Harmonica AMT

Student name: Liam Riche

Student number: B00789989

Course: Computing Systems PT

Date: 2022/2023

Supervisor: Glenn Hawe

Contents

[Abstract 3](#_Toc131092236)

[Acknowledgements 3](#_Toc131092237)

[Listing of Figures 3](#_Toc131092238)

[Listing of Tables 3](#_Toc131092239)

[Listing of Code Listings 3](#_Toc131092240)

[Introduction 3](#_Toc131092241)

[Background 3](#_Toc131092242)

[Project Aims 3](#_Toc131092243)

[Project Objectives 3](#_Toc131092244)

[Outline of Dissertation Structure 3](#_Toc131092245)

[Literature Review 3](#_Toc131092246)

[Existing Solutions 3](#_Toc131092247)

[Current System 3](#_Toc131092248)

[Desired Solution 3](#_Toc131092249)

[Potential Existing Solutions 3](#_Toc131092250)

[Weighing up the options 3](#_Toc131092251)

[Technical Background 3](#_Toc131092252)

[Choosing a Software Development Life-Cycle 3](#_Toc131092253)

[Considerations 3](#_Toc131092254)

[Software Development Life-Cycles 3](#_Toc131092255)

[Choosing a life-cycle model 3](#_Toc131092256)

[Choosing the tools 3](#_Toc131092257)

[Identifying the needs 3](#_Toc131092258)

[Choosing a web framework 3](#_Toc131092259)

[Choosing a database backend 3](#_Toc131092260)

[Development Environments and Other Tools 3](#_Toc131092261)

[Project Planning 3](#_Toc131092262)

[Resource Consideration 3](#_Toc131092263)

[Risk Assessment 3](#_Toc131092264)

[Data Management 4](#_Toc131092265)

[Knowledge and Skills Required 4](#_Toc131092266)

[Involving Stakeholders 4](#_Toc131092267)

[System Requirements 4](#_Toc131092268)

[Requirements Gathering 4](#_Toc131092269)

[Requirements 4](#_Toc131092270)

[Functional Requirements 4](#_Toc131092271)

[Non-Functional Requirements 4](#_Toc131092272)

[Design 4](#_Toc131092273)

[UX Design 4](#_Toc131092274)

[Design Principles 4](#_Toc131092275)

[UI Designs 4](#_Toc131092276)

[UI Implementations 4](#_Toc131092277)

[Data Modelling 4](#_Toc131092278)

[Database Design 4](#_Toc131092279)

[Data Models 4](#_Toc131092280)

[Implementation 4](#_Toc131092281)

[Architecture Overview 4](#_Toc131092282)

[Architecture 4](#_Toc131092283)

[Additional libraries used 4](#_Toc131092284)

[Code Overview 4](#_Toc131092285)

[Obstacles Encountered 4](#_Toc131092286)

[Testing & Evaluation 4](#_Toc131092287)

[Testing 4](#_Toc131092288)

[Unit Testing 4](#_Toc131092289)

[System Testing 4](#_Toc131092290)

[Manual Testing 4](#_Toc131092291)

[Compatibility Testing 4](#_Toc131092292)

[Evaluation 4](#_Toc131092293)

[Evaluation of technologies used 4](#_Toc131092294)

[Evaluation of system 4](#_Toc131092295)

[Results 4](#_Toc131092296)

[Fulfilment of project objectives 4](#_Toc131092297)

[Fulfilment of functional requirements 4](#_Toc131092298)

[Fulfilment of non-functional requirements 5](#_Toc131092299)

[Additional work completed 5](#_Toc131092300)

[Conclusion 5](#_Toc131092301)

[Reflecting on the project 5](#_Toc131092302)

[Project scope 5](#_Toc131092303)

[Products produced 5](#_Toc131092304)

[Managing the process 5](#_Toc131092305)

[Suggestions for future improvements 5](#_Toc131092306)

[References 5](#_Toc131092307)

[Appendices 5](#_Toc131092308)

[Appendix 1 – Initial Gantt Chart 5](#_Toc131092309)

[Appendix 2 – Unit tests listing 5](#_Toc131092310)

# Abstract

# Acknowledgements

# Listing of Figures

# Listing of Tables

# Listing of Code Listings

# Introduction

## Background

This project came into being from my interest in Automatic Music Transcription services (AMTs) that take an audio input for a specific instrument and convert it into a sheet music notation. When looking into these I found many AMTs for popular instruments like the guitar and piano, but very few for more niche instruments. I chose to base my project around the harmonica as it is niche enough that there is not an AMT already created for it, as well as the fact that the harmonica has enough unique aspects to it that a tailored solution would offer more to a user compared to a general use AMT (e.g. specific tunings, blow direction, etc.)

