上海交通大学学位论文

Note-X: 实时音乐转录和编辑应用程序

姓　　名：贾逸凡, 祁令雨, 蘇文俊, 雷双羽, 黄越, 朱菁菁

学　　号：520370910121,520370910124,520370910011,
520021911089, 520370910002，520370910020

导　　师：皮宜博

学　　院：密西根学院

专业名称：电子与计算机工程

申请学位层次：学士

2024 年 08 月

A Dissertation Submitted to

Shanghai Jiao Tong University for Bachelor's Degree

NOTE-X：REAL-TIME MUSIC TRANSCRIPTION AND EDITING APP

Author: Jia Yifan, Qi Lingyu, SOUZA MARCO AUGUSTO DE,

Lei Shuangyu, Huang Yue, Zhu Jingjing

Supervisor: Pi Yibo

UM-SJTU JOINT INSTITUTE

Shanghai Jiao Tong University

Shanghai, P.R. China

June, 2024

上海交通大学

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全知晓本声明的法律后果由本人承担。

学 位 论 文 作 者 签 名：

LingyuQi,MarcoSouza

日期：2024 年 8 月 8 日

上海交通大学

学位论文使用授权书

本人同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。

本学位论文属于：

□**公开论文**

□**内部论文**，保密□ 1 年/□ 2 年/□ 3 年，过保密期后适用本授权书。

□**秘密论文**，保密____ 年（不超过 10 年），过保密期后适用本授权书。

□**机密论文**，保密____ 年（不超过 20 年），过保密期后适用本授权书。

（请在以上方框内选择打 “✓”）

学 位 论 文 作 者 签 名:

LingyuQi,MarcoSouza

日期: 2024 年 8 月 8 日

指 导 教 师 签 名:

LingyuQi,MarcoSouza

日期: 2024 年 8 月 8 日

摘 要

本文介绍了一个创新的音乐转录软件 Note-X 的设计与开发。该应用程序旨在解决音乐家、教育工作者和爱好者在转录过程中面临的几个挑战，包括模糊的音乐线条、复杂的和弦记录、频繁的速度变化和合唱中的声音区分。通过广泛的用户访谈和市场分析，我们确定了有效转录工具所需的基本功能，强调需要实时编辑功能和支持多种文件格式。该应用程序提供了用户友好的界面，简洁的编辑，准确的音乐转录和全面的输出选项，将其定位为现有产品 (如 Melodyne, Sibelius 和 Musescore) 的卓越替代品。此外，我们还针对特定任务进行了可用性测试，以评估关键功能的易用性和效率，例如文件上传、转录编辑和历史音频管理。结果表明，Note-X 满足了用户的关键需求，为音乐转录和编辑提供了直观而强大的工具。

关键词：音乐转录，实时编辑，音频处理，用户界面设计，多种格式输出

ABSTRACT

This paper presents the design and development of an innovative music transcription software, named Note-X. The application aims to address several challenges faced by musicians, educators, and enthusiasts in the transcription process, including obscured musical lines, complex chord recording, frequent speed changes, and differentiating voices in choruses. Through extensive user interviews and market analysis, we identified the essential features required for an effective transcription tool, emphasizing the need for real-time editing capabilities and support for multiple file formats. The app offers a user-friendly interface, concise editing, accurate music transcription, and comprehensive export options, positioning it as a superior alternative to existing products like Melodyne, Sibelius, and Musescore. Additionally, we conducted usability testing with specific tasks designed to evaluate the ease of use and efficiency of key functions, such as file upload, transcription editing, and history audio management. The results demonstrate that Note-X meets the critical needs of users, providing an intuitive and powerful tool for music transcription and editing.

Key words: Music Transcription, Real-Time Editing, Audio Processing, User Interface Design, Multi-Format Export

Contents

Chapter 1 Introduction

1.1 Foreword

In the digital age, the fields of music creation and education have undergone significant transformations, largely driven by the proliferation of smart devices and software applications. These advancements have provided musicians and students with various tools to simplify and enhance their creative processes. However, one of the enduring challenges in these fields is the manual transcription of music. Transcribing music involves translating audio or improvisational performances into standard musical notation, a task that demands extensive knowledge of music theory, a high level of skill, and considerable time and effort. This process can be daunting and error-prone, particularly for those who lack formal training in music notation.

The emergence of artificial intelligence (AI) and machine learning technologies has revolutionized several domains, including speech recognition, image analysis, and natural language processing. Deep learning, a subset of machine learning, has proven particularly effective in tasks requiring pattern recognition and data analysis. Given these technological advancements, there is a compelling opportunity to apply AI to the music domain, specifically in automating the transcription of audio into musical notation. Such automation could potentially alleviate the burden of manual transcription, making it more accessible and efficient for a broader range of users, including amateur musicians, students, and professional composers.

This paper presents an innovative Android application that integrates AI technology with a user-friendly interface to address the challenges of music transcription and editing. The application offers two primary features: automatic recognition of music audio and conversion into sheet music, and an intuitive interface for users to edit the resulting sheet music using a doodling method. The motivation behind this project stems from a desire to streamline the music creation process and provide users with a practical tool that can simplify one of the most time-consuming aspects of music composition.

To expand on the introduction with a specific focus on the motivation behind implementing the doodling feature for music notation editing, here's an additional paragraph:

Another unique feature of our application is the ability for users to edit sheet music through a doodling interface. The motivation behind this feature stems from the recognition that traditional music notation software often demands a steep learning curve, particularly for those unfamiliar with complex notation systems. For many musicians and students, the rigidity of conventional digital notation tools can stifle creativity and limit their ability to quickly capture musical ideas. The doodling interface addresses this issue by offering a more intuitive and flexible way for users to interact with sheet music. It allows users to freely draw musical symbols, annotations, and other marks directly onto the score, much like writing on a physical sheet of paper. This approach not only makes the editing process more accessible but also accommodates the nuances and personal styles of individual musicians. By integrating a doodling method, we aim to provide a seamless and enjoyable user experience that encourages spontaneous musical expression, making the application appealing to a broader audience, including beginners, hobbyists, and professionals alike. This feature underscores our commitment to enhancing the user experience by prioritizing ease of use and creative freedom.

The development of this application involved several key technical challenges and research questions. Firstly, we sought to enhance the accuracy and efficiency of converting music audio into sheet music using AI models. This task required the careful selection of deep learning models capable of handling the nuances of musical data. Secondly, we aimed to design a user-friendly interface that would allow users to easily interact with the application, draw musical symbols, and edit the automatically generated sheet music. This aspect of the project necessitated a deep understanding of user experience (UX) design principles and the specific needs of our target audience. Lastly, we had to ensure the security and privacy of user data, particularly in relation to the storage and management of musical works.

In this paper, we detail the design and implementation of the Android application, including the selection of AI techniques and the development of the front-end and back-end components. We also discuss the iterative design process, which involved extensive user testing and feedback collection to refine the application's features and improve its usability. Furthermore, we explore the challenges encountered during the development process, such as the integration of AI-generated outputs with user-edited content, and the technical limitations that influenced our design decisions.

The significance of this project extends beyond its practical applications. By automating the transcription process and offering a flexible editing interface, our application has the potential to democratize music creation, making it more accessible to individuals with varying levels of expertise. Additionally, our work contributes to the growing body of research on AI applications in the arts, providing insights and methodologies that could be applied to other creative domains. Through this project, we aim to highlight the possibilities of combining AI with creative technologies, fostering new approaches to artistic expression and education.

In the following sections, we provide a comprehensive overview of the related work in the field, outline our methodology, present the results of our experiments and user studies, and discuss the implications and future directions of our research. We believe that this application not only addresses a practical need in the music industry but also serves as a case study in the effective integration of AI technologies into mobile applications.

1.2 Literature Review

In recent years, the intersection of music technology and information technology has led to the development of various applications aimed at simplifying the music creation process. Several software solutions have been introduced to assist musicians and composers in transcribing audio into sheet music and editing musical scores. This section reviews some of the prominent applications in this domain, compares their features, and highlights the unique aspects of our proposed application.

1.2.1 MuseScore

MuseScore is a widely-used open-source music notation software that offers comprehensive tools for creating and editing sheet music. It supports a range of musical symbols and notations, and its user-friendly interface allows for precise placement of notes. MuseScore also provides a robust community for sharing and downloading scores. However, its focus on traditional score editing may pose a steep learning curve for users unfamiliar with music theory. Additionally, while MuseScore allows for MIDI input and playback, it lacks built-in features for converting audio recordings directly into sheet music.[1]

Illustration 1–1 Musescore app

1.2.2 Notion

Notion, developed by PreSonus, is a powerful notation and composition software that integrates traditional notation with playback and recording capabilities. Notion offers advanced features such as virtual instruments and real-time recording, making it a versatile tool for composers. However, similar to MuseScore, Notion's interface is geared towards users with a solid understanding of music theory. The software supports handwritten input through stylus devices, but it does not offer a freeform doodling interface for less structured edits.[2]

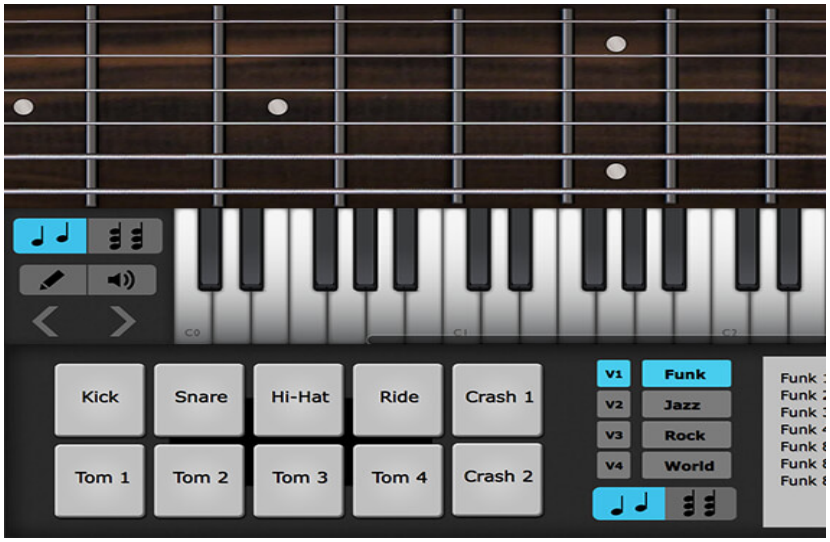

Illustration 1–2 Notion app

1.2.3 ScoreCloud

ScoreCloud provides an automatic music notation solution, converting audio recordings and MIDI files into sheet music. Its cloud-based platform enables users to access and edit their scores from any device. ScoreCloud's primary strength lies in its audio-to-sheet music conversion capabilities, using AI to analyze and transcribe musical elements. However, ScoreCloud's transcription accuracy can vary based on the complexity of the audio input, and its editing tools are relatively basic compared to more comprehensive notation software.[3]

Illustration 1–3 ScoreCloud app

1.3 Our Proposed Application'S significance

Our Android application distinguishes itself from the aforementioned software by integrating a doodling-based editing interface with an AI-powered audio-to-sheet music conversion system. Unlike traditional notation software, our doodling interface allows users to freely draw musical symbols and annotations directly onto the score, providing a more accessible and creative way to edit music. This feature is particularly beneficial for users who may not be familiar with formal music notation, offering them a more intuitive and flexible

approach to score editing.

The AI-powered conversion feature of our application leverages deep learning models to transcribe audio recordings into sheet music. This functionality aims to provide a high degree of accuracy and efficiency, streamlining the transcription process for users. Unlike ScoreCloud, which primarily focuses on cloud-based editing, our application emphasizes local data storage, ensuring user privacy and data security. Furthermore, the integration of these features within a single mobile application enhances portability and convenience, allowing users to create and edit music on the go.

