



WHY JOIN THE NAVY WHEN YOU CAN BE A PIRATE?

Musifre

Music was never meant to be SOLD.

By Yash Dhingra

School: Salwan Public School, MV-III

Competition Category: HackHeist

Index

Sno.	Particulars	PageNo.
1	Preface	2
2	Introduction to Project	3
3	The Objective of the Project	4
4	Scope of the Project	5
5	Input/Output Requirement	6
6	Hardware and Software	7
7	System Design	8
8	Future Scope of the Project	9
9	Conclusion	10
10	Reference or Bibliography	11

Preface

This report on "**Musifre**" is published to set the foundations of the project.

Music is the art of arranging sounds in time through the elements of melody, harmony, rhythm, and timbre. It is one of the universal cultural aspects of all human societies.

For years companies have been leveraging music to earn fortune, where it should have been free of cost and available to everybody.

The Aim of this project is to create a breakthrough GUI Application which acts as a frontend to stream & play music from various sources free of cost.

Currently the project only supports MacOS, Linux & Windows Operating system and in future will be scaled to support multiple platforms like iOS, iPadOS, Android.

Introduction

Music is heard by millions of people every day and it's a crucial part of our lives. For years companies have been using music to earn profit by selling either digital copies or charge monthly subscription, for music which can be heard and enjoyed free of cost.

“Free Music Services” either steal users info or play thousands of ads rendering miserable listening experience for users and forcing them to upgrade by paying them a premium.

Musifre is a revolutionary tool which acts as a front end and downloads music from various sources, providing an ad free experience for our users. It also manages to provide an effortless music listening experience which removes the hassle of going to the internet to search for music on different sites.

The Project will be completely Open Source so that people all around the globe can help and improve our application.

We intend to make this application simple and effortless so that the user can start using the application right away with **no registration process involved**.

We've also added the functionality for user to transfer song library to a different computer.

While making this application we've also spent great emphasis on **privacy**.

All the process are **in-device** and are performed locally with **no cloud data logging**.

Objective

Our Objective for this Project is to fulfil the points mentioned below

- 1) **Music Streaming:** We'd like to give the user seamless music listening experience without charging a dime.
- 2) **Offline Listening:** Not only to stream music online but offline as well.
- 3) **Music Player:** Allow the user to listen to music directly from the application.
- 4) **Privacy:** To make all the processes in-device which are performed locally with no cloud data logging.
- 5) **Transferring data:** Ability for user to transfer his or her song library to a difference compatible device.
- 6) **Open Source:** Open Source allows people from all around the globe to help improve the application while also ensuring that no data is being stolen or compromised.
- 7) **Optional Music Player:** One of the problems with music services is that they use proprietary music formats preventing users to listen music on different music player of their choice, well we indent to use wav format which allows user to use any supported music player, of their own choice.

Scope

We Intend to deploy this application all around the globe. Since all the processing is done locally no cloud systems are involved, allowing millions and millions of people to use this application at once. Currently this application only supports three of the most common desktop platforms

1. MacOS
2. Linux
3. Windows

Our Application supports both ARM and x86 computing architectures making this application future ready.

Input/Output Requirements

Creating such an application on classic terminal would be boring and so 1970s. Hence its necessary for us to create this application with a GUI so that everyone will be able to interact with our application naturally without a start-up guide. We are a yardstick when it comes to design. We've come up with a simple yet functional UI which is divided in just two subsections

1) Music String Input

2) The Audio Player

Section	Input Requirements	Output
Music String Input	<ul style="list-style-type: none">• Name of the track• Confirmation downloading	<ul style="list-style-type: none">• Downloading Status
Audio Player	<ul style="list-style-type: none">• Play• Pause• Dropdown Menu(Song Selection)	<ul style="list-style-type: none">• Song Being Played

Hardware & Software Req

Hardware Requirements:

