Alchemy Downloader

Yunus Irfan Maniyar
Information Technology Department
M. H. Saboo Siddik Polytechnic

Amaan Fakir Mohammed Shaikh
Information Technology Department
M. H. Saboo Siddik Polytechnic

Faiz Salim Shaikh
Information Technology Department
M. H. Saboo Siddik Polytechnic

Mujibullah Khan
Information Technology Department
M. H. Saboo Siddik Polytechnic

Sameera Khan
Information Technology Department
M. H. Saboo Siddik Polytechnic

Abstract **— We all are witnessing the growth of the Internet Community. High number of Internet users are leading to a lot of stress on servers of websites providing required files. This load on server and network obviously results in slower speed of downloading. Alchemy Downloader is inspired by these problems faced in downloading large files. Alchemy Downloader is a service which provides faster downloading of files with the help of the servers with high bandwidth networks provided by cloud providers like Google Cloud Platform, Amazon Web Services, Heroku and Oracle Cloud. With Alchemy Downloader users can take high-speed network of Google Drive™ in use for getting their files quickly. This paper will provide explanation about the things that make this service possible. This service will consists of two main thing, the Android Application and the server application, roles of each of these main components will be mentioned. This paper will also provide information about the pre- development period, in which we found that how this problem of slow download speed can be solved with the help of the available tools. The methodology that we used to develop this project as well the reasons that lead us to chose the particular methodology will be discussed as well in this paper.**

Keywords – Download, bandwidth, file, web file

I. INTRODUCTION

In this digital age, everyone of us must have come across with downloading a file. For most of us, it's something that we frequently do. Whether it be downloading a document, a video or some packages of huge size. Most of the websites, that provides these files and also the sites that let users share their files, does not have enough resources to provide faster downloading to users located at different parts of the world. And thus, most of the times, when you download a file, you don't get a good speed. Alchemy Downloader is developed to try to solve this very problem. Thanks to the API provided by generous Google that lets us take in use the powerful resources provided by Google Drive™.

In this paper, we will be mentioning the technologies that powers this great tool. We will also mention the ways that we used to get the benefits of the technologies. We'll also give you a glimpse of our journey in developing this service.

II. MOTIVATION

The community of Internet is ever growing. The resources of the websites, used by users to get their desired contents, generally be under a heavy load. This load on server and network obviously results in slower speed of downloading. So, we started thinking of ways that can help users in solving this issue.

What we found was that not all websites face this issue of slow downloading. Services provided by huge companies like Google, Microsoft, Samsung, etc. do not face these issues. For an instance, when you download a file stored in your Google Drive space, you'll find that the speed of downloading is much better then most of the other websites. Most of the times it gives us the complete speed of our Internet Plan, and sometimes it also goes beyond our Internet Plan's speed.

So, we thought, what if we can get our desired file in our Google Drive space and download it from there. This would help in solving the issue. And this technique motivated us to create such a service that would facilitate users with this technique to download the files very quickly, while hiding the complexity behind it, making whole process just a matter of few clicks just as someone would normally do in a web browser to download a file.

III. PROBLEM STATEMENT

Downloading files from certain websites takes a lot of time because of the load that the particular server and its network infrastructure faces because of high number of users and less resources. Availability of great service like Google Drive provides a way to get relief of these issues.

IV. METHODOLOGY

This service mainly consists of two things - the mobile app and the server. We designed and finalized the system architecture. We wanted to create this service as a web app, because of the fact that web apps does not get much affected by the change in platform. Unfortunately, this service consists of some requirements that were not feasible to implement in a web app, and thus we had to go for other options to develop the mobile app.

Our team had quite good experience in JavaScript and web development but we did not had experience in developing mobile application in their native way. So we had to find an option which can help us develop the mobile app with the skills that we already had. We did some research and found a mobile application development framework named Apache Cordova.

Cordova makes it possible to create mobile application using web technologies.

