UE19CS390A - Project Phase - 1

End Semester Assessment

Project Title : Synthesizing music by variation of musical elements and artists' style

Project ID : 46

Project Guide : Prof. P Kokila

Project Team : B Pravena PES2UG19CS076

Rishi Patel PES2UG19CS329

Swarnamalya A S PES2UG19CS418

Varna Satyanarayana PES2UG19CS448

Agenda

- Introduction and Motivation
- Problem Statement
- Abstract and Scope
- Literature Survey / Existing System
- Suggestions from Review – 3
- Requirements Specification
- Design Approach
- Design Constraints, Assumptions & Dependencies
- Proposed System / Approach
- Architecture
- Design Description
- Project Progress
- References

Introduction

- Music is playing an increasingly important part in day to day life.
- From medieval classical music to modern techno, different musical styles emerged from different culture and eras, serving different purposes, conveying different emotions.
- Music is fundamentally a sequence of notes.
- We are curious how could neural networks could help us with that since it has been very popular recently working on the audio task on neural networks.

Introduction and Motivation

Motivation

- Can computers mimic the human creative process
- Create new content
- Keep the legacy of an artist alive
- Adapt to old songs into current styles

Problem Statement

Rendering music in different singers' styles by modifying various elements of music such as Pitch, Duration, Dynamics, Tempo, Timbre, Texture and Structure.

Abstract and Scope

The goal is to obtain a better knowledge of automatic music generation and style transfer by evaluating where different computer models succeed and fail in superimposing the style of a singer on a song in the Indian Classical genre when compared to one another, and then to develop a model for the same.

- Focusing on Indian Music
- Retaining the song's melody and changing to new artists' style
- Endeavouring to preserving the songs' overall emotion and structure

Literature Survey/Existing System

Ref.	Authors	Name of Paper	Learning Model	Advantages	Drawback
[1]	O. Cífka, A. Ozerov, U. Şimşekli and G. Richard	Self-Supervised VQ-VAE for One-Shot Music Style Transfer. (2021)	VQ-VAE	There is a significant gap between the emotional similarity scores of the other three categories, so most of the emotional similarity scores can be learned The subjects' satisfaction with the experimental results is considerably high.	The use of a deterministic decoder. Sound quality of outputs is not nearly perfect. Faster notes and polyphony seem to pose a problem. Less predictable on real data.
[2]	J. Wang, C. Jin, W. Zhao, S. Liu and X. Lv	An Unsupervised Methodology for Musical Style Translation(2019)	LSTM, GAN and VAE	In order to improve generality and suppress overfitting, Gaussian noise is added to both real and fake input of the two discriminators. Accuracy of 75.05% for style translation from classical to jazz.	Dissonant notes occur occasionally in the generated samples. Plan on deploying more constraint on the generators in order to eliminate these dissonant noise.
[3]	Y. -Q. Lim, C. S. Chan and F. Y. Loo	ClaviNet: Generate Music With Different Musical Styles(2021)	HyperLSTM	Proposed continuous style embedding outperforms discrete label in guiding our model to generate music that resembles closely to a conditioned style.	Disentanglement issue Cannot generate music with longer duration while maintaining long-term structure.
[4]	C. -F. Huang and C. -Y. Huang	Emotion-based AI Music Generation System with CVAE-GAN(2020)	GAN	There is a significant gap between the emotional similarity scores of the other three categories, so most of the emotional similarity scores can be learned. The subjects' satisfaction with the experimental results is considerably high.	Manual collection of data results in a smaller dataset.

Literature Survey/Existing System

Ref.	Authors	Name of Paper	Learning Model	Advantages	Drawback
[5]	Sageev Oore, Ian Simon, Sander Dieleman	This time with feeling: learning expressive musical performance	LSTM	Voiced sounds can be duration–modified to match the target song duration. They have added naturalness to the synthesized song.	Creating music with long-term structure (e.g., more than several seconds of structure) is still a very challenging problem.
[6]	R. Lang, S. Wu, S. Zhu and Z. Li	SSCL: Music Generation in Long-term with Cluster Learning	VQ-VAE	Generates high fidelity and coherent songs. Models can follow along most prompts and even sing new words that are reasonably pronounceable.	Doesn't have a developing musical and emotional structure across the entire piece.
[7]	en-Yu Liu, Yu-Hua Chen, Yin-Cheng Yeh, Yi-Hsuan Yang	Score and Lyrics-Free Singing Voice Generation	GRU and GAN	The transformation is at least partly successful in more than 75 % of cases	Sound Quality of this model gets rated lower than the other known models (Sinsy, Synthesizer V). Hence, there is room for improvement.
[8]	M. Ragesh Rajan	Singing Voice Synthesis System for Carnatic Music	Harmonic plus Noise Model	Mean Opinion Score (MOS) is chosen as the metric after bringing in 10 subjects to listen to the music.	For synthesizing Carnatic music properly, "gamakas" also need to be perfectly analyzed, modeled and added to the pitch contour of the song, but it's very hard to find its notes.

