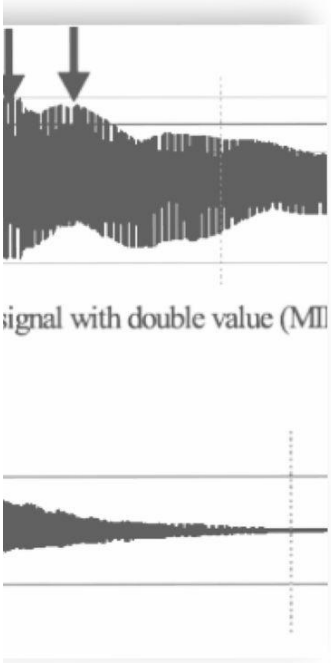
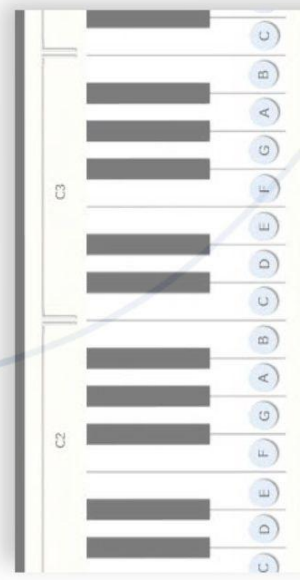
Project Specifications Report

Group 20



pianissimo



Team Members:
Melisa Kamacı
Tamer Alkım Tokuç
Nazem Yıldırım
Begüm Hatipoglu

Project Specifications Report

1. Introduction

This report is intended to give a brief description of Pianissimo by stating its initial requirement specifications with a follow-up discussion of its further project constraints and professional and ethical issues. The content is to define fundamental requirements that will guide the development process of Pianissimo.

1.1 Description

Pianissimo aims to make practicing the piano easier for anyone who is interested in this instrument by offering sheet music for requested piano compositions. Its main feature is the extraction of the underlying notes using a variety of technologies and providing them in either PDF or MIDI format after the upload of plain piano music in an audio format. The response rate is expected to be in seconds reduced from hours of manual work. The software will initially be available as a program for computers. In future development goals, it will be available as a mobile application and web application.

1.2 Constraints

1.2.1 User Interface Constraints

- The graphical user interface (GUI) shall meet industry standards in order for the GUI to look the same in different operating systems.
- The GUI design shall be kept simple as to not increase the complexity and keep the application easily usable.
- The GUI will follow material design patterns.

1.2.2 Hardware Constraints

- The application requires Python 3, therefore minimum hardware specifications of Python 3 will be applicable.
- Enough disk space should be available.
- In addition to these, if the web interface of the program has been used, internet connection is required.

1.2.3 Software Constraints

- The software requires Python 3 to be installed in the computer.
- Management of additional packages will be done using shipping frameworks, such as Docker, reducing dependency on individual external packages.

1.2.4 Operational Constraints

- This section will discuss the restrictions on how Pianissimo will work in its own environment.
- The time that Pianissimo outputs the notes and the estimated time for their feedback under intensity: The efficiency of the system is kept at an average of 90 seconds, and a maximum of 600 seconds if the system is under heavy load.
- There will be an average of 10 current projects and a maximum of 50 projects stored in the system.
- There will be an average of 50 users currently logged into the system and a maximum of 100 users.
- The system will have a Mean Time Between Failures (MTBF) of 240 hours.
- The system will have a Mean Time to Repair (MTTR) of 2 hours.
- Maximum Error or Error Rate
- Minor Errors (incorrect information display, other screen errors) these will be kept at a maximum of 5 errors/KLOC.
- Critical Errors (such as users having too much access, not enough access) these will be kept at maximum 1 errors/KLOC.
- Major Bugs (such as system crash, data loss, inability to generate output) these will be kept at maximum 0.25 errors/KLOC.