In summary, while existing music notation software offers a range of powerful tools and features, our application addresses specific user needs through its unique combination of a doodling interface and AI-based transcription. By lowering the barriers to entry for music notation and providing a versatile toolset, our application seeks to democratize music creation and cater to a broader audience, from casual hobbyists to professional musicians.

1.4 Teamwork

In our software development project, the importance of clear communication, shared expectations, and mutual accountability cannot be overstated. To ensure a cohesive and productive working environment, our team established a comprehensive Team Commitment Contract at the outset of the project. As detailed in the appendix titled 1 (see Appendix 1), our team has established norms and expectations to ensure smooth collaboration throughout the project. This contract serves as a guiding document that outlines the norms, expectations, and protocols we adhere to throughout the development process. The contract is designed to foster collaboration, maintain professionalism, and support the successful completion of our project.

The Team Commitment Contract covers several critical aspects of team dynamics and project management. First, it addresses team member work patterns, acknowledging the diverse preferences and schedules of individual members. For instance, some team members prefer to work late at night, while others are early risers. To accommodate these differences, we established flexible work times, with an emphasis on working ahead when possible.

The use of communication channels was another key focus, where we agreed to respond to emails and messages within four business hours. Our communication style is casual but

respectful, with clear markers for urgent matters. This approach ensures efficient and effective communication without sacrificing professionalism.

We also established a routine for work-together time, setting aside at least four hours per week for all hands meetings, in addition to individual work. Punctuality is emphasized, with a tolerance of within 10 minutes of the scheduled time. Meetings are conducted online, with cancellations requiring at least 24 hours notice.

Decision-making within the team is collaborative, with no single member holding unilateral authority. This democratic approach allows for diverse input and shared responsibility. We have yet to appoint a permanent project manager, instead opting for a flexible arrangement where leadership responsibilities may rotate or be shared as needed.

Our contract also outlines protocols for project publication, opting to publish our final product on GitHub for open access. In line with agile methodologies, we adopted a ”fail fast, fail early, fail often” philosophy, encouraging prompt communication when encountering difficulties, with a 48-hour window for reporting issues.

Effort expectations were candidly discussed, though we have not yet reached a consensus on the desired grade outcome. We addressed potential conflicts and agreed on a process for resolving them, starting with direct discussion and, if necessary, escalating to the instructor. The contract specifies grounds for potential dismissal, including failure to complete tasks, respond to communications, attend meetings, or maintain professional behavior.

This Team Commitment Contract reflects our commitment to transparency, accountability, and mutual respect. It establishes a framework for our interactions and decision-making processes, ensuring that all team members are aligned in their goals and responsibilities. As we proceed with our project, this contract will serve as a crucial reference point, helping us navigate challenges and achieve our objectives with a unified approach.

Chapter 2 Design Specifications

2.1 User Interview and Questionnaire Overview

To gain a comprehensive understanding of the needs and challenges faced by users in the field of music transcription, we conducted a series of user interviews and distributed a questionnaire. The participants in this study included a diverse group of individuals, comprising students, teachers, hobbyists, and professionals within the music industry.

2.1.1 Participants

- **Students:** Individuals currently enrolled in music-related academic programs.
- **Teachers:** Educators who specialize in music instruction.
- **Hobbyists:** Enthusiasts who engage in music as a personal interest or pastime.
- **Professionals:** Experienced musicians, composers, and other industry professionals.

2.1.2 Venues

The interviews and questionnaires were conducted in a variety of settings to ensure a broad range of perspectives:

- **Music Studio:** Professional environments where music production and practice occur.
- **Online Meeting:** Virtual platforms to facilitate remote discussions and surveys.
- **Classroom:** Educational settings where students and teachers interact.

The detailed results of the questionnaire, including the specific questions asked and responses collected, can be found in the Appendix 2of this report. This comprehensive data provides valuable insights into user preferences and pain points, which have informed the development and enhancement of our music transcription software.

2.2 Pain Points and Challenges in Music Transcription

Through our questionnaire and interviews with music professionals and dedicated enthusiasts, several critical challenges in music transcription have emerged. These challenges highlight the complexities involved in accurately transcribing music and underscore the need

for advanced tools to assist in this process.

2.2.1 Obscured Musical Lines

One of the primary difficulties faced by transcribers is dealing with obscured musical lines in mixed audio. When multiple instruments or voices are present, distinguishing individual musical lines becomes challenging. This issue is particularly pronounced in densely arranged compositions or poorly recorded tracks where certain musical elements may be masked or indistinct.

2.2.2 Complex Chord Recording

Accurately recording and notating complex chords is another significant challenge. Musicians often encounter intricate chord structures that are difficult to transcribe manually. These complex chords can include extended harmonies and altered tones that require precise identification and notation. The manual transcription process may not capture these nuances effectively, leading to incomplete or inaccurate chord representations.

2.2.3 Frequent Speed Changes

Music that features frequent tempo changes or fluctuating speeds adds another layer of complexity to transcription. Variations in tempo can disrupt the consistency of the transcription process, making it harder to maintain accurate timing and rhythm throughout the piece. This challenge is particularly relevant in pieces with rubato passages or dynamic tempo alterations.

2.2.4 Differentiating Voices in Choruses

Distinguishing between multiple voices or harmonies in choral pieces presents a considerable challenge. In choruses where voices overlap and blend, transcribers must identify and separate each voice to create an accurate transcription. This task becomes more difficult when voices have similar timbres or when there are multiple layers of harmony.

2.2.5 Additional Challenges

In addition to the key challenges, our research identified other issues affecting transcription accuracy:

- **Hard to Identify Phrases in Mixed Audio:** Extracting specific musical phrases from a mix where different elements are layered together can be challenging.
- **Hearing Chords with Poor Pitch Perception:** Transcribers with less accurate pitch perception may struggle to identify chords correctly.
- **Quick Notation Needs:** Musicians often require tools that facilitate rapid notation of musical ideas and chord structures.
- **Distinguishing Similar-Sounding Chords in Vocals:** Identifying chords that sound similar, especially in vocal arrangements, can be difficult.

Addressing these challenges requires a sophisticated approach to music transcription, incorporating advanced technologies and intuitive tools to enhance accuracy and efficiency. Our application aims to tackle these issues by leveraging AI and information technology to provide a more reliable and user-friendly transcription experience.

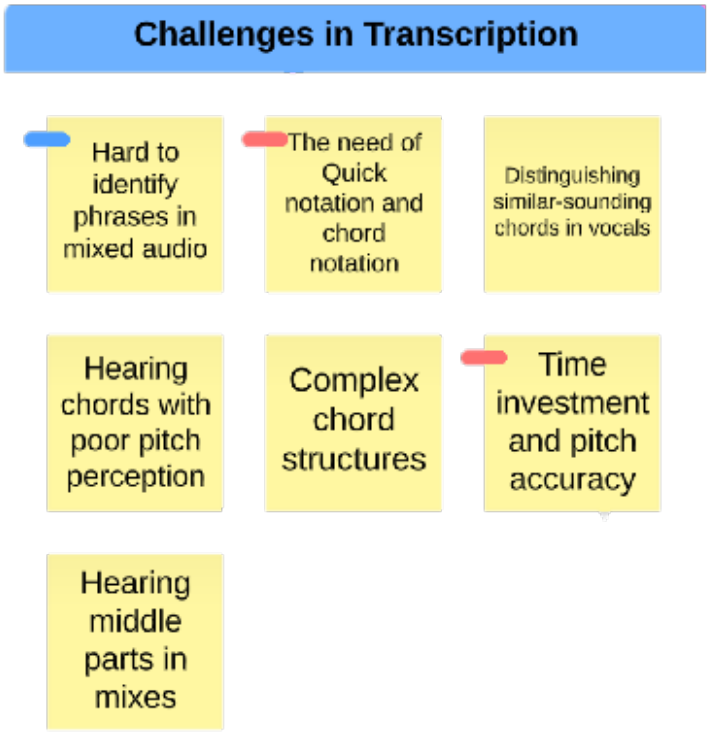

Illustration 2–1 Transcription Challenges

2.3 Essential Features for Auto-Transcription Software

In our research on essential features for auto-transcription software, we gathered insights from various composers and musicians. The following key features were identified as crucial for an effective and efficient auto-transcription tool:

2.3.1 Key Features Identified

- **Composer's Perspective Output:** Ensuring the output aligns with the composer's intent, providing a user-friendly and intuitive representation.
- **Accurate Beat Detection and Readable Sheet Music:** Precision in identifying beats and producing easily readable sheet music.
- **Instrument Recognition:** The ability to distinguish and correctly identify different instruments in a composition.
- **Chord Quality Recognition:** Accurately identifying and notating chord qualities.
- **Ensuring Accuracy:** Maintaining a high level of transcription accuracy, minimizing errors and discrepancies.
- **Differentiating Similar-Sounding Lines:** The capability to distinguish between similar-sounding musical lines, especially in complex arrangements.
- **Real-Time Editing:** Allowing users to correct and annotate transcriptions in real-time, directly addressing any inaccuracies as they occur.
- **Correct Tempo, Measure Numbers, and Chords Recognition:** Ensuring accurate detection and representation of tempo, measure numbers, and chords.
- **Voice and Accompaniment Separation:** Differentiating between the main voice and accompanying elements in a piece.

2.3.2 Key Challenge and Solution

Among the identified features, the lack of real-time editing capabilities emerged as the most critical issue faced by users. The absence of real-time editing hinders users' ability to promptly correct and annotate their transcriptions, potentially leading to inaccuracies and a less efficient workflow. Our research indicated a strong preference for a feature that allows users to correct and annotate transcriptions on the go.

To address this need, our application has been designed to incorporate robust real-time

editing capabilities. Users can now make adjustments and annotations directly within the transcription interface, ensuring that any discrepancies can be addressed immediately. This feature not only improves the accuracy of the final transcription but also streamlines the overall process, making it more efficient and user-friendly.

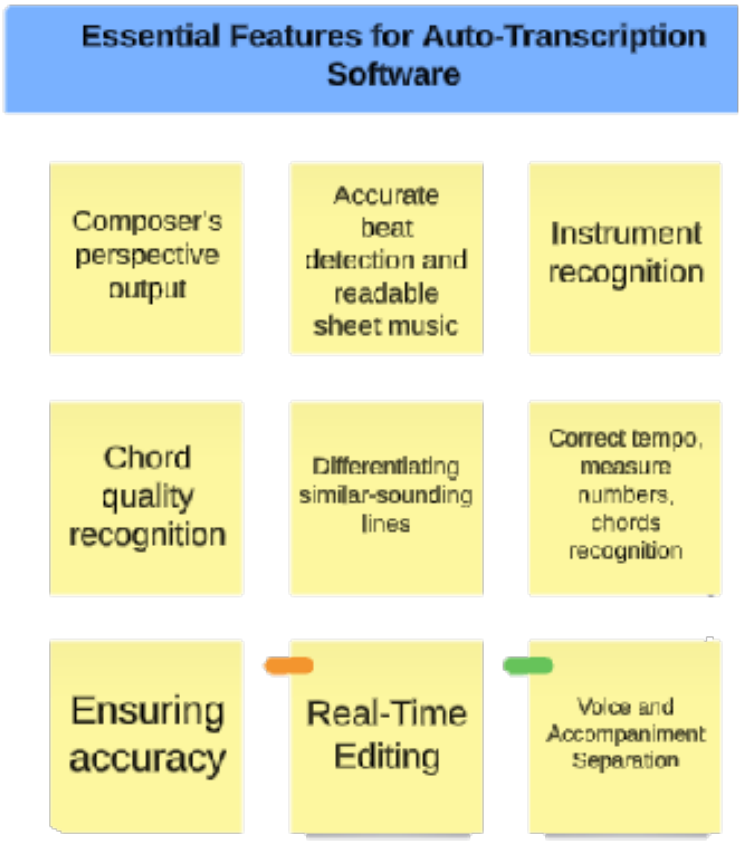

Illustration 2–2 Essential Features

2.4 Market Analysis of Music Editing Software

In our market analysis, we investigated various music editing software tools to understand their strengths and limitations. The following observations summarize the key challenges identified across different platforms:

2.4.1 Key Observations

- **Sibelius:** Lacks quantization features, which limits its capability to align notes to a strict rhythmic grid, making it less ideal for precise rhythmic editing.
- **Guitar Pro 7:** The note-setting process is cumbersome, resulting in a less efficient workflow for users.
- **Sibelius:** Does not adequately support symbols for electronic instruments, restricting its use for modern electronic music compositions.
- **Musescore, Melodyne:** Lack of comprehensive transcription support, making it challenging to convert audio recordings into accurate sheet music.
- **Cubase, Dorico:** MIDI import issues, which can complicate the process of integrating MIDI data into the software.
- **Musescore:** Limited formatting options, restricting users' ability to customize the visual layout of their scores.