Literature Survey/Existing System

SL No.	Authors	Name of Paper	Learning Model	Advantages	Drawback
[9]	Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, Ilya Sutskever	Jukebox: A Generative Model for Music	VQ-VAE	Generates high fidelity and coherent songs. Has ability to record an incomplete idea and explore various continuations without ever needing to tabulate in symbolic representations.	Doesn't have a developing musical and emotional structure across the entire piece.
[10]	Nachmani, E. and Wolf, L.	Unsupervised singing voice conversion	CNN	Sample collection is easy as this is an unsupervised model	Training the model without data augmentation makes accuracy of the model greatly suffer. Works on audio without background music
[11]	Blaauw, M., Bonada, J. and Daido, R.	Data efficient voice cloning for neural singing synthesis	Multi speaker model along with wavenet vocoder	Wavenet vocoder decreases the buzzy sound of the synthesized waveform. Multiple speaker model makes it such that no need to train different models for different singers	Constant pitch singing is not there in most songs to have pseudo singing. Listening tests for evaluation. Clones only timbre and not expressive aspects of target voice
[12]	Daher, R., Zein, M.K., El Zini, J., Awad, M. and Asmar, D.	Change your singer: a transfer learning generative adversarial framework for song to song conversion.	SCM-GAN	Makes use of transfer learning along with SCM_GAN to improve performance, increases naturality and similarity. The gated linear units (GLUs) that act as data driven activation functions	Difficult to objectively test the effect of splitting the background music from vocals. Dta with background music has an adverse effect on the performance of the conversion model

Summary of Literature Survey in Review 2

- Manual data collection
- Noisy generated output
- Faster notes seem to pose a problem during generation
- Wavenet decoder produces the least noisy output
- VQ-VAE models produce better results than GANs (less noisy output)
- Data augmentation can be done using GANs

Summary of Literature Survey in Review 2

- GANs can also be used for style transfer
- Data augmentation done while using GANs as main model for style transfer
- Self supervised models require lesser data for training
- Self supervised models require much more computational power than supervised learning(data labeling)
- Self supervised models compromise on the output classification and quality
- Supervised produces better output but requires greater amount of data

Suggestions from Review - 3

- Start preparing the dataset to train and test the model
- Decide on number of artists' voice to train model
- Decide number of songs for each artist

Functional Requirements

- Dataset creation and augmentation by collecting data (songs) of various singers
- The model is trained based on the singer's voice and style
- The model is tested by inputting a new song and a selected singer to convert it to
- Testing of the output is done manually due to the lack of a quantitative measure
- Data augmentation to expand the dataset using GANS
- Singers' style transfer and learning done using VQ-VAE

Non - Functional Requirements

- Performance Requirement
  - Accurate Performance
  - Ability to work on data from different platforms
- Security Requirements
  - The user data should be anonymous so as to adhere to privacy of the people uploading music.
- Other Requirements
  - Usability: The system should be user-friendly.
  - Availability: The system should be available at any point in time.
  - Robust: System should give proper result for a given input to the user. Reducing False Positivity and False Negativity
  - Scalability: Systems need to work for large data values /audios.

Design Details

1. Novelty :

Most of the research and work in automatic music generation has been done on various Western Music Genres and instrumental music. The proposed model will implement style transfer on Indian Music. The input taken by most research papers are in midi files and majorly done on instrumental music and not songs.

2. Innovativeness :

The majority of the works consider emotion as well as genre, but not style. This would be a novel approach for the user to listen to a song in the style of another artist.

Design Details

3. Interoperability:

Regardless of the operating system on which it runs, the system will operate admirably.

Google Colab will be used to run the model, it is compatible with Linux, Windows and Mac.

4. Performance:

The model used for implementation will generate high quality music in another singers' style.