The intended user base for this would be people who play the harmonica and either want to:

1. Find the notes to a song they are listening to
2. Create sheet music for a song they have come up with to share with others

### Project Aims

This project aims to create a machine learning product that will ‘listen’ to music passed to it and gather information on:

* tempo (speed at which music is played),
* note length (how long a note is held / a silence lasts)
* pitch of note (which note is being played)
* key (the tuning of the instrument)
* blow direction (is the musician sucking or blowing into the harmonica)

the AMT will then piece these together with a renderer to generate sheet music in the notation of a harmonicas playstyle and return it to the user who provided the song.

The software would be created in jupyter notebook and potentially packaged into a website or application once AMT has been completed

## Project Objectives

## Outline of Dissertation Structure

# Literature Review

## Existing Solutions

## Current System

## Desired Solution

## Potential Existing Solutions

## Weighing up the options

# Technical Background

## Choosing a Software Development Life-Cycle

* Agile was chosen for use on the project for its benefits below:
  + Easy and swift change adaption in case of change or oversight in original plans
  + Transparency in current state of project compared to projected progress
  + Allows for continual testing of overall product more often than waterfall model
  + Better stakeholder engagement, updates on current project states more often

## Considerations

## Software Development Life-Cycles

## Choosing a life-cycle model

## Choosing the tools

### Identifying the needs

### Choosing a web framework

### Choosing a database backend

### Development Environments and Other Tools

# Project Planning

## Resource Consideration

## Risk Assessment

## Data Management

## Knowledge and Skills Required

## Involving Stakeholders

# System Requirements

## Requirements Gathering

## Requirements

### Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Requirement** | **ID** |  |
| R1 | Application must be user friendly and easy to navigate |  |  |
| R2 | application must be able to record or upload sound clip |  |  |
| R3 | Application must return error message if file uploaded is not compatible |  |  |
| R4 | Application must return error message if file uploaded is longer than max length |  |  |
| R5 | Application must be self contained and have ability to function on its own |  |  |
| R6 | Application should return sheet notation of an uploaded clip in pdf format |  |  |
| R7 | Application should have accessible gui menu |  |  |

### Non-Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Requirement** | **ID** |  |
| R1 | Application must be user friendly and easy to navigate |  |  |
| R2 | application must be able to record or upload sound clip |  |  |
| R3 | Application must return error message if file uploaded is not compatible |  |  |
| R4 | Application must return error message if file uploaded is longer than max length |  |  |
| R5 | Application must be self contained and have ability to function on its own |  |  |
| R6 | Application should return sheet notation of an uploaded clip in pdf format |  |  |
| R7 | Application should have accessible gui menu |  |  |

# Design

## UX Design

Graphical user interface, diagram

Description automatically generated

### Design Principles

### UI Designs

### UI Implementations

## Data Modelling

Replace this section with csv file editing or similar

### Database Design

### Data Models

# Implementation

## Architecture Overview

### Architecture

### Additional libraries used

## Code Overview

## Obstacles Encountered

# Testing & Evaluation

## Testing

### Unit Testing

### System Testing

### Manual Testing

### Compatibility Testing

## Evaluation

### Evaluation of technologies used

### Evaluation of system

# Results

## Fulfilment of project objectives

### Fulfilment of functional requirements

### Fulfilment of non-functional requirements

### Additional work completed

# Conclusion

## Reflecting on the project

### Project scope

### Products produced

### Managing the process

## Suggestions for future improvements

# References

# Appendices

## Appendix 1 – Initial Gantt Chart

A picture containing graphical user interface

Description automatically generated

## Appendix 2 – Unit tests listing

## https://blog.pythonanywhere.com/169/