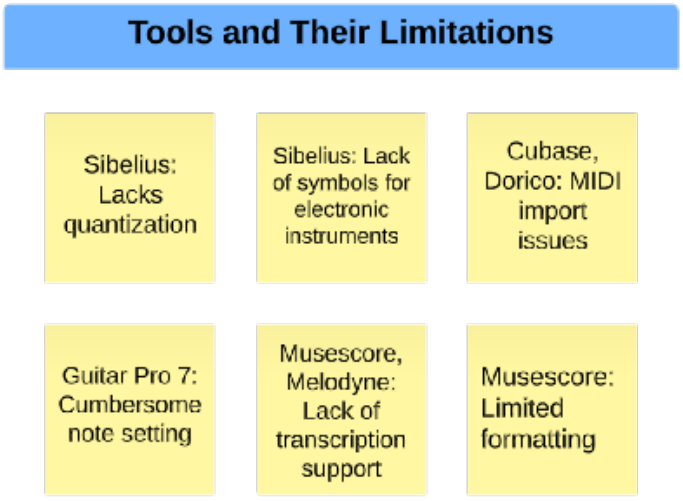

Illustration 2–3 Existing Tools and Limitations

2.4.2 User Preferences for Output Formats

In these software-specific issues, we found import and output formats are serious problems. Our research also revealed a strong user preference for software that supports a variety of output formats. Users expressed a need for flexible and diverse output options to accommodate different use cases. The desired output formats include:

- **PDF:** For easy sharing and printing of sheet music.
- **MP3:** For audio playback and distribution.
- **MusicXML:** For interchangeability between different music notation software.
- **MP4:** For video presentations, including synchronized audio and visual elements.
- **MIDI:** For use in digital audio workstations (DAWs) and electronic instruments.
- **.sib:** For compatibility with Sibelius software.

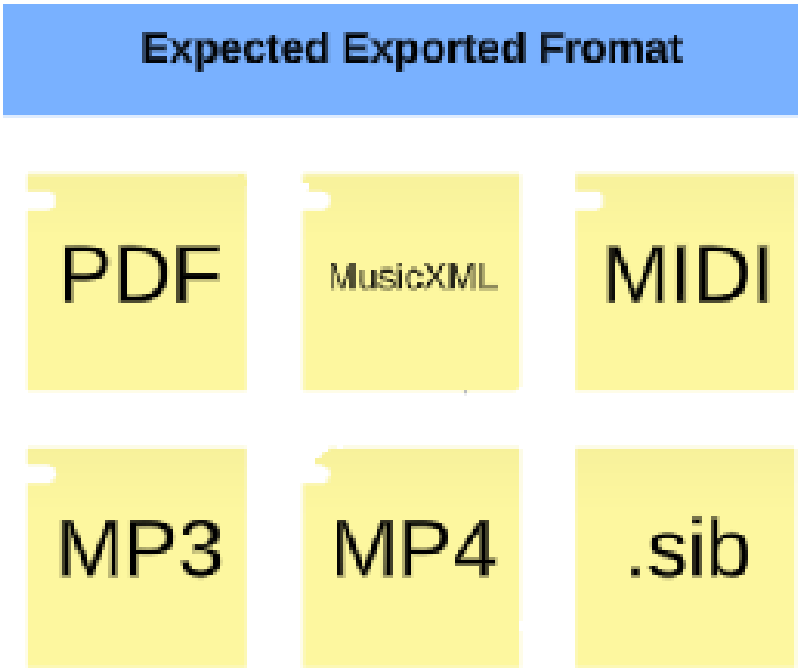

Illustration 2–4 User expectations for formats

2.4.3 Multi-Format Import/Export Problem

A significant challenge identified during our research was the problem with multi-format import/export capabilities. Many existing software tools offer limited support for various input and output formats, which can be a major drawback for users who need flexibility in handling different file types.

User Preference: The demand for software that supports various input and output formats is evident. Users expressed a strong preference for tools that enable seamless transitions between different formats, allowing them to work more efficiently and effectively. This includes the ability to import and export files in widely used formats like PDF, MP3, Mu-

sicXML, MP4, MIDI, and .sib. Providing comprehensive support for these formats can significantly enhance the usability and versatility of the software, catering to the diverse needs of musicians, composers, and educators.

Our application aims to address these issues by offering robust multi-format import/-export capabilities, ensuring that users can easily manage and convert their work between different formats without hassle. This feature not only enhances the user experience but also broadens the application's applicability across various musical contexts.

2.5 Competitor Analysis

We conducted a competitor analysis to evaluate how our product, Note-X, compares with existing solutions in the market. The analysis covers several key features, including automatic audio transcription, online editing capabilities, user-friendliness, support for multiple export formats, vocal separation, and cost.

Product	Automatic Audio Transcription	Online Editing	User-Friendly Interface	Multiple Export Formats	Vocal Separation	Cost
Melodyne	No	No	Low	Medium	High	High
Noteflight	Low	High	High	Medium	Low	Low
Sibelius	No	No	Low	High	Low	High
Musescore	No	Yes	High	High	Low	Free
SourceCloud	Medium	Medium	High	Medium	Medium	Low
Finale	No	No	Low	High	Low	High
Our product: Note-X	**High**	**Medium**	**High**	**High**	**High**	**Free**

Illustration 2–5 Competitor analysis

2.5.1 Analysis

Our analysis reveals that Note-X offers high performance across all key features compared to other products. Notably, Note-X excels in automatic audio transcription, user-friendly

interface, and vocal separation, while maintaining a competitive cost of being free. This comprehensive feature set positions Note-X as a strong contender in the market, addressing the common limitations observed in existing products.

2.6 Story Map

The story map serves as a crucial component in the development and user experience design of the Note-X app. It provides a detailed visual representation of the user's journey through the application, highlighting key features and interactions. This map is essential for several reasons:

Firstly, it helps to clarify the overall structure and flow of the application, ensuring that all team members have a shared understanding of the user's path. By outlining each stage, from initial sign-in to the final sharing of sheet music, the story map ensures that every aspect of the user experience is carefully considered and designed.

Secondly, the story map identifies and delineates the core functionalities and processes within the app. It illustrates how different components, such as audio input, preprocessing, pitch detection, and sheet music editing, are interconnected and how they contribute to the overall user experience. This holistic view is invaluable for developers, designers, and stakeholders, as it facilitates coordinated efforts and highlights areas that may require additional attention or improvement.

Moreover, the story map plays a significant role in enhancing user-centric design. By visualizing the user's journey, it becomes easier to identify potential pain points and opportunities for enhancing usability and satisfaction. This ensures that the app not only meets functional requirements but also provides a seamless and intuitive experience for the user.

In essence, the story map is a foundational tool in the development process of Note-X. It provides a comprehensive overview of the user experience, supports collaborative development efforts, and ensures that the app delivers a high-quality, user-friendly experience. The following sections will delve into the specific stages and processes outlined in the story map, providing detailed insights into the user's interaction with the app.

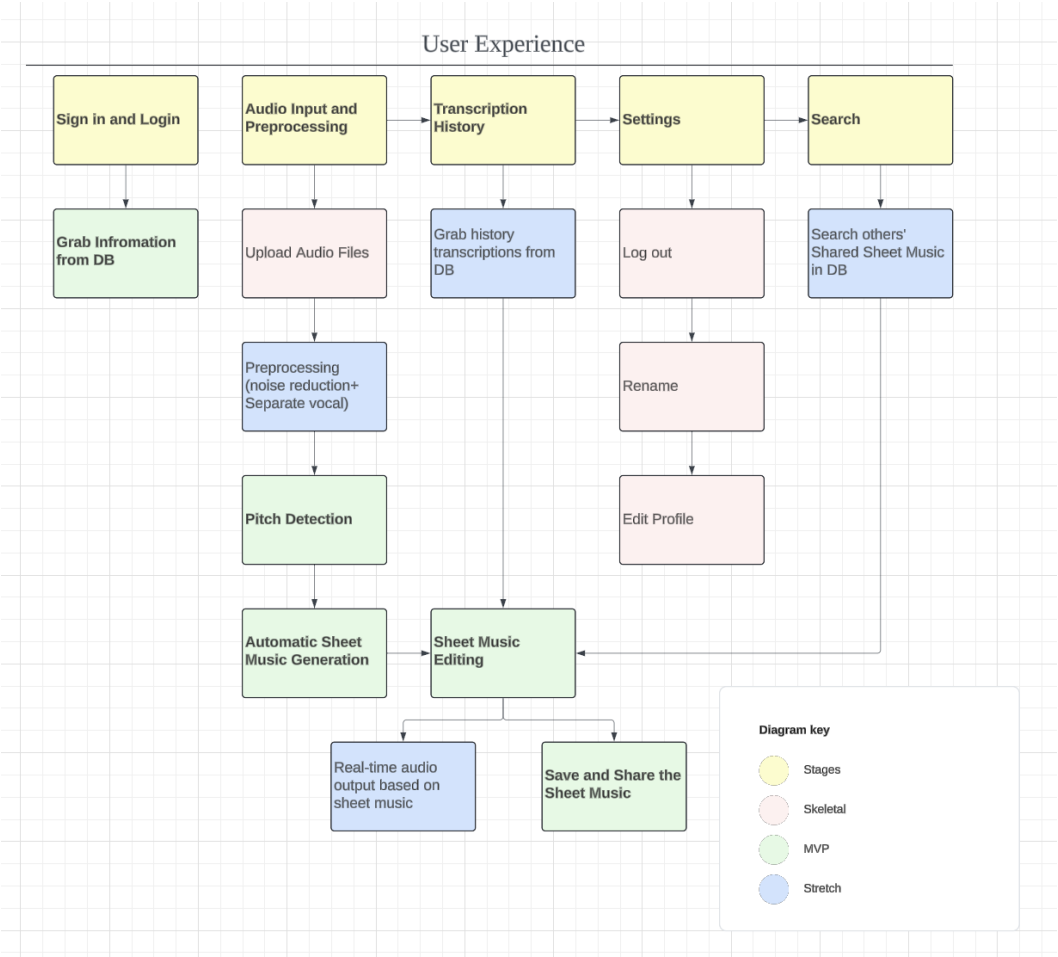

Illustration 2–6 Competitor analysis

2.6.1 Sign in and Login

The user experience begins with the ”Sign in and Login” stage, where users authenticate themselves to access the app. Information is retrieved from the database (DB) to verify credentials and load user-specific data. This initial step ensures that each user has a personalized experience, with access to their previous transcriptions and settings. The process is streamlined to minimize any barriers to entry, ensuring that users can quickly and efficiently begin using the app.