5. Maintainability:

The model aims for high cohesion and low coupling which will help make our system modules robust, reliable, reusable and understandable.

Design Details

6. Security

The user data should be anonymous in order to adhere to the privacy of the people uploading the music.

7. Reliability

The model will perform consistently well and will be fault tolerant.

8. Portability

Model will work on any operating system since Google Colab will be used for implementation.

Current System / Approach

- The current systems that are now available make use of VQ-VAE for style transfer
- Use GANs for data augmentation
- Models work only on instrumental music
- The music is represented using STFT spectrogram
- Disentanglement of pitch and timbre using spectrogram
- Use codebook for style embedding
- They make use of the standard decoders for obtaining the waveform

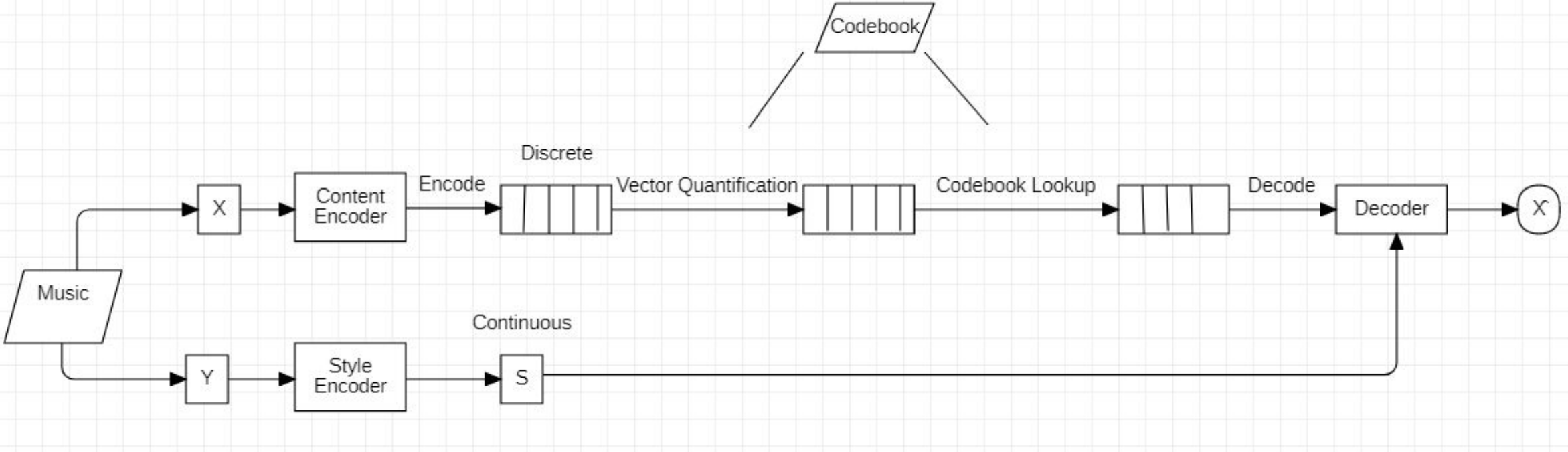
Current System / Approach

- Enhance current system by making use of CQT spectrogram instead of STFT spectrogram
- CQT representation of the song provides us pitch equivariance
- This property is very important since the pitch should remain constant in both the input and the output audio but only the timbre (quality of the tone) of the new artists' song should be transferred onto the current song.
- Further, to obtain better decoding of the spectrogram to the audio waveform we will be using the Wavenet Decoder.

