfer files.

Dependencies:

1.FileZilla is an open-source software, and users must ensure that they download it from a reliable source to avoid any potential security risks.

2.FileZilla supports various file transfer protocols, and users must ensure that the remote server supports the protocol they want to use.

3.Users must have the necessary permissions to access the local and remote directories involved in file transfer tasks.

4.The performance of FileZilla is dependent on the speed and stability of the internet connection. Therefore, users must ensure that they have a stable and reliable internet connection for efficient file transfer.

2.3 Functional and Non-Functional dependencies(if any)

Functional Dependencies:

1.File Transfer Protocol (FTP), FTPS, and SFTP must be supported by the remote server for file transfer.

2.The user interface of FileZilla should be intuitive and easy to use, enabling users to navigate the software and perform file transfer tasks efficiently.

3.The transfer speed of FileZilla should be fast and reliable, ensuring that file transfer tasks are completed quickly and accurately.

4.The transfer logs and site manager features of FileZilla must be functional, providing users with a clear and detailed overview of their file transfer tasks.

Non-Functional Dependencies:

1.Security: FileZilla must ensure that file transfer tasks are secure and protected from potential attacks, such as man-in-the-middle attacks or brute force attacks.

2.Reliability: FileZilla must ensure that file transfer tasks are completed accurately and without any errors or data loss.

3.Performance: FileZilla must ensure that it operates efficiently, utilizing system resources effectively, and enabling users to complete file transfer tasks quickly and effectively.

4.Compatibility: FileZilla must be compatible with multiple operating systems and support multiple protocols for file transfer.

5.Accessibility: FileZilla must be accessible to users with disabilities, providing features such as keyboard shortcuts or screen readers to ensure that everyone can use the software effectively.

In summary, the functional dependencies of the project are centered around ensuring that FileZilla is easy to use, fast, and reliable for file transfer tasks. The non-functional dependencies are focused on ensuring that the software is secure, compatible with multiple operating systems, and accessible to all users.

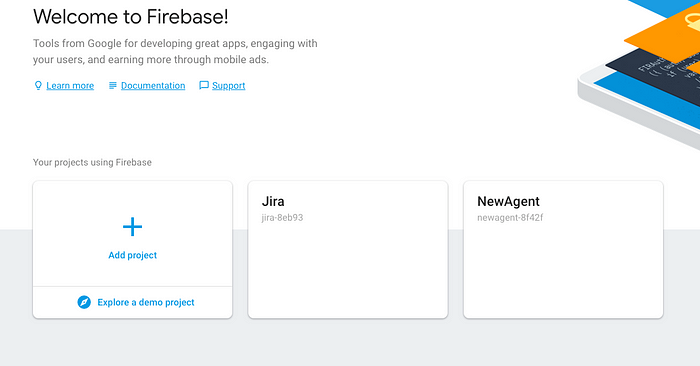
2.4 Data set used in support of your project(if any, then paste the link)

3. Analysis report

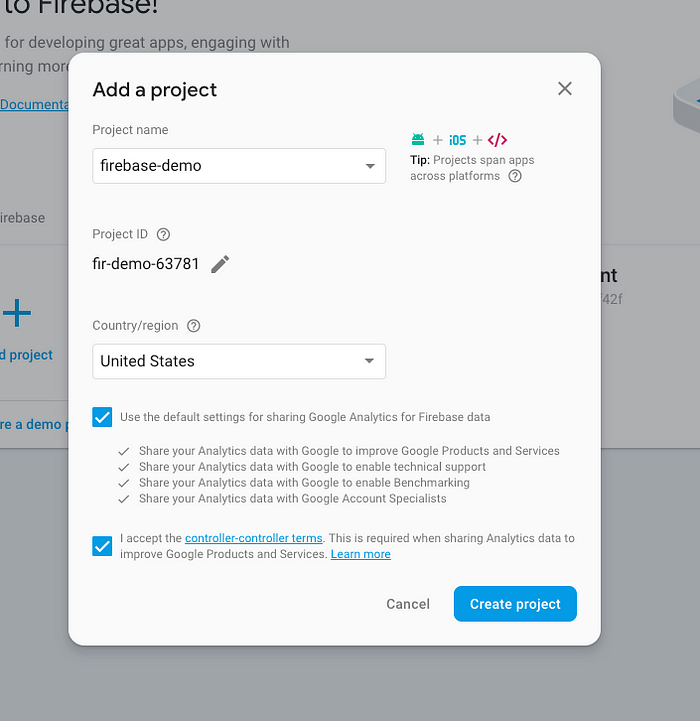
3.1 System snapshots and full analysis report