2.6.2 Audio Input and Preprocessing

After logging in, users proceed to the ”Audio Input and Preprocessing” stage. Here, users upload audio files that they want to transcribe into sheet music. The app supports various audio formats to ensure compatibility with different recording devices. The preprocessing step involves noise reduction to enhance the clarity of the audio and separation of the vocal track from other instruments. This step is crucial because it prepares the audio data for accurate pitch detection by isolating the primary melody from background noises and harmonics. The use of advanced algorithms in noise reduction ensures that even recordings made in less-than-ideal conditions can be processed effectively.

2.6.3 Pitch Detection and Sheet Music Generation

The next critical step is ”Pitch Detection,” where the app analyzes the preprocessed audio to identify the musical notes. This involves converting the audio signal into a spectrogram and using machine learning algorithms to detect the fundamental frequencies. This process is followed by ”Automatic Sheet Music Generation,” where the detected pitches are converted into musical notation, forming the initial draft of the sheet music. The app uses sophisticated algorithms to ensure that the generated sheet music accurately reflects the timing, dynamics, and articulation of the original performance. The precision of these algorithms is paramount, as it directly impacts the quality and usability of the transcribed music.

2.6.4 Sheet Music Editing

Users can then move to the ”Sheet Music Editing” stage. In this phase, they can make modifications to the automatically generated sheet music, correcting any errors and making

adjustments to better reflect the intended musical piece. The editing tools are designed to be intuitive, allowing users to easily add, remove, or alter notes and symbols. This stage is critical for ensuring the final transcription is accurate and meets the user's expectations. The user-friendly interface provides a range of editing options, from basic note adjustments to more complex changes involving dynamics and articulations, ensuring that users can achieve a high level of precision in their transcriptions.

2.6.5 Real-time Audio Output

One of the advanced features of our app is the ability to provide "Real-time Audio Output" based on the sheet music. This feature allows users to hear the transcription and make further refinements, ensuring the accuracy and quality of the transcription. By listening to the playback, users can identify any discrepancies between the original performance and the transcription, making it easier to fine-tune the sheet music. This feedback loop is essential for musicians who rely on auditory confirmation to validate the accuracy of the transcribed notes, ensuring that the final output is both musically and technically sound.

2.6.6 Save and Share the Sheet Music

Once satisfied with the edits, users can "Save and Share the Sheet Music." This functionality allows them to store their transcriptions in the database and share them with others. Users can also search for and view sheet music shared by other users, fostering a collaborative environment. This feature is particularly useful for music educators and students, as it facilitates the sharing of practice materials and collaborative projects. The ability to share and collaborate on sheet music within the app enhances its utility and encourages community engagement among musicians.

2.6.7 Transcription History

The app maintains a "Transcription History," enabling users to access and manage their past transcriptions. This feature provides an easy way to revisit and edit previous work, ensuring continuous improvement and refinement of their music transcriptions. Users can sort and filter their history based on various criteria, such as date, genre, or project, making it easy to find specific transcriptions. This archival capability is essential for users who work

on multiple projects simultaneously or need to reference previous transcriptions for ongoing work.

2.6.8 Settings

In the ”Settings” section, users can log out, rename their profiles, and edit other personal information. This stage ensures that users have control over their account settings and preferences. The settings section also includes options for customizing the app's interface, such as theme selection and notification preferences, to enhance the user experience. By providing a high degree of customization, the app ensures that it can meet the diverse needs and preferences of its user base, enhancing overall satisfaction and usability.

2.6.9 Search

Finally, the ”Search” functionality allows users to explore other shared sheet music available in the database. This feature enhances the collaborative aspect of the app, enabling users to learn from and contribute to the community of musicians using Note-X. Advanced search filters allow users to narrow down results based on specific criteria, such as instrument, difficulty level, or composer. This functionality not only supports users in finding relevant material quickly but also fosters a sense of community by encouraging the sharing and discovery of new musical works.

Chapter 3 Front end Design

3.1 UIUX Test and Analysis

The UIUX test was designed to evaluate various aspects of the app, focusing on ease of use and functionality of the user interface. Below is an analysis of each research question we set for the UIUX test. And next is the analysis of the Tasks and Evaluation Metrics we set for the UIUX test.

3.1.1 Research Questions

1. How easily can a user upload a music file?

- **Purpose:** To assess the intuitiveness of the file upload process, a fundamental feature of the app.
- **Rationale:** Users need to upload music files effortlessly to begin transcription. A complicated upload process could deter users from utilizing the app.
- **Significance:** Ensuring a smooth upload process is critical as it directly affects user satisfaction and the efficiency of initiating the transcription process.

2. How easily can a user save the transcription?

- **Purpose:** To evaluate how straightforward it is for users to save their work after transcription.
- **Rationale:** Users need a quick and easy way to save their progress. Complications in this area could lead to frustration and potential data loss.
- **Significance:** A simple saving mechanism is crucial to avoid user frustration and to enhance the reliability and efficiency of the app.

3. How easily can a user edit the transcription of vocal or instrumental parts, or both?

- **Purpose:** To investigate the app's editing capabilities, specifically regarding the differentiation between vocal and instrumental elements.
- **Rationale:** Users may need to adjust or correct transcriptions. The ability to easily switch and edit these components is essential for detailed work.

- **Significance:** This feature's usability directly impacts the app's value proposition, especially for users requiring detailed and accurate editing functionalities.

4. How easily can a user search for other users' transcriptions?

- **Purpose:** To assess the search functionality within the app.
- **Rationale:** Efficient search capabilities allow users to find specific transcriptions quickly, enhancing the app's collaborative and community aspects.
- **Significance:** A robust search function improves usability and fosters a sense of community by making it easy to access and share work.

5. How easily can a user deal with historical audio data?

- **Purpose:** To evaluate how users interact with previously uploaded or saved audio data.
- **Rationale:** Users may need to revisit and modify past work. An intuitive interface for managing historical data ensures continuity and ease of access.
- **Significance:** The ability to handle historical data seamlessly is crucial for users who work on multiple projects over time.

3.1.2 Tasks and metrics

Task 1.1: Find out what type of file can be accepted by the app.

- **Reason:** Ensures users know the app's capabilities and limitations regarding file compatibility.
- **Function:** Provides clarity and sets user expectations, preventing frustration from unsupported formats.
- **Evaluation Metric:** Task 1.1 should be completed in ≤ 2 clicks.

Task 1.2: Choose a file and upload it to the app.

- **Reason:** Tests the actual upload process and the app's responsiveness.
- **Function:** Verifies the ease of use and efficiency of initiating a new project.
- **Evaluation Metric:** Task 1.2 should be completed in ≤ 2 clicks.

Task 2.1: Follow the pop-up window to save the generated transcription.

- **Reason:** Ensures the saving process is clear and straightforward.

- **Function:** Confirms that users can easily save their work without confusion.
- **Evaluation Metric:** Task 2.1 should be completed in $\leq$ 5 seconds.

Task 3.1: Try to switch between transcriptions of vocal and instrument and both.

- **Reason:** Assesses the app's capability to differentiate and manage multiple audio components.
- **Function:** Ensures flexibility and control over the transcription editing process.
- **Evaluation Metric:** Task 3.1 should be completed in $\leq$ 5 seconds.

Task 3.2: Try to edit the transcription as you wish.

- **Reason:** Evaluates the comprehensiveness and intuitiveness of the editing tools.
- **Function:** Determines if the app provides adequate editing functionalities to meet user needs.
- **Evaluation Metric:** Task 3.2 should be completed in $\leq$ 1 minute, including discovering all provided functions.

Task 4.1: Try to find one transcription using the search window.

- **Reason:** Tests the ease of navigation and the effectiveness of the search function.
- **Function:** Ensures users can efficiently locate specific transcriptions, enhancing the app's utility.
- **Evaluation Metric:** Task 4.1 should be completed in $\leq$ 10 seconds to find the search window and understand its usage.

Task 5.1: Find out where to edit historical audio.

- **Reason:** Assesses the app's organization and user interface design.
- **Function:** Ensures that users can easily manage and access their previous work.
- **Evaluation Metric:** Task 5.1 should be completed in $\leq$ 5 seconds.

Task 5.2: Try to open past audio.

- **Reason:** Tests the app's ability to handle and display historical data.
- **Function:** Confirms the app's functionality in providing continuity and access to past projects.

- **Evaluation Metric:** Task 5.2 should be completed in ≤ 5 seconds.

3.1.3 Pre-Test and Post-Test Questionnaires

Pre-Test Questionnaire

- **1. Are you a music lover?**
  - **Reason:** Identifies participants' background and interest level in music, which may influence their interaction with the app.
  - **Function:** Helps in understanding the participants' familiarity with music and their expectations from the app.
- **2. Have you used transcriptions?**
  - **Reason:** Determines participants' experience with transcription tools, which could affect their usability of the app.
  - **Function:** Provides insight into the participants' prior experience and potential learning curve with the app.

Post-Test Questionnaire

- **1. Is it easy for you to manipulate the product?**
  - **Reason:** Assesses the ease of use and user-friendliness of the app based on the participants' experiences.
  - **Function:** Provides feedback on the app' s usability and identifies areas for improvement.
- **2. Do you feel it easy to get started?**
  - **Reason:** Evaluates how intuitive and user-friendly the app is for new users.
  - **Function:** Helps determine the effectiveness of the onboarding process and initial user experience.

3.2 Front End design of the APP

The front-end user interface (UI) of the app is crafted with a focus on simplicity, intuitiveness, and user experience. The design follows a clean and functional approach, ensuring that users can easily navigate through the app's features. The interface consists of several key views, each tailored to specific functionalities. The login view serves as the entry point

for users, providing seamless access to the app through secure login and registration options. Once logged in, users are greeted with the main view, which acts as a hub, offering quick access to primary features such as uploading audio, viewing history, editing, and adjusting settings. Each view is designed to be visually coherent, with consistent layout and design elements that enhance usability while maintaining a modern aesthetic. The upload view allows users to process audio files with options for selecting vocal or instrumental tracks. The history view provides a record of previous uploads, enabling users to edit and export their audio files. Finally, the settings view offers customization options, allowing users to personalize their profile and application settings. This UI design is aimed at delivering a smooth and engaging user experience, catering to both new and returning users.