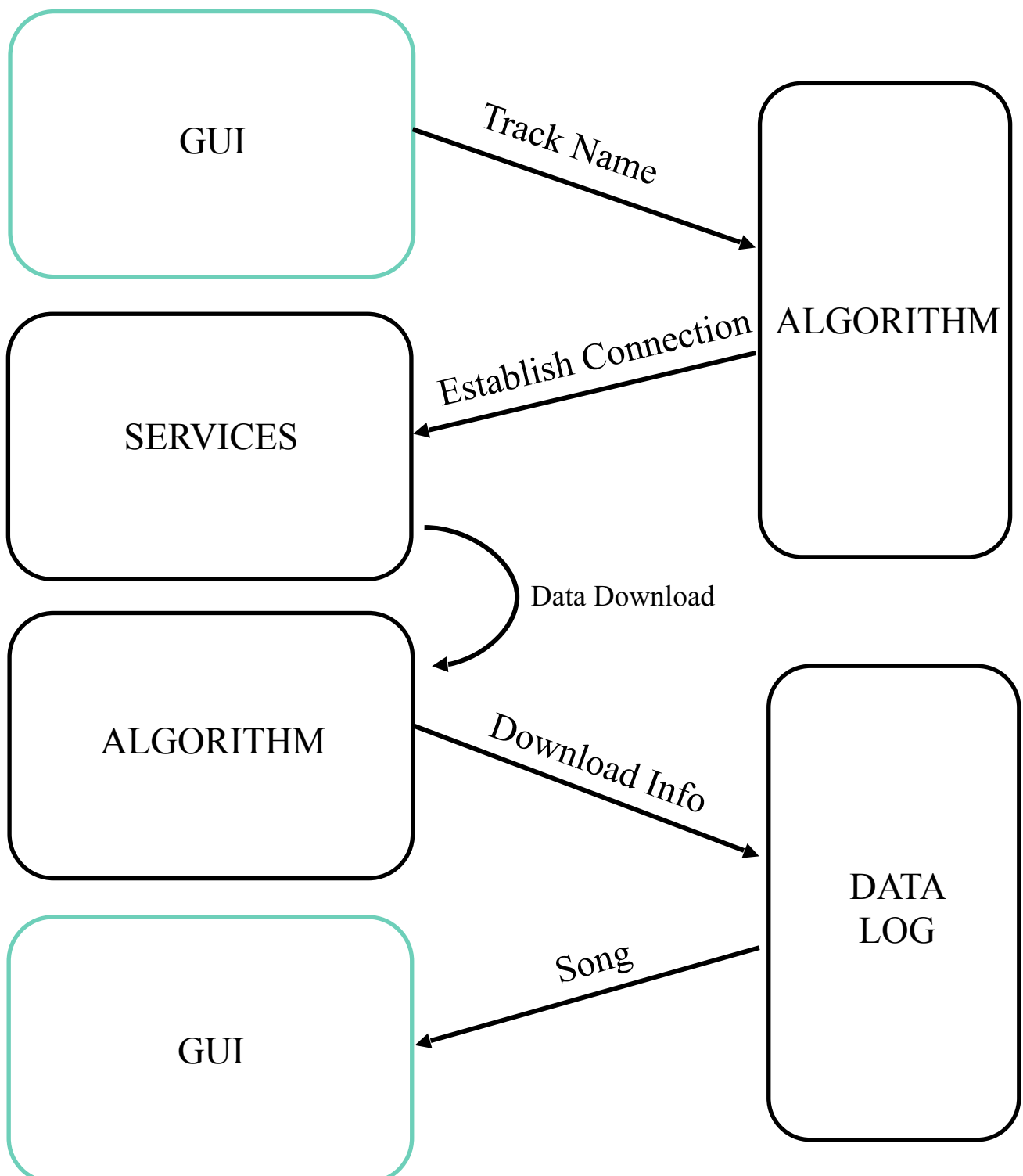
1. A Computer Running **MacOS**(Since Under MacOS all platforms can be tested)

Software Requirements:

1. Python3
 - PyQt
 - Subprocess
 - Pytube
 - Urllib
 - Re
 - CSV/Binary
2. Qt Designer
3. VS Code

System Design

The Code Design is best explained with a flow chart(Given Below).



Future Scope

This program itself is fully functional although nothing in this world is perfect and so is the case with Musifre. In Future we intend to

1. Push this application to audiophiles by including lossless audio playback.
2. Bring Musifre to iOS, iPadOS & Android.
3. Offload Algorithm in the form of API in cloud for slow computers for improved speed.
4. Siri, Google Assistant Support which will allow people to play music handsfree.
5. Make it so it runs on a Browser without needing python to be locally installed on the computer

Conclusion

This project has taught us a lot about computing. In the development of this project we had to go through multiple problems, which not only improved our skills but changed the way we approach problems.

To make this concept possible we had to make multiple layers of code which interacted with various parts of the software. To see such complex work flow run smoothly and completely hidden from the user is practically magic.

This program does not end here, as explained in the future scope, this program can be expanded further.

References

1. **PyQT Tutorial YouTube** - <https://www.youtube.com/watch?v=Vde5SH8e1OQ&t=2s>
2. **CSS Tutorial (W3 Schools)**-
<https://www.w3schools.com/css/>