# Step 1: Creating firebase project

Go to [**firebase**](https://firebase.google.com/) and sign in with your Google account



then create new project, enter your project name (**firebase-demo** in my case)



# Step 2: Firebase Login

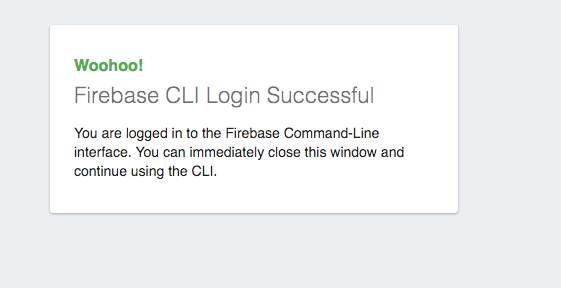
Now come back to the command line and go to your project folder

cd firebase-demo

First we have to login into firebase from command line. Type in the following command.

firebase login

It will take you to the sign-in page in the browser, once you’ve successfully logged in it will show you something like this



on browser

Graphical user interface, text, application, email

Description automatically generated

on command line

# Step 3: Initializing project

To initialize firebase project you have to enter the command

firebase init

Then you have to select **Hosting**feature and click enter



Then it will ask you to select firebase project, select project which we created in **step 1**(firebase-demo in my case)

Text

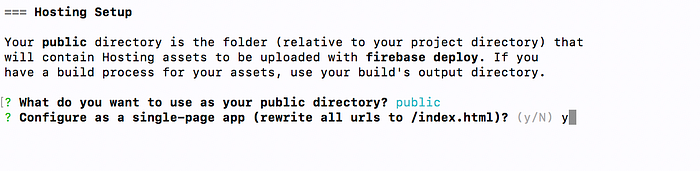
Description automatically generated

Then it will ask you enter the main folder in which all your website assets are present. (public folder in my case)

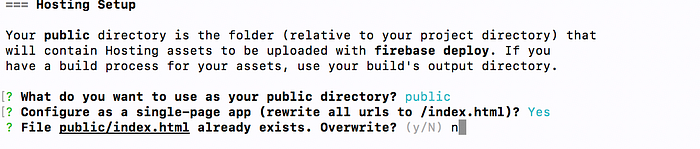
Text, letter

Description automatically generated

It will ask you whether your application is single page or not, for now enter **y**



It will try to override your **index.html** file, to avoid doing that enter **n**

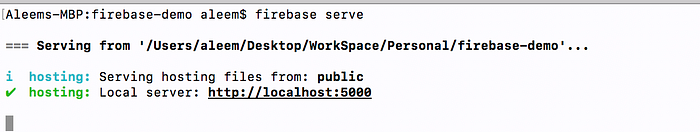


# Step 4: Checking setup (optional )

If everything goes right you can check you website locally by running command

firebase serve

It will run you website locally on [http://localhost:5000](http://localhost:5000/) by default.



# Step 5: Deployment

For deployment of your project you have to run command

firebase deploy

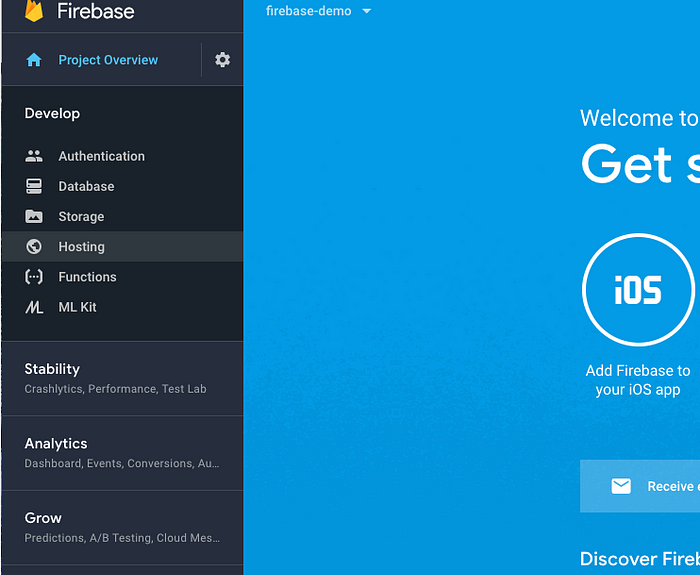
Graphical user interface, text, application, email