3.2.1 Login View

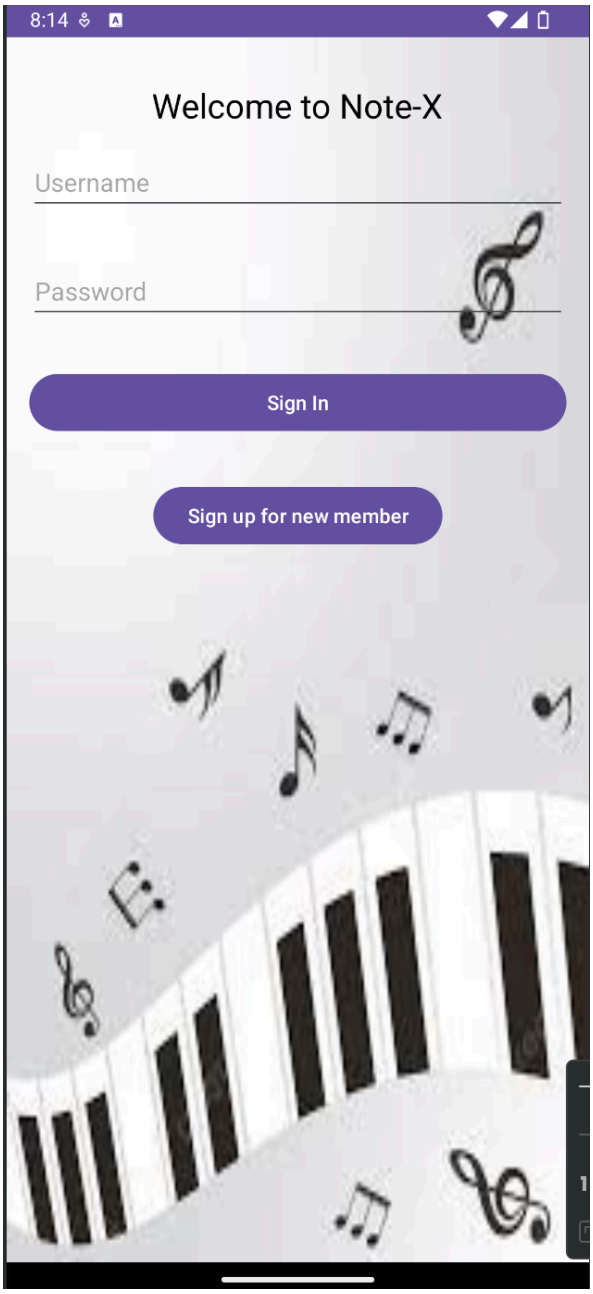

Illustration 3–1 Login View

The login view is the gateway to accessing the app's features, where users are prompted to enter their username and password to securely log in. For first-time users, the interface also provides a sign-up option, ensuring a seamless onboarding process. Upon pressing the

sign-up button, users are navigated to a dedicated registration screen, allowing them to create a new account effortlessly.

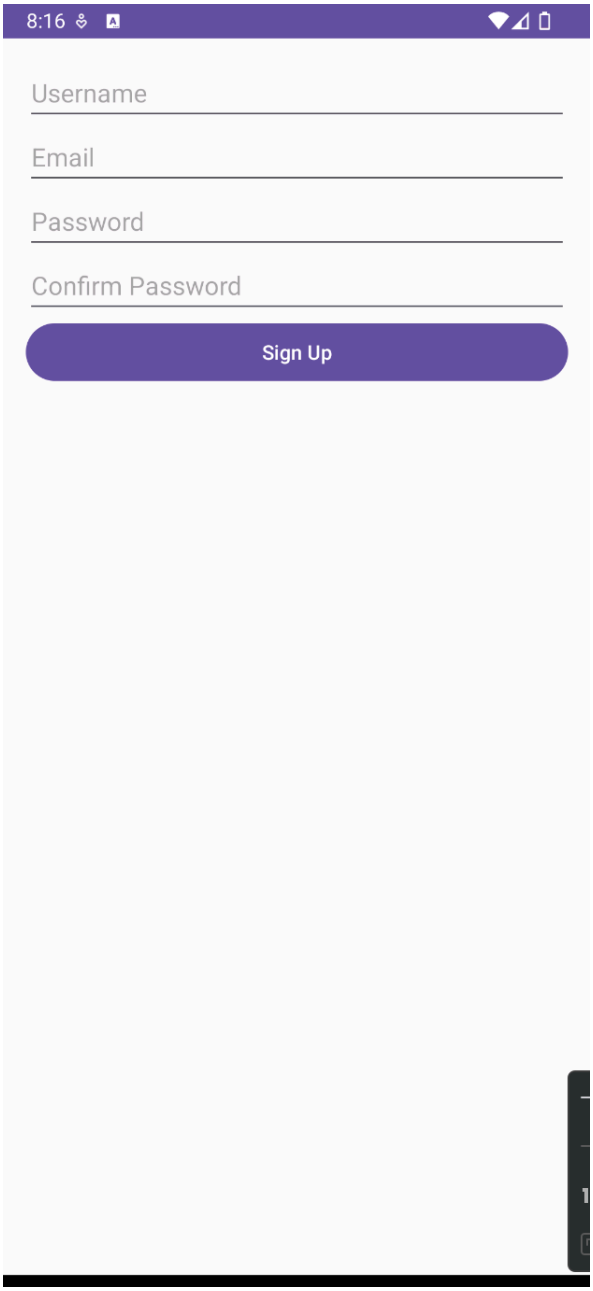

Illustration 3–2 Sign Up View

The sign-up view facilitates new user registration. It collects essential information such as username, password, and email to create a user account. The design focuses on simplicity and ease of use, guiding users through the process without unnecessary complexity.

3.2.2 Main View

The main view acts as the central hub of the app, providing users with quick access to key functionalities through four prominent buttons: Upload, History, Edit, and Settings. Each button is clearly labeled and designed for intuitive navigation. When a button is pressed, the user is seamlessly directed to the corresponding feature of the app, enabling efficient interaction with its core functions.

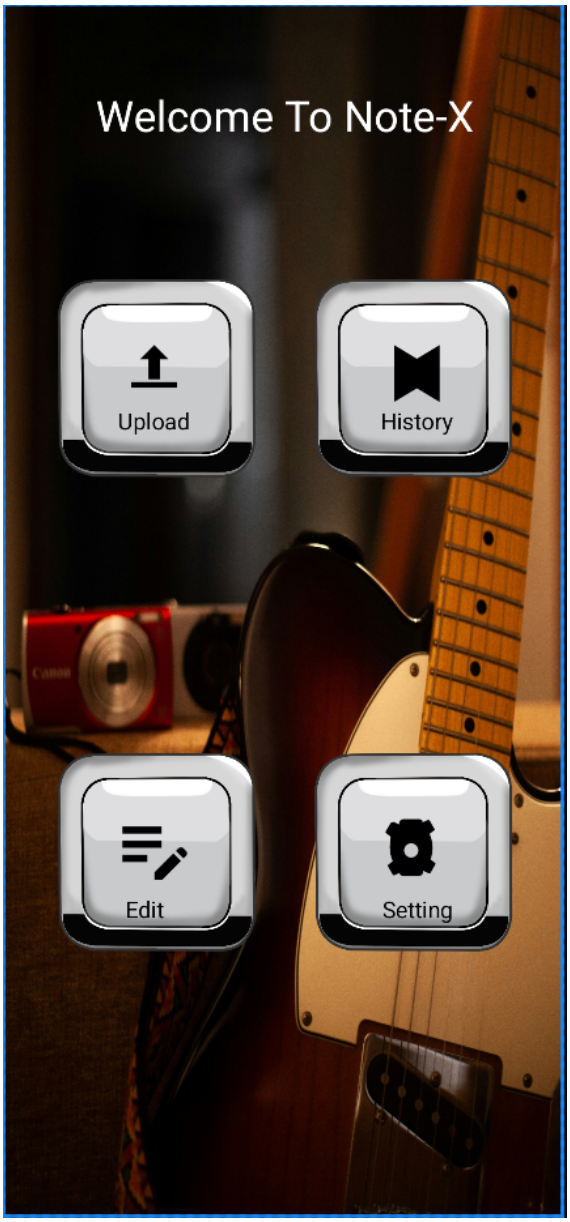

Illustration 3–3 Main View

3.2.3 Upload View

The upload view is designed to allow users to easily select an audio file for processing. Users can choose the audio they wish to process from their device. The interface also includes two switches that enable users to specify whether the audio is primarily vocal or instrumental. Once the desired options are selected, pressing the ”Start” button initiates the processing of the audio file.

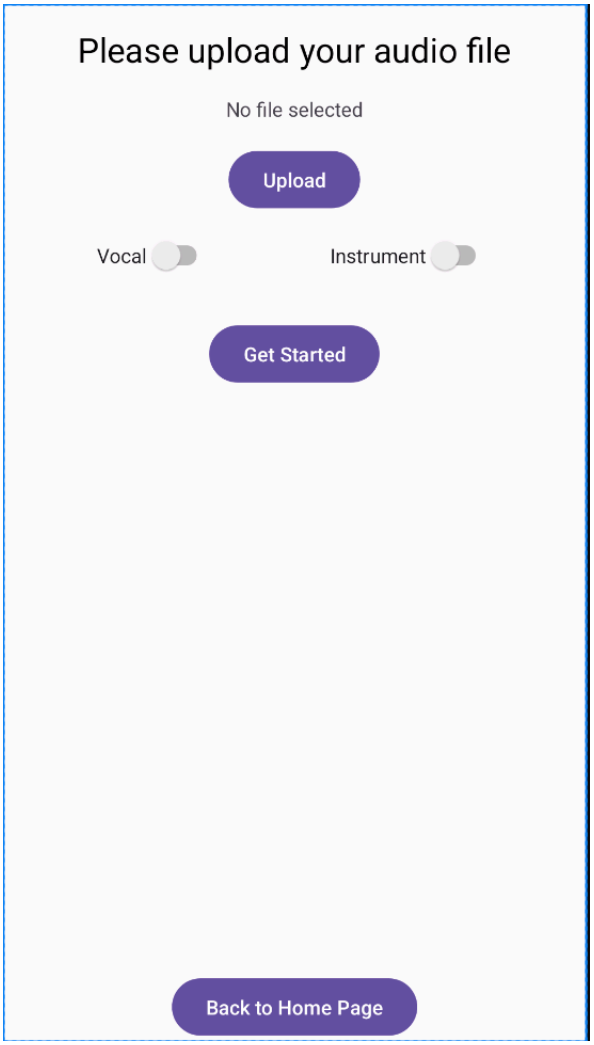

Illustration 3–4 Upload View

Once the processing is complete, the showing view automatically displays the generated musical notation. The interface is designed to be minimalistic and functional. The generated musical score is presented as a high-quality PNG image, allowing users to visually inspect

the transcription. There is alse a Save button. Users can save the displayed musical score in PNG or PDF format to their device, enabling easy sharing or printing.

This seamless integration between the upload and showing views ensures that users can efficiently process, view, and save their transcribed musical scores with minimal effort.

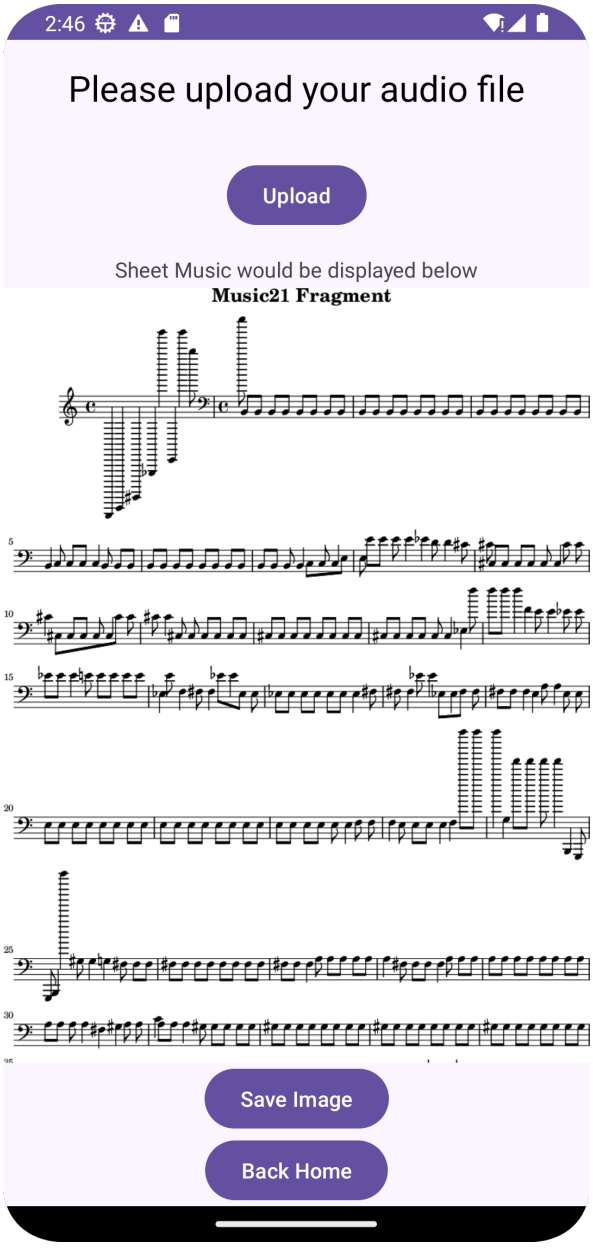

Illustration 3–5 Sheet Music

3.2.4 History View

The history view provides users with a comprehensive list of their previously uploaded audio files. This view allows users to revisit their past uploads, and each audio file can be further edited or exported. By pressing the corresponding buttons, users can either modify the audio or save it to their device.