1.2.5 Design Standards Constraints

- Main program will use Python 3.
- Future development goal mobile applications will use Kotlin/Java.
- Web app data transmission will be isolated from any third party access.
- The audio files that are to be uploaded by the users must have 224kbps or higher bitrate in order for the application to process the audio.
- The audio files that are to be uploaded by the users must not have file size bigger than 250 MB.

1.2.6 Site Adaptation Constraints

- Future development goal of a web app will be built on the React framework.
- The website will be able to handle 50 users at peak.
- The website will check file extension to ensure the audio file has been uploaded.
- The website will handle up to 250 MB of music files per user at a time.

1.2.7 Communications Constraints

This section contains a list of communication devices or protocols that Pianosimo must interact with the WEB application.

- The website will use HTTPS as a connection protocol.
- Handling printer connections will be done by OS.

The system must communicate over TCP/IP. All devices connected to the Internet can communicate with each other thanks to this method. Computers connected to a network communicate with TCP/IP protocol to receive and transmit data among themselves. Thus, data transfer is ensured.

1.2.10 Data Management Constraints

[2]:This section provides a detailed description of the restrictions for data flow to data management software and equipment outside of Pianosimo's scope.

- The system must be able to interface with other parts in accordance with their requirements.

1.3 Professional and Ethical Issues

Pianissimo will operate in accordance with the general principle of user privacy which indicates that a minimum amount of requisite personal data should be gathered in a system [1]. Likewise, any personal data that is associated with a user will be never shared with third parties to honor confidentiality [1]. To build up a system that ensures these necessities, it will be made sure that security is the primary concern and the app will be equipped with reliable foundations to establish it [1]. The overall process will be carried out with the intention of maintaining the public good [1] since Pianissimo is expected to be highly utilized by people having piano education.

2. Requirements

2.1 Functional Requirements

- The program shall identify notes with %95 accuracy, including accidentals.
- The program shall determine subdivided notes with %95 accuracy.
- The program shall determine BPM and key with %90 accuracy.
- The program shall accept lossy compressed files such as MP3 and AAC; and lossless files such as FLAC and WAV. Other file types shall be disregarded.
- The program shall output as MIDI or PDF file.

2.2 Non-Functional Requirements

- The program shall have a clear UI.
- The program shall have %90 uptime per month.
- The program shall comply with local musical note naming convention.
- The program shall not use more than 2 GB of RAM.

2.2.1 Performance

The performance of the application heavily depends on the audio quality and the size of the audio file. User's internet connection stability and speed must also be considered. Processing and transforming the audio should not take more than 2-3 minutes provided the mentioned criteria meets certain standards.

2.2.2 Usability

The graphical user interface (GUI) is to be designed minimalistic in order to keep the user experience non-complex. The instructions and limitations on how to use the application efficiently is to be displayed on the interface.

2.2.3 Reliability

The downtime of the application user interface should be limited to a maximum of 5 hours a week. In case of a crash in the application, it must be repaired in at most an hour. The system shall display appropriate messages to the user while the system is down.

2.2.4 Supportability

The application is to be cost-effective to maintain and the maintenance of the system shall be done regularly. The application must be suitable for the addition of new features and the updating of the current features.

2.2.5 Scalability

The scalability of the system will be taken into consideration according to the changing needs of the users and the industry. While scaling up the application in terms of the conditions mentioned in hardware and software constraints, the performance of the system shall be maintained if not improved.

2.2.6 Extensibility

The application is to be designed considering SOLID principles in software engineering, namely open-closed principle. Therefore the system will be open to extensions such as the addition of new features according to the needs and suggestions of the users without changing the majority of the foundation of the code.

3. References

[1]: "ACM Code of Ethics and Professional Conduct", Association for Computing Machinery, 2018, https://www.acm.org/code-of-ethics

[2]: Ohio State University Department Of Computer Science and Engineering, Systems Requirements Specification (SRS), 2006
https://web.cse.ohio-state.edu/~bair.41/616/ReqDoc_Outline.htm