Description automatically generated

**Congratulations! your website is now live**, you can check by going to url which is provided in the command line in my case it is <https://fir-demo-63781.firebaseapp.com/>

# Bonus Step: Connecting your domain

Go to [firebase console](https://console.firebase.google.com/), select the project which you’ve created (firebase-demo in my case) select hosting tab under develop tab (develop -> hosting)

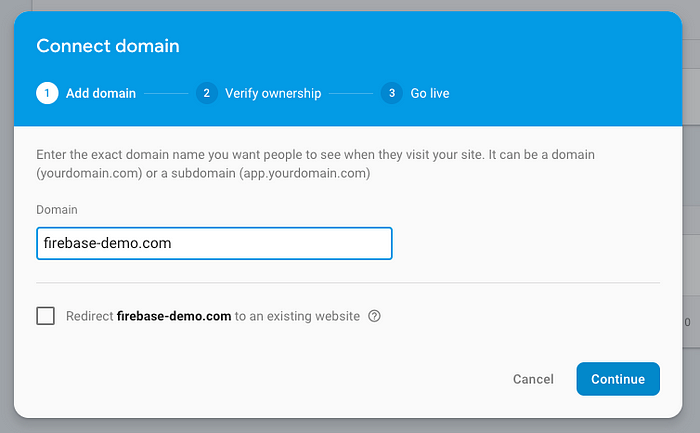


Then click on connect domain

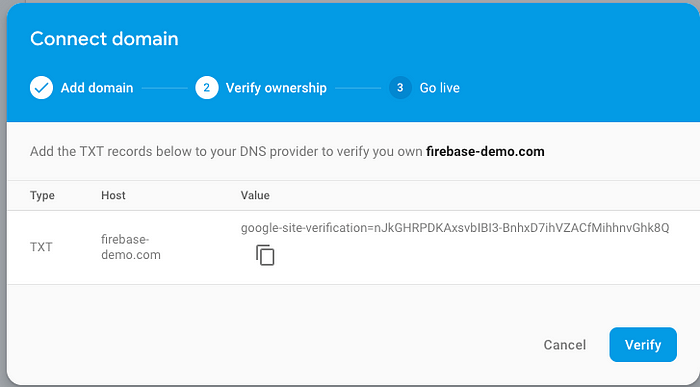
Graphical user interface, application, Teams

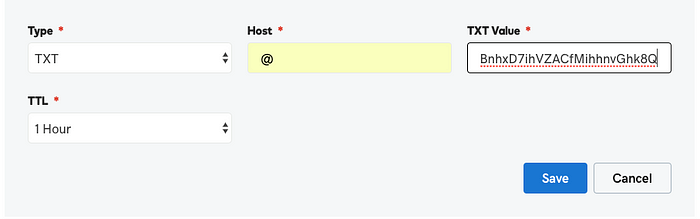
Description automatically generated

Then enter your domain url



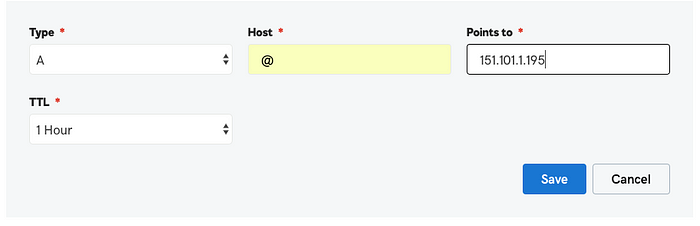
Then add the txt record and value into your domain provider dns

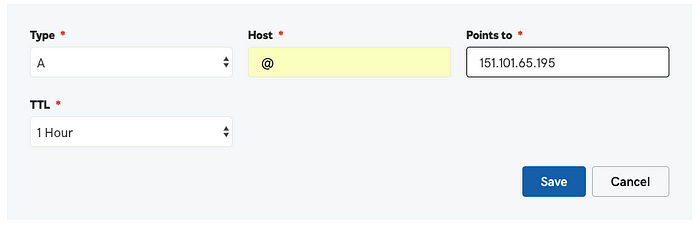




Godaddy Example

Then add **A**records to you domain provider dns





It will take upto **5min-1hour** depending on your domain provider to update dns records

**That’s how easy it is to setup your website on the firebase.**

4.References/bibliography

Wikipedia

Github

Medium.com

5.Github link

<https://github.com/Anikchaudhary/Project_report>