Illustration 3–6 History View

3.2.5 Setting View

The settings view offers users the ability to personalize and configure the app according to their preferences. Within this view, users can change their profile picture and update their username, allowing for a tailored user experience.

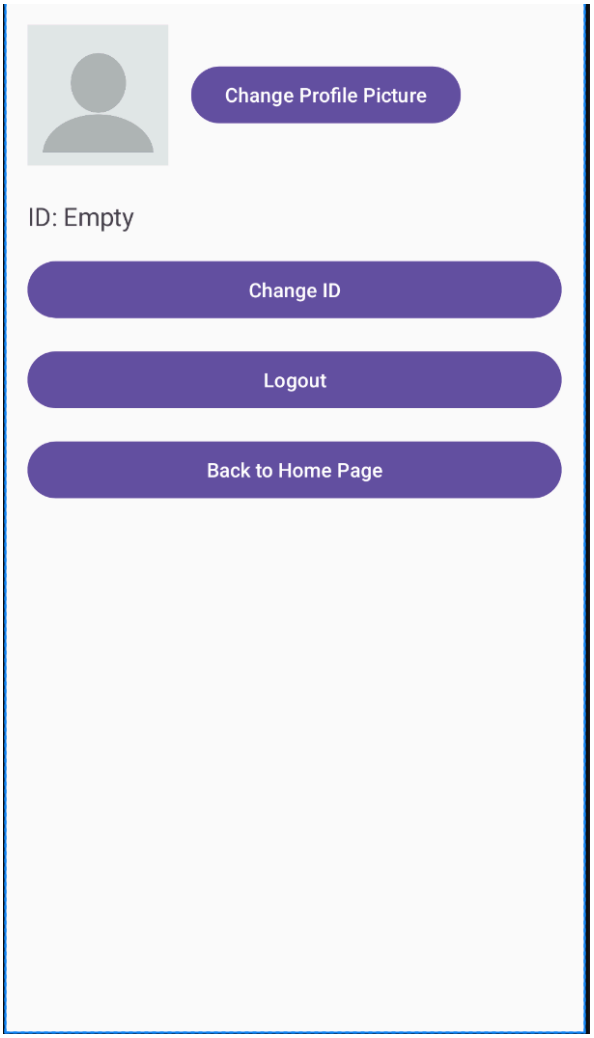

Illustration 3–7 Setting View

3.2.6 Sheet Editing

For this feature, we built it based on 1993hzw's github open source api:doodle[4], and we make it more user-friendly and more suitable for sheet editting.

For the entrance page, we make it really concise and straightforward, as shown in the pic 3–8.

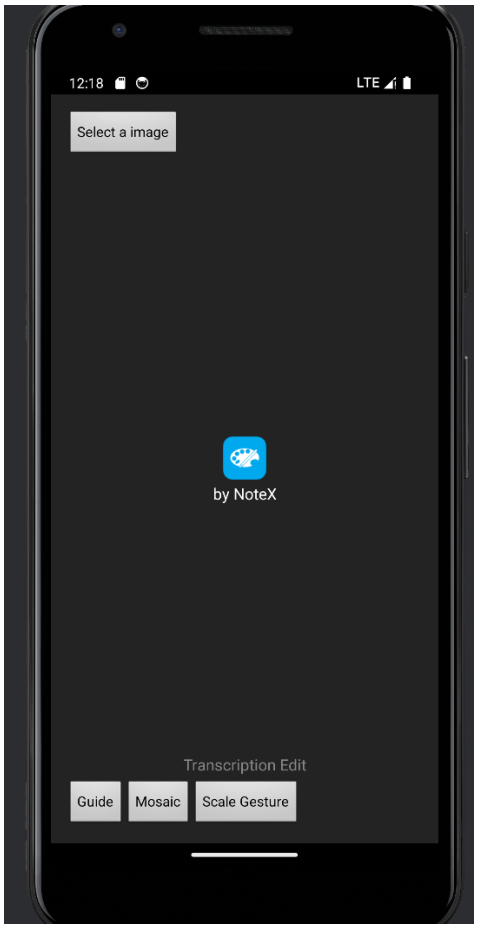

Illustration 3–8 Entrance for Editting

For the editting, we have the sheet in the middle of the screen, like pic 3–9, to give users a large place to focus on creating art.

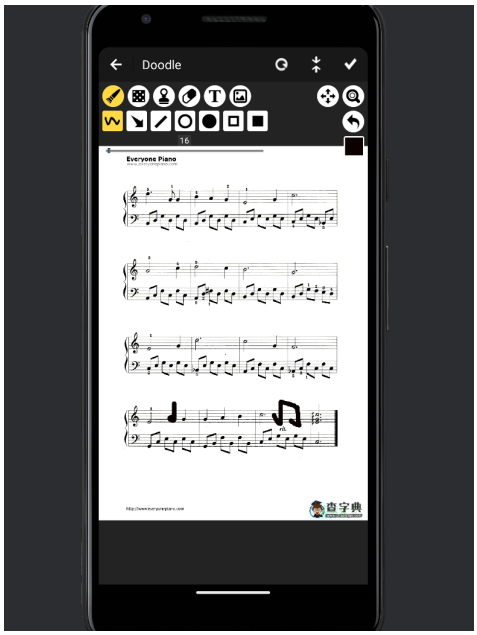

Illustration 3–9 Editting UI

We provide different colors for doodling, and we have a straightforward color circle for users to adjust their favorite colors by dragging on the circle, as shown in pic3–10

Illustration 3–10 Choosing color

We provide different brush and shape. The brush can choose hand-painted, mosaic, imitation, eraser, text, texture, and the imitation function is similar to that in PS, copying some-

where in the picture. Shapes can be selected from hand-drawn, arrows, lines, circles, rectangles, and so on. The background color of the brush can be selected as a color, or an image. All these options are listed on the first and second line on the top of the screen, in pic3–11, user choose brush in the first line first and choose shape in the second line, which is really easy to use and understand.

Illustration 3–11 Editting UI

In pic 3–12, we put the undo and zoom and rotate on the right top corner, which is separated from the brushes and shape options, to let users know the different zone for different type of functions.

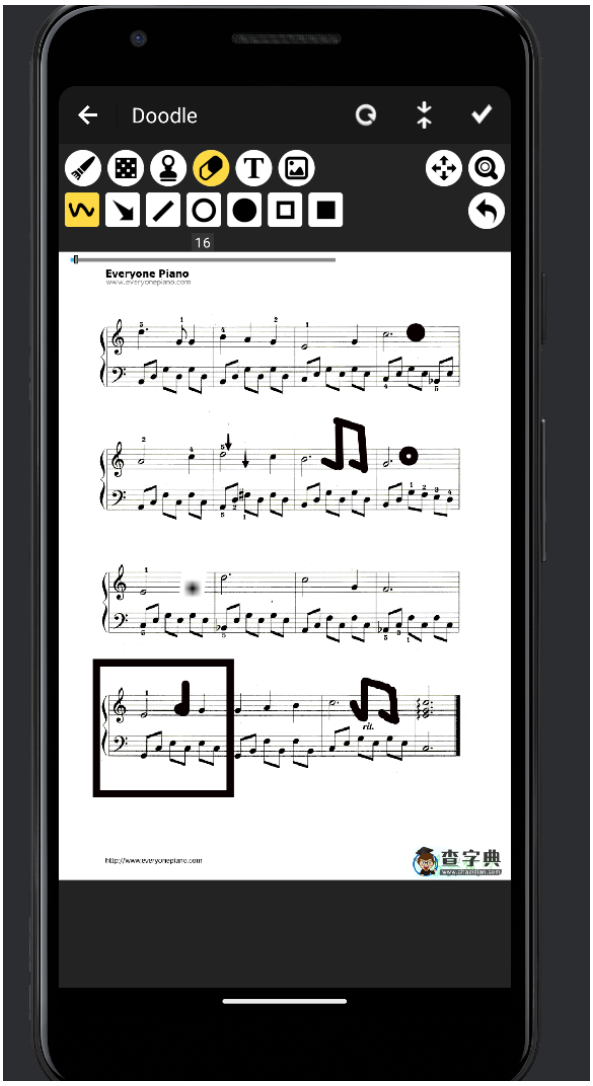

Illustration 3–12 Editting UI

Users can add another picture on the original picture, and we put a slider on the top to let user slide the bar and adjust the size of the picture, which is straightforward. We also make buttons for users to put the picture on the top or the bottom of the screen at one go, as shown in pic3–13

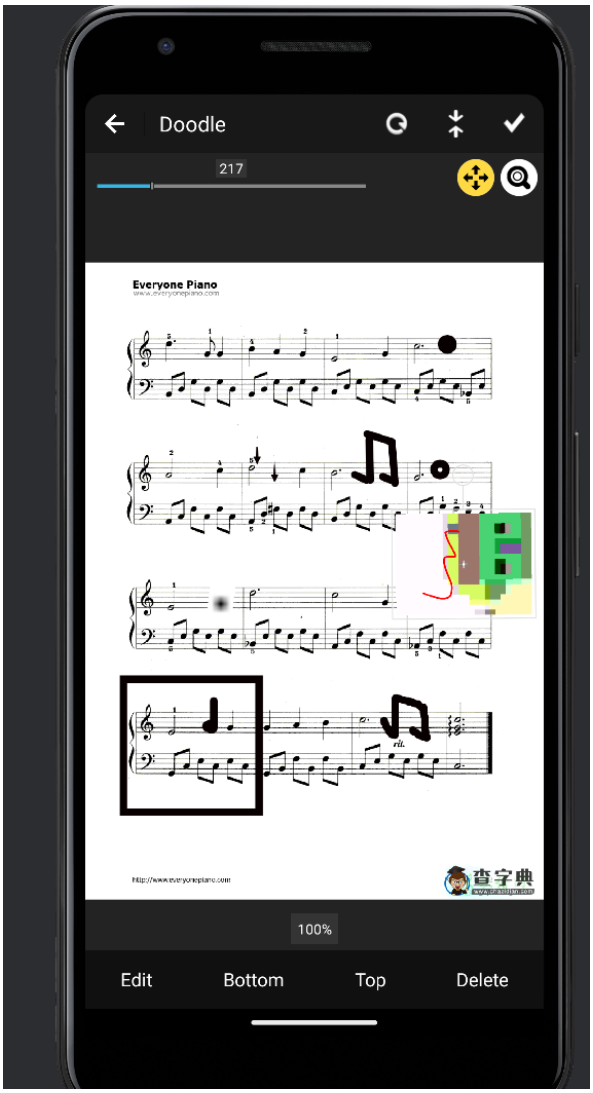

Illustration 3–13 Editting UI

In the final saving page, when user choose to save the editted picture, we show the path where the picture is saved on the top of the screen, in order to make it easy for user to track the picture. We also put selecting a new image at the left side to make it convenient for user to quickly start another editting.