With the help of cordova and it's plugins, We developed our application as per our system architecture using web technologies. Some plugins that would have helped us to use device functions were required to be modified to achieve our requirements. These modifications were platform-dependent so we had to modify the codes of all the platforms that we would have like to implement. Doing so was not feasible for our small team, so we decided to choose a single platform for our mobile application and thus we went ahead with Android as platform for our app. We did the modifications in Android specific codes of the plugins and then implemented them in our mobile app. With all these and more, the Android Application was prepared.

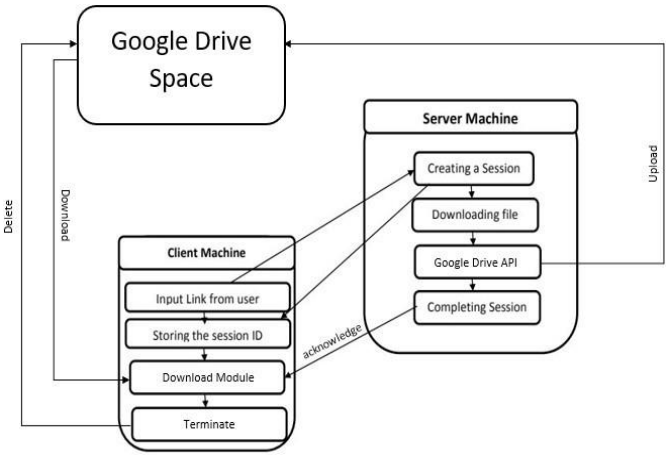
As we had good experience in JavaScript, we decided to take nodejs as our main runtime environment for our back-end server. The most important thing for the back-end was the implementation of the Google Drive API. We registered our project in Google Developer Console and obtained the required configuration information and files to use the API. The second important thing was the component responsible for downloading the file in our back-end server. In order to make our service compatible with most of the downloading methods, we had to cover the requirements of different downloading methods. We completed our back-end development and deployed it in an instance of Google Cloud Platform's, Compute Engine.

Following are the list of websites and the average amount of time taken by our service to download afile from them, in comparison to the conventional downloading process.

- Archive.org : 85%
- Mediafire : 35%
- Pcloud : 36%
- altushost-swe : 78%
- iweb : 63%
- liquidtelecom : 83%
- inode : 52%
- nkn : 42%
- iitk : 28%

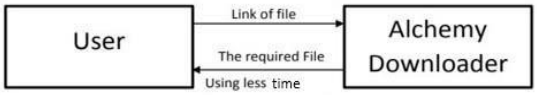
V. SYSTEM ANALYSIS

A. System Architecture

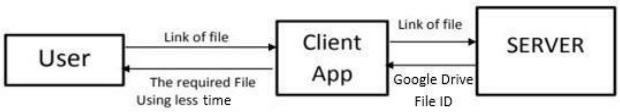


B. DFD

- **Level 0**



- **Level 1**



VI. RESULT

Through our service user will able to download their files easily and with more bandwidth utilization. After developing this app we provided our app with different file links in order to test the efficiency of our app and download speed.

CONCLUSION

We hope that Alchemy Downloader will help the p in downloading their files quickly. We wil continue to work on improving this service by making it more efficient. Important work will be done to make our service more flexible and cost- effective in future. We will make sure to introduce new features and to cover more download methods.

REFERENCES

Following websites were referred in development of our project.

[1] Google Drive API: https://developers.google.com/drive/api/v3/reference
[2] NodeJS: https://www.w3schools.com/nodejs/
[3] Cordova Plugins https://www.npmjs.com/package/ cordova-plugin-inappbrowser
[4] https://github.com/apache/cordova-plugin-file- transfer#readme
[5] GCP Compute Engine: https://cloud.google.com/compute/docs
[6] Client Side Scripting: https://www.tutorialspoint.com/javascript/index.htm
[7] Server Side Scripting: https://www.upwork.com/hiring/development/serverside-scripting-back- endwebdevelopment-technology
[8] Cordova: https://cordova.apache.org/docs/en/latest/guide/cli/