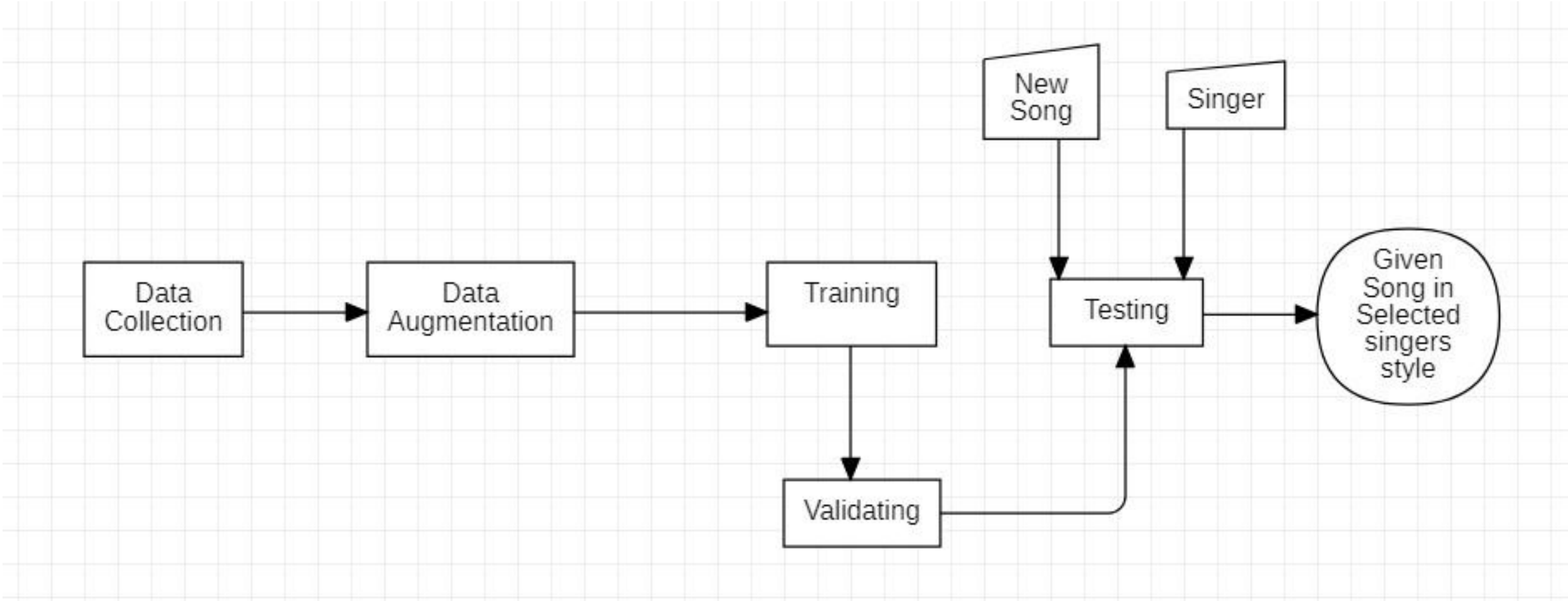
Proposed Methodology / Approach

- Input songs for both content and style are taken in the form of an mp3 file
- Input converted into spectrogram format (CQT)
- Passed through VQ-VAE model
- Two bidirectional encoders - content and style
- Two segments of same song X, Y is considered
- Both have different pitch but same timbre
- X and Y are passed through content and style encoder respectively
- Make use of code book for retrieving style embeddings
- Dot product between Encoders' output and codebook
- Pass output obtained from Encoder through unidirectional Wavenet decoder to merge new style and content
- Output the audio

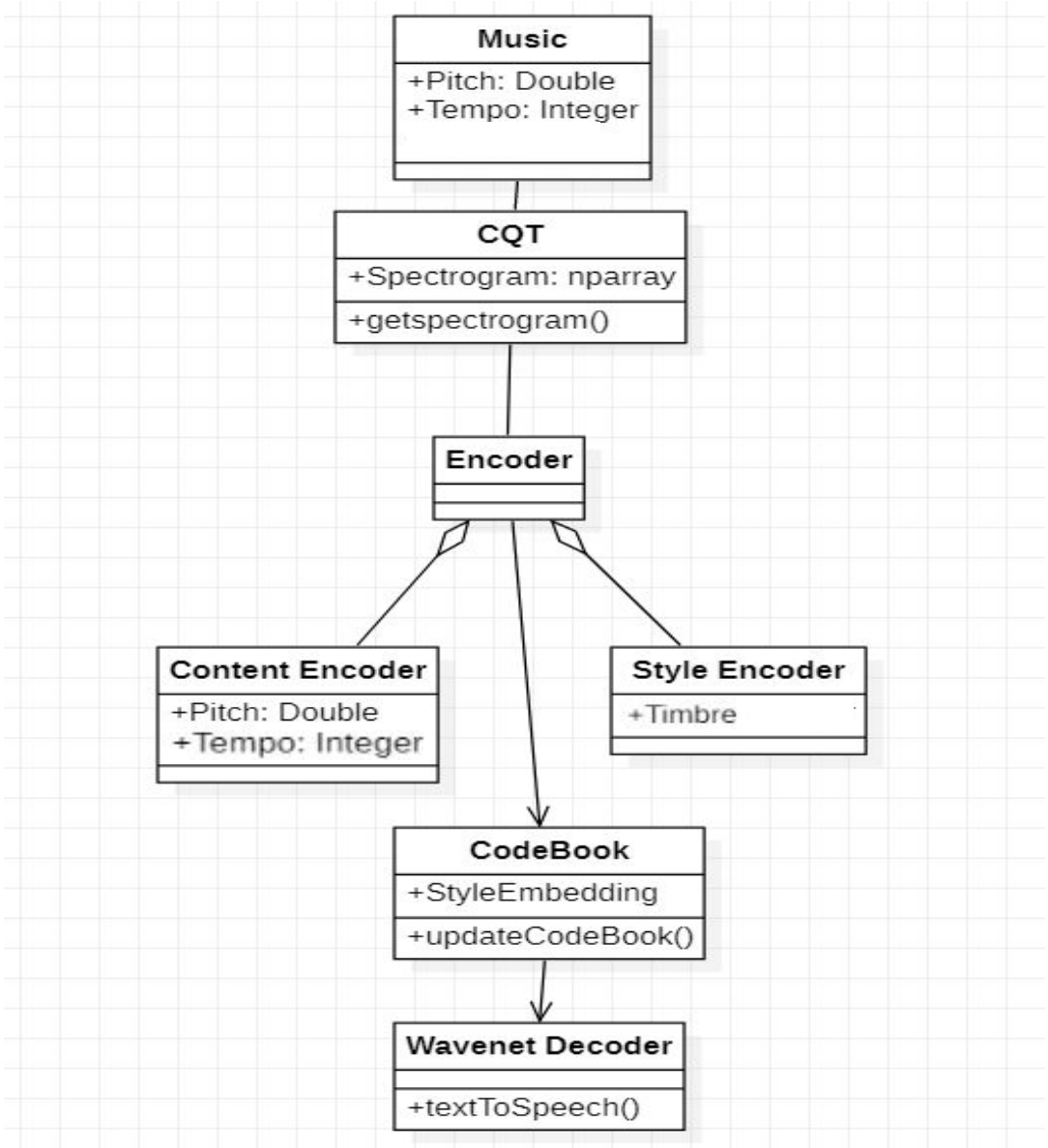
Proposed Methodology



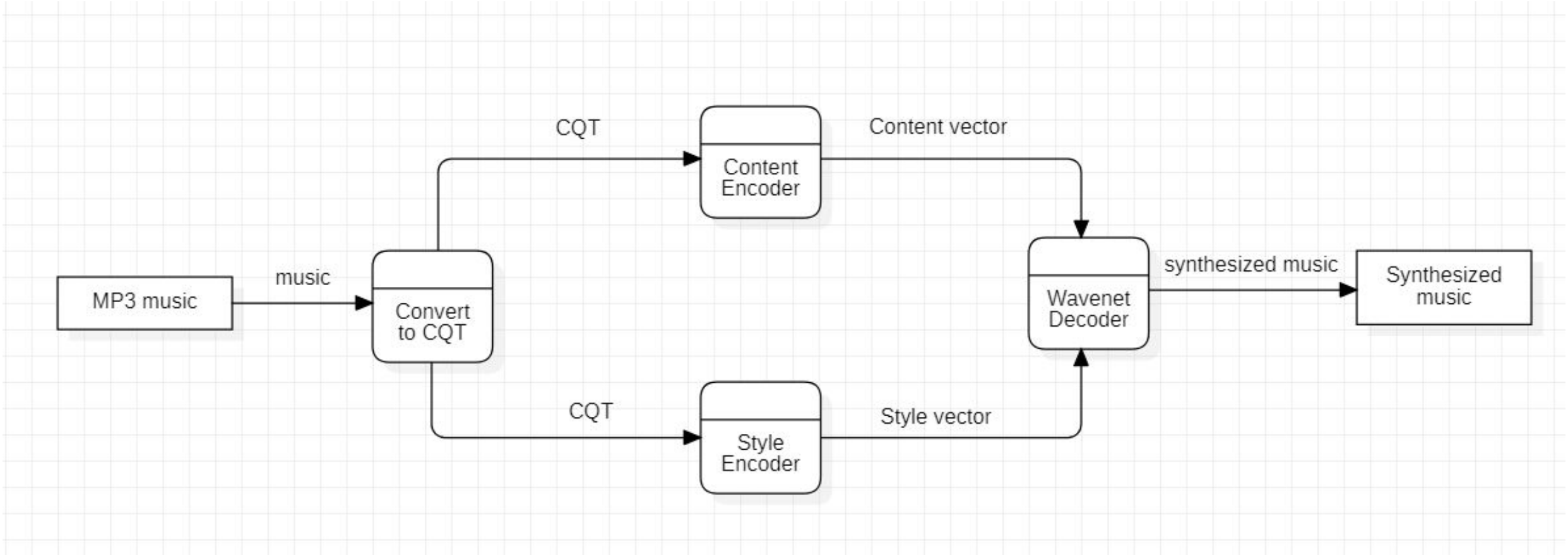
Architecture (if applicable)



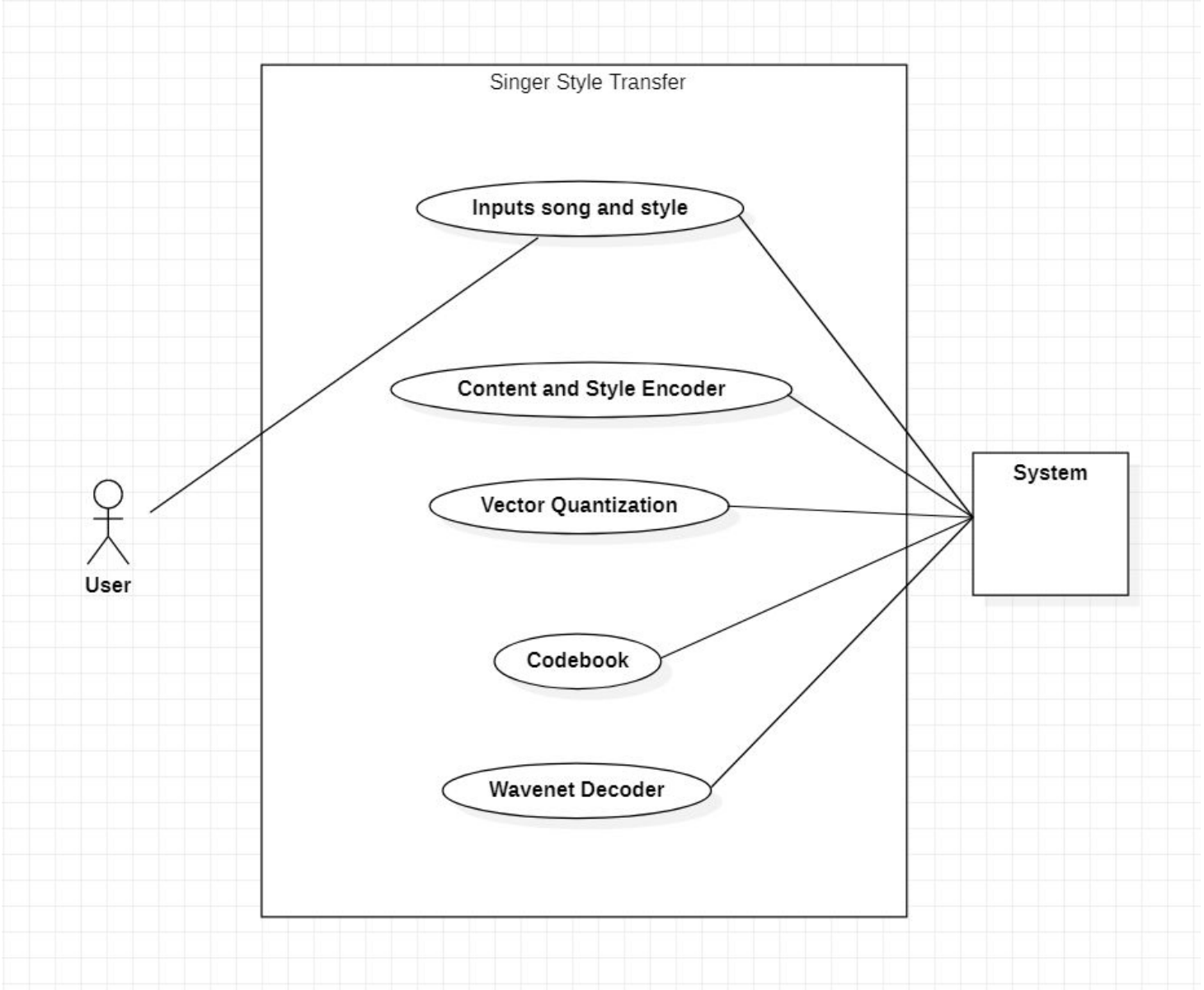
Class Diagram



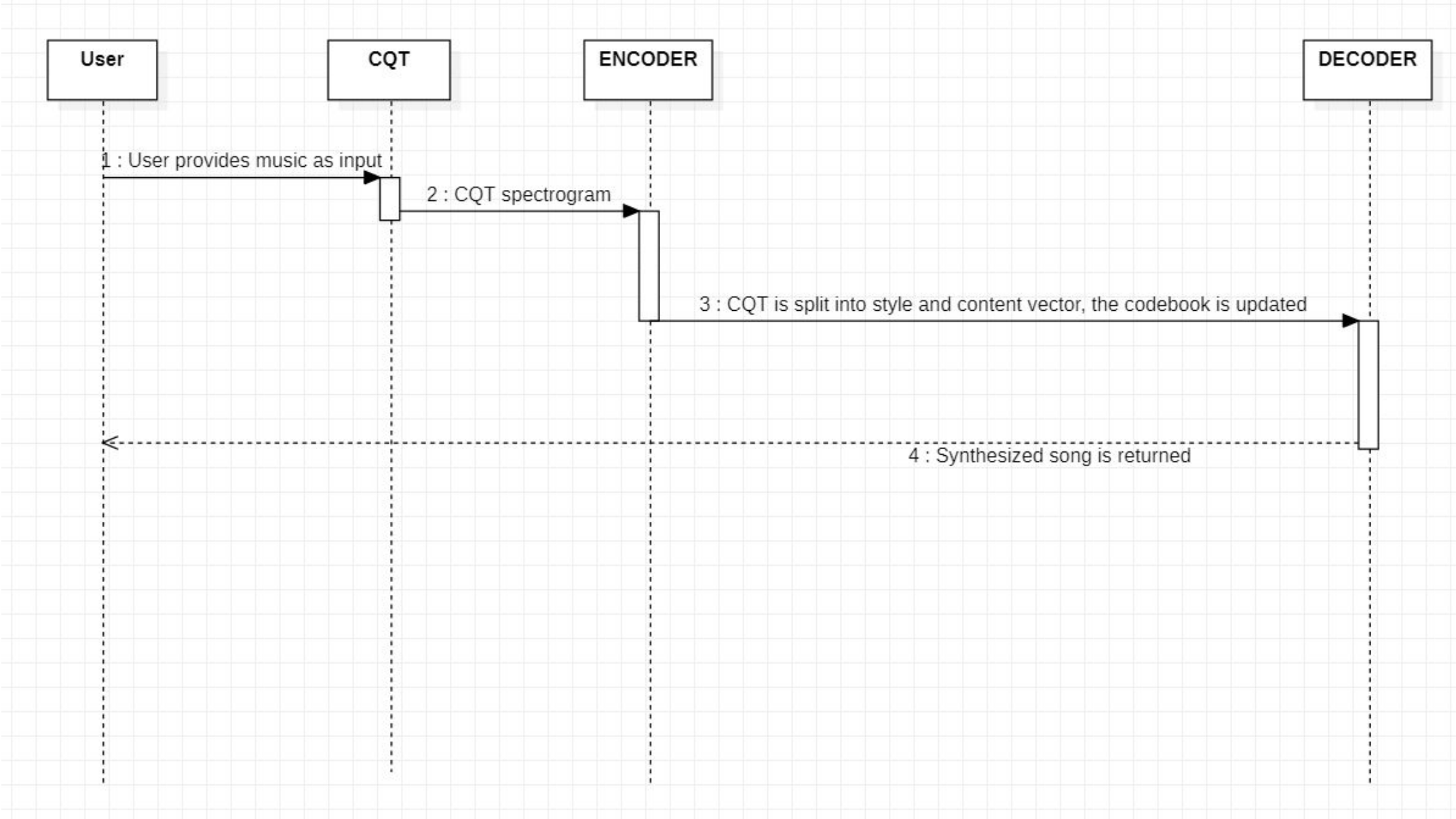
Data flow Diagram



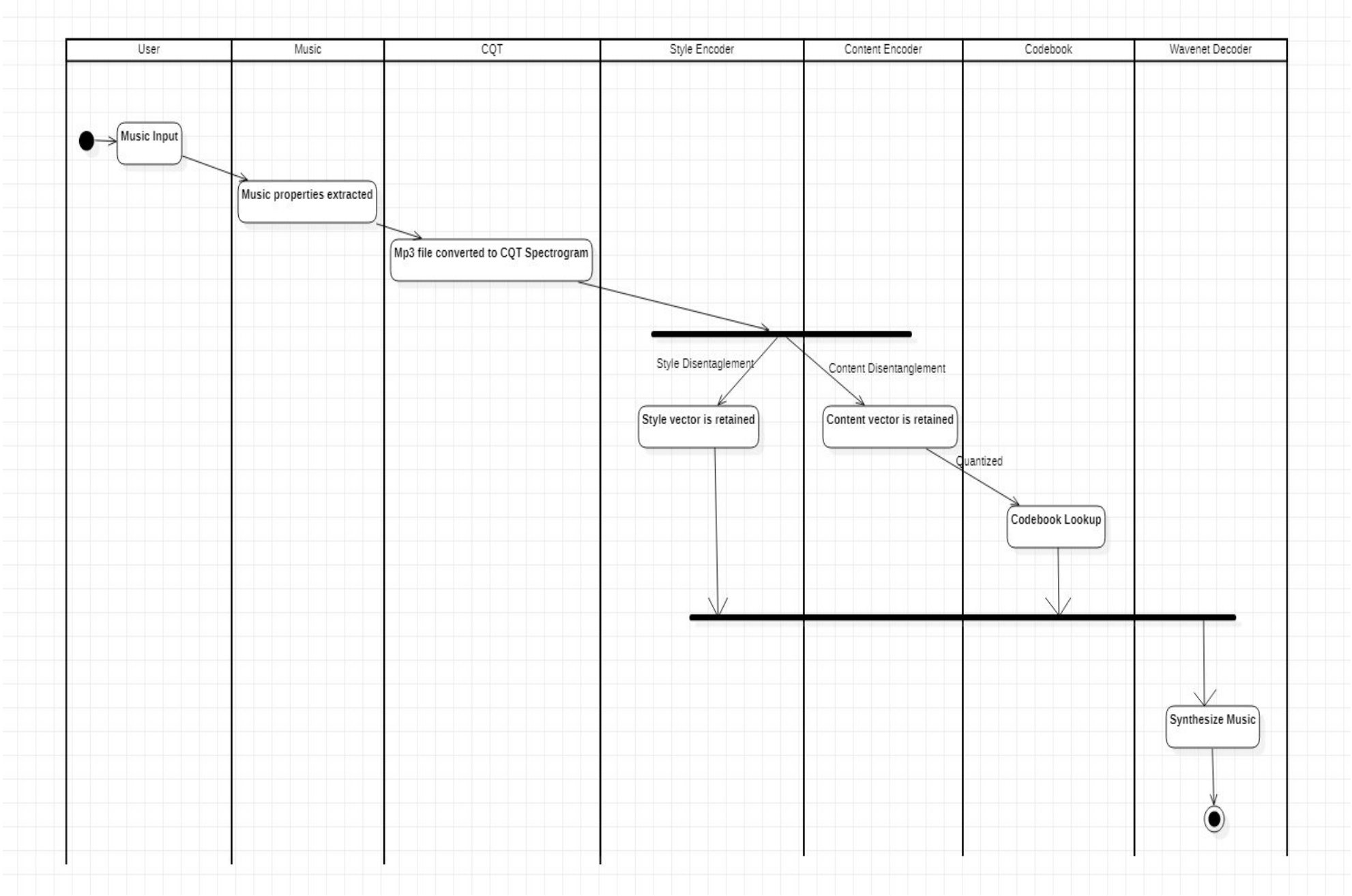
Use Case Diagram



Sequence Diagram



Swimlane Diagram



Technologies Used

1. Operating System : Since the project will be developed using Google Colab, the project is interoperable with any OS.

2. Tools Libraries : Tensorflow, Keras, PyTorch and other Python modules

Project Progress

- Project progress so far: Working on creating the Data set
- Percentage completion of the project: 30%

Capstone (Phase-I & Phase-II) Project Timeline