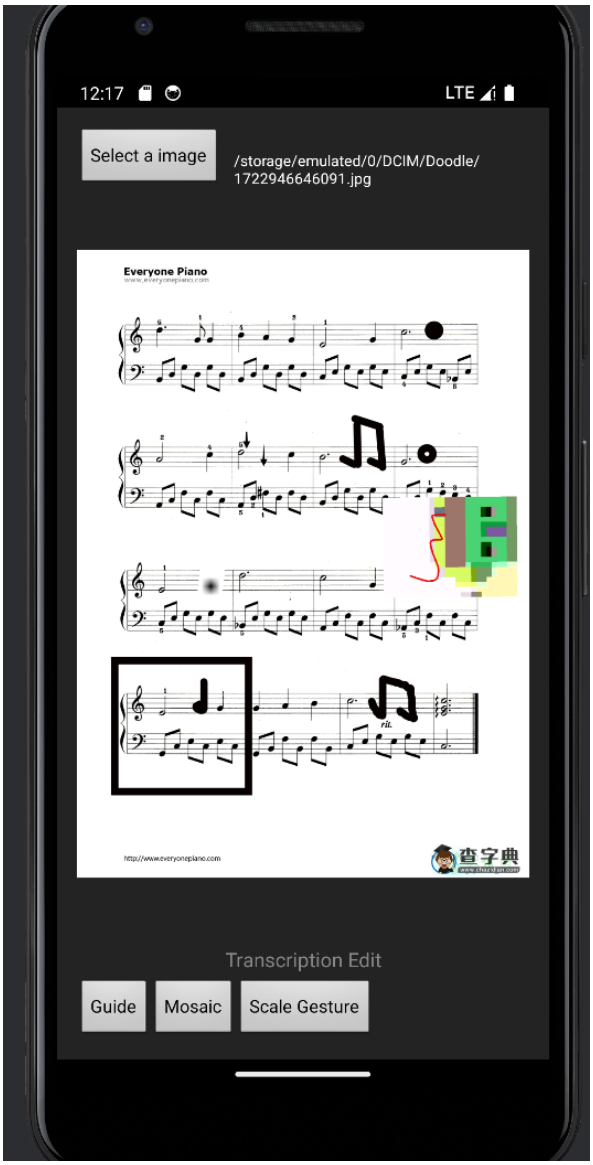

Illustration 3–14 Saving UI

Chapter 4 Back End Design

4.1 Back End Implementation of Note-X Application

The back-end implementation of our Note-X application is designed to efficiently manage the complex task of converting audio inputs into musical notation while ensuring scalability, maintainability, and reliability. This section details the core components, architectural decisions, and technologies used in the back-end system to support the application' s music transcription and audio detection functionalities.

4.2 Architectural Overview

The back-end architecture of Note-X follows a microservices-inspired design, where each core functionality is modularized into separate components. This approach allows for scalability, easier maintenance, and the ability to update or replace individual services without affecting the entire system.

The core back-end services include:

- **User Authentication:** Handles the users' accounts management including signing up, signing in, logging services, Renaming and password changing services.
- **Audio File Handling Service:** Responsible for receiving, validating, and storing audio files uploaded by users.
- **Audio Processing Service:** Handles the conversion of audio files to MIDI and subsequently to MusicXML format.
- **Music Transcription Service:** Transcribes the MIDI files into musical notation and generates a corresponding PNG image.
- **Error Handling and Logging Service:** Captures and logs any errors that occur during processing, ensuring the system remains robust and that issues can be traced and resolved.

This modular architecture not only enhances the system's scalability and maintainability but also allows for easier updates and improvements to individual components without affecting the overall system functionality. By isolating specific functionalities into distinct

modules, we can ensure that each component can be developed, tested, and maintained independently, improving the overall robustness of the system.

4.3 Core Functionalities

4.3.1 User Authentication

User authentication employs secure methods for handling user credentials and sessions. We utilized Django REST Framework's (DRF) built-in authentication mechanisms, including token-based authentication and session management. User account is used to identify the permission, ensuring the robustness and security of the user data. Users are required to register for an unique account in order to manipulating their own music base. Invalid or repeated registration information would be rejected and users would be informed of seriously registration. In this way, each user only has access his/her own music space and editing are allow inside the space. This ensures secure access to user data and prevents unauthorized access. We also incorporated additional security measures such as two-factor authentication and regular security audits to further protect user information. Additionally, the young generation's customization demands are met by enabling personalization of the head portrait as well as the login name. Whenever the user is proved to be legal, our server provide a token to recognize the user and the following management to music files are signed by the token. The authentication process involves validating user credentials, generating secure tokens, and managing user sessions to ensure continuous and secure access to the system.

4.3.2 Audio File Handling

The Note-X application allows users to upload audio files in various formats, including WAV and MP3. Upon receiving an upload request, the file handling service performs the following tasks:

- **File Validation:** Ensures that the uploaded file is of a supported type (WAV or MP3) and within the acceptable size limits.
- **Format Conversion:** Converts MP3 files to WAV format to standardize the processing workflow. This conversion is performed using the `pydub` library.
- **File Storage:** The validated and/or converted audio files are stored securely on the

server’s file system, with organized directory structures for easy access.

4.3.3 Audio Processing and MIDI Conversion

The audio processing service is the backbone of the Note-X application, where the core task of converting audio files into MIDI sequences occurs. This service utilizes the `aubio` library to detect pitch and extract musical notes from the audio waveform:

- **Pitch Detection:** Using the YIN algorithm, the audio is processed to detect pitch in the frequency domain. This is achieved by breaking the audio signal into frames and analyzing each frame for its fundamental frequency.
- **MIDI File Generation:** The detected pitches are translated into corresponding MIDI notes. The `midiutil` library is then used to construct a MIDI file from these notes, representing the musical structure of the audio input.

4.3.4 Music Transcription and PNG/PDF Generation

Once the MIDI file is generated, the music transcription service converts the MIDI data into musical notation:

- **MIDI to MusicXML Conversion:** The `music21` library is employed to parse the MIDI file and convert it into MusicXML format, a standard format for representing musical notation in a structured, machine-readable way.
- **LilyPond Integration for Image and PDF Generation:** The MusicXML file is then processed by LilyPond, a powerful music engraving program, to generate a high-quality PNG image of the musical staff. In addition to PNG, LilyPond also supports the generation of PDF files, providing users with a print-ready format of their transcribed sheet music. This flexibility allows users to choose the most appropriate format for viewing, sharing, or printing their musical scores.
- **Error Handling:** The system includes robust error detection mechanisms to handle issues such as incomplete pitch detection, invalid MIDI data, or failures in the MusicXML conversion. Errors are logged for further analysis and debugging.

4.3.5 Historical Audio Search

Note-X offers users the ability to search and retrieve previously uploaded or recorded audio files. This functionality is supported by a robust database management system that indexes audio files by metadata, such as date, title, and tags. The back-end handles search requests through efficient query processing, ensuring that results are delivered promptly even as the database grows.

4.4 Challenges and Solutions

The implementation of Note-X's back-end services posed several challenges, particularly in ensuring the accuracy and performance of the music transcription process. These challenges were addressed as follows:

- **Handling Large Files:** To mitigate issues with large audio files, such as memory overflow or long processing times, the system implements chunked file processing and optimized pitch detection algorithms.
- **Maintaining System Stability:** The modular architecture allows for isolating and handling failures in individual components without affecting the overall system. For instance, if the MIDI generation fails, the error is logged, and the user is notified, but other system functionalities remain operational.
- **Ensuring Cross-Format Compatibility:** By incorporating libraries like `magic` to detect file types and `pydub` for audio format conversion, the system supports a wide range of input formats while maintaining consistency in output.
- **Data Storage:** Manages user data, transcriptions, and shared resources. By using a relational database management system, we ensure data consistency, integrity, and efficient retrieval. Our database schema is optimized for performance, allowing for quick access to large datasets. The data storage component is designed to support high volumes of read and write operations, ensuring that user data and transcription records are efficiently managed and retrievable.

4.5 Security Considerations

Given that Note-X processes and stores user-uploaded files, several security measures are implemented:

- **Input Validation:** All uploaded files undergo rigorous validation to ensure they are of expected types and sizes.
- **Secure File Storage:** Files are stored in a secure directory with appropriate permissions to prevent unauthorized access.
- **Logging and Monitoring:** Comprehensive logging is in place to track all file uploads, processing steps, and any errors that occur, ensuring quick identification and resolution of potential security issues.

4.6 Back End Implementation of Editor

The back-end implementation of our Note-X application is crucial for ensuring that the system is scalable, maintainable, and capable of handling a wide range of user requests efficiently. By adopting robust design principles and modern architectural patterns, we developed a back-end system that meets these demands. This section delves into the specific aspects of our back-end system, highlighting the architectural decisions, core functionalities, and the overall design philosophy that guided our development process. The meticulous design and implementation of these components ensure that our application can handle various tasks seamlessly, providing a reliable user experience. This foundational setup is designed to support the complex processing required for converting audio inputs into musical notation, ensuring high accuracy and efficiency.

Chapter 5 Conclusions

5.1 Main Conclusions

The development of the Note-X project has significantly advanced the field of automatic music transcription by leveraging cutting-edge technologies, including deep learning algorithms, real-time audio processing, and user-centered design principles. This project successfully addressed the challenges associated with music transcription, such as the accurate detection of pitches, real-time editing, and the generation of readable sheet music.

One of the main achievements of this project is the integration of the SPICE algorithm within the Note-X system, enabling efficient pitch detection and audio analysis. The implementation of this algorithm, packaged as an API, allowed the system to deliver high-fidelity music transcription, even in complex polyphonic contexts. This achievement underscores the importance of advanced signal processing techniques and their potential to revolutionize traditional approaches to music transcription.

Another key outcome is the user interface design, which was guided by the principles of user experience (UX) design. The interface allows users to interact with the transcription system in a seamless manner, providing real-time feedback and editing capabilities. The user experience was further enhanced by the ability to separate vocals and accompaniment, a feature that caters to the specific needs of musicians who require distinct tracks for different instruments.

Moreover, the Note-X project demonstrated the feasibility of real-time music transcription and editing, which is a significant leap forward in the field. The ability to transcribe music in real-time, combined with intuitive editing tools, empowers musicians to create and refine their work on the fly, enhancing creativity and productivity.

5.2 Technical Contributions

From a technical standpoint, the Note-X project made several noteworthy contributions. The integration of deep neural networks for pitch detection and the use of spectrogram analysis were pivotal in achieving high accuracy in transcription. These methods allowed the

system to handle a wide range of musical genres and styles, making it versatile and adaptable.

The architecture of the Note-X system, which includes the use of Django REST Framework for API development, provided a robust and scalable solution for handling the computational demands of real-time transcription. The modular design of the system ensures that it can be easily extended or modified to incorporate new features or adapt to different use cases.

Furthermore, the project explored the use of data augmentation techniques to enhance the training of neural networks, thereby improving their ability to generalize across different types of music. This approach not only enhanced the system's performance but also provided valuable insights into the application of machine learning in music processing.

5.3 Research Outlook

Looking forward, there are several areas where the Note-X project could be expanded or improved. One potential direction is the exploration of unsupervised learning techniques to further reduce the reliance on labeled datasets. This could open up new possibilities for developing transcription systems that are capable of learning from large amounts of unlabeled data, thus increasing their versatility and applicability.

Another area for future research is the enhancement of real-time processing capabilities, particularly in terms of reducing latency and improving the responsiveness of the system. As technology continues to advance, there is potential to achieve near-instantaneous transcription and editing, which would greatly benefit live music performances and other time-sensitive applications.

Finally, the integration of more advanced user interface features, such as gesture-based controls or voice commands, could further enhance the user experience. These features would make the system more accessible to a wider audience, including those with disabilities or those who prefer hands-free interaction.

In conclusion, the Note-X project represents a significant step forward in the field of music transcription, offering a powerful tool for musicians, composers, and music educators. The technical achievements, combined with the user-friendly interface, make it a valuable contribution to the field. As the project continues to evolve, it holds the promise of further

innovations that could transform the way music is transcribed, edited, and shared.

References

[1] MuseScore. MuseScore: Free music composition and notation software[EB/OL]. 2024. https://musescore.org/zh-hans.
[2] PreSonus. Notion[EB/OL]. 2024. https://legacy.presonus.com/products/Notion.
[3] Scorecloud. scorecloud: Free music notion software[EB/OL]. 2024. https://scorecloud.com/.
[4] Haozhi Zhang. oodle: A Real-time Doodle Animation Framework. GitHub repository.[EB/OL]. 2020. https://github.com/1993hzw/Doodle.

Team Commitment Contract Appendix 1

Team Commitment Contract

Team Commitment Contract

Team Norms & Expectations

As a team, it is important to name norms and expectations that you share moving forward. These agreed-upon norms will guide behavior and enable accountability as the year progresses. The specifics are important.

Team member work patterns

Some students like to work late at night, others are early risers and prefer to work in the mornings. Some students work best on a tight deadline, others prefer to work ahead as much as possible. Discuss as a team and share your personal work habits and tendencies.

Work at night, work ahead.

Use of communication channel

Frequency of response: how much time is reasonable for email and Feishu responses when communicating with each other on the team?	within 4 business hours
Tone and usage: how will you indicate if something is urgent? What type of discussions can happen in group vs. what type discussions should happen one-on-one?	casual but no profanity

Work-together time

When will your team get together to work on the project each week?
Even if you decide to set aside time for sub-teams to meet, you MUST still have an "all hands" meeting to work out the interfacing between sub-teams.

Time expectation: 16 hours/week for a 4-credit course, incl. class time.

Based on previous years experience, successful teams who have earned the highest grades schedule time and work together for at least 4 hours per week outside of the instructor meeting time. The remaining hours are spent on individual efforts.

Who will be working together	Day	Time	Location
All hands meeting (**REQUIRED**, online ok)	4h a week	Flexible	Online

1 of 5

Team Commitment Contract

Punctuality

What is your expectation for punctuality at team working and meeting times?	within 10 mins of starting time

Meeting management

How will your meetings be run? Will you ever cancel a weekly meeting? How much advance notice is required to cancel work-together meetings?

Online. Cancellation needs a notice 24 hours in advance.

Decision making

How will your team make decisions? All collaborative? At some point will you give autonomy to certain members? How will you communicate decisions that are made to one another during independent or subteam work time?	All collaborative

Project Manager

Will you appoint one person for the term or will you rotate the role?	TBD

The PM serves as the project lead/scrum manager; turns in all team assignments; manages the project plan and timeline; delegates tasks; ensures project milestones and deadlines are met.

All team members must review and approve all assignments prior to submission by the Project Manager.

Due to the relatively small size of 441 projects, the PM role cannot be exclusive: even if you appoint one person to be the PM, they MUST also contribute code.

Project publication

How do you want to publish your project at the end of term? By making your project's GitHub repo public or by publishing your app to Apple App Store/Google Play (not free)?

Github repo

Fail fast, fail early, fail often

Team Commitment Contract

When a team member realizes they are stuck and cannot meet deadline, when must they inform the rest of the team?	48 hours wall-clock time

Effort expectation

Some students strive for A+, others are happy to pass the course with a C. Have you had an honest discussion among yourselves as to what each member expects from the team? (Y/N)

N

Team conflict

Occasionally, there will be interpersonal team conflicts. How will you handle them? E.g. first discuss with the team member in person? When would you escalate to your instructor?

Discuss with the team member in person first.

3 of 5

Team Commitment Contract

Cause for dismissal

Following are expected team member behaviors. A demonstrated inability to follow any one of these expectations may cause dismissal from the team. Please let the teaching staff know when this happens. The teaching staff will talk to the team member in person and improve the situation.

Team member did not complete a task that was assigned to them and did not:

- promptly alert the team and teaching staff that they were struggling
- reach out for help to another team mate or teaching staff when struggling
- document their attempts to complete the task

  acceptable documentation: several git commits over the days leading up to a deadline, with code showing that the member had put the time and effort into attempting the task rather than fail to complete the task due to procrastination and lack of effort.

2. Team member failed to respond to messages from other team members within the "Frequency of response" stipulated in this Team Commitment Contract and did not:
   - warn the team of a family or medical emergency that would leave them unable to communicate
   - provide teaching staff with a proof of medical/family emergency as reason why they were unavailable
3. Team member failed to show up, call in, or participate in 3 or more team meetings and did not:
   - warn the team of a medical or family emergency that would leave them unable to attend
   - provide teaching staff with a proof of medical/family emergency as reason why they were unavailable
4. Team member failed to conduct themself professionally and collegially. Professional and collegial interaction and behavior means civil interactions with other team members; it is not acceptable, for example, to yell at or openly insult other team members during group meetings or showing other disrespectful attitudes.

Team Commitment Contract

Team member signatures

By signing electronically below, you are attesting that you have read this whole Team Commitment Contract, that you have been part of the discussion on each item listed in the Contract, and that you swear to abide by the decisions documented herein.

Member name	jAccount	Signature
Lingyu Qi	sjtu_qly	Lingyu Qi
Yifan Jia	Jiayf_0915	Yifan Jia
Marco Souza	marco216	Marco Souza
Yue Huang	hyhy2001	Yue Huang
Shuangyu Lei	shuangyu	Shuangyu Lei
Jingjing Zhu	jingjingzhu	Jingjing Zhu

5 of 5

Questionare Appendix 2

Questionare

• 1. What is the biggest challenge you face when transcribing music?

• 2. What tools or software do you usually use to generate and edit sheet music? What are their shortcomings?

• 3. If there were a software that could automatically convert audio to sheet music, what do you think would be the most important feature? Why?

• 4. If there were a software that could automatically convert audio to sheet music, do you think having a real-time editing feature is necessary? Why?

• 5. After creating sheet music, in what formats do you usually export it? What are the different uses of these formats?

• 6. When using sheet music software, do you need to separate vocals and accompaniment? How helpful is this feature for your music creation?

• 7. In your daily music creation, what other features do you think are lacking in current tools that you consider essential?

UI/UX Mockups Appendix 3

The UI/UX mockups were an essential part of our design process, allowing us to visualize and refine the user experience before development. Below is a summary of the results from our user testing sessions.

3.0.1 Test Recording

A video recording of our user test sessions can be found at the following link: https://youtu.be/tRtmCntXWeI.

3.0.2 UI/UX Mockups Results

Test 1:

- **Tester:** Marco Souza, 苏文俊, Huang Yue, 黄越
- **Recorder:** Lingyu Qi, 祁令雨
- **Test Participants:** Bohan Shu, Bole Li, Hangxin Chen
- **Results:** The difference between the Brush and Edit tools in the editing interface was not clear. Additionally, multiple languages were not supported.

Test 2:

- **Tester:** Shuangyu Lei, 雷双羽, Jingjing Zhu, 朱菁菁
- **Recorder:** Yifan Jia, 贾逸凡
- **Test Participants:** Yilin Jia, Ziheng Li, Haoyu Chen
- **Results:** All features were functioning as expected.

3.0.3 Summary of Findings

Qualitative Results:

- The distinction between the Brush and Edit tools in the interface was unclear to users.
- The application did not support multiple languages.

Quantitative Results:

- Task 1.1 (Sign in): Completed in 2 clicks.
- Task 1.2 (Upload file): Completed in 2 clicks (Test 1) and 1 click (Test 2).

- Task 2.1 (Save transcription): Completed in 4 seconds (Test 1) and 3 seconds (Test 2).
- Task 3.1 (Switch between transcriptions): Completed in 3 seconds (Test 1) and 2 seconds (Test 2).
- Task 3.2 (Edit transcription): Completed in 90 seconds (Test 1) and 70 seconds (Test 2).
- Task 4.1 (Search for a transcription): Completed in 8 seconds (Test 1) and 10 seconds (Test 2).
- Task 5.1 (Find edit history): Completed in 4 seconds (Test 1) and 5 seconds (Test 2).
- Task 5.2 (Open past audio): Completed in 4 seconds (Test 1) and 2 seconds (Test 2).

3.1 Additional Observations

Choosing the Brush for Editing: The Brush tool provides higher precision by allowing users to draw or modify notes directly on the score. This method can be more intuitive and efficient for users, especially when dealing with complex musical notations or ornaments.

Adding English to the App: Supporting the English language in the application will significantly widen its accessibility, making it usable by a broader audience globally. This addition will also facilitate the creation and distribution of support materials and documentation, thereby enhancing user experience.

3.2 Conclusion

The results from the UI/UX mockups and subsequent user testing provided valuable insights into the functionality and user experience of the application. While certain areas, such as the distinction between tools and language support, were identified as needing improvement, overall, the application performed well in achieving its intended tasks.

Acknowledgements

The team would like to express deepest gratitude to the supervisor, Professor Pi Yibo, for his invaluable guidance, support, and encouragement throughout the development of this project. His expertise and insights were instrumental in shaping the direction and success of this thesis.

Appreciation is also extended to the project team members: Jia Yifan, Qi Lingyu, Marco Souza, Lei Shuangyu, Huang Yue, and Zhu Jingjing, for their hard work, collaboration, and dedication to the Note-X project. The success of this project is a testament to the collective effort and teamwork of all involved.

Furthermore, the team extends sincere thanks to the UM-SJTU Joint Institute for providing the necessary resources and environment that made this project possible. The knowledge and skills acquired during the time at the institute were crucial in the successful completion of this thesis.

Lastly, heartfelt gratitude is extended to family and friends for their unwavering support and encouragement throughout the academic journey. Their belief in the team's capabilities kept the members motivated and focused, and their love and support are deeply appreciated.

NOTE-X：REAL-TIME MUSIC TRANSCRIPTION AND EDITING APP

The Note-X project represents a significant advancement in the field of music transcription and editing, combining the latest developments in machine learning with a user-friendly interface. This thesis has explored the various challenges associated with automatic music transcription (AMT), particularly focusing on the complexities of polyphonic music and the integration of deep learning techniques to enhance transcription accuracy.

The project developed a comprehensive application that not only automates the process of music transcription but also provides robust editing capabilities, allowing users to fine-tune the transcribed music. By leveraging machine learning models, specifically Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), the system effectively handles the intricacies of overlapping notes and varying instrument timbres. This approach has resulted in significant improvements in transcription accuracy, making the Note-X application a valuable tool for both amateur and professional musicians.

Throughout the development process, the team faced several challenges, including the need for large, annotated datasets to train the machine learning models and the complexity of designing an intuitive user interface. However, through iterative design and testing, these challenges were successfully addressed, resulting in a robust and user-friendly application.

The thesis also highlights the importance of real-time feedback in the music editing process, which was achieved through the integration of advanced signal processing techniques and interactive features within the application. This allows users to immediately hear the effects of their edits, making the transcription and editing process more efficient and accurate.

In conclusion, the Note-X project demonstrates the potential of combining machine learning with user-centric design to create powerful tools for music transcription and editing. The application not only meets the current needs of musicians but also sets the stage for future developments in the field of AMT. The project's success underscores the importance of interdisciplinary collaboration, bringing together expertise in computer science, music theory, and user experience design to achieve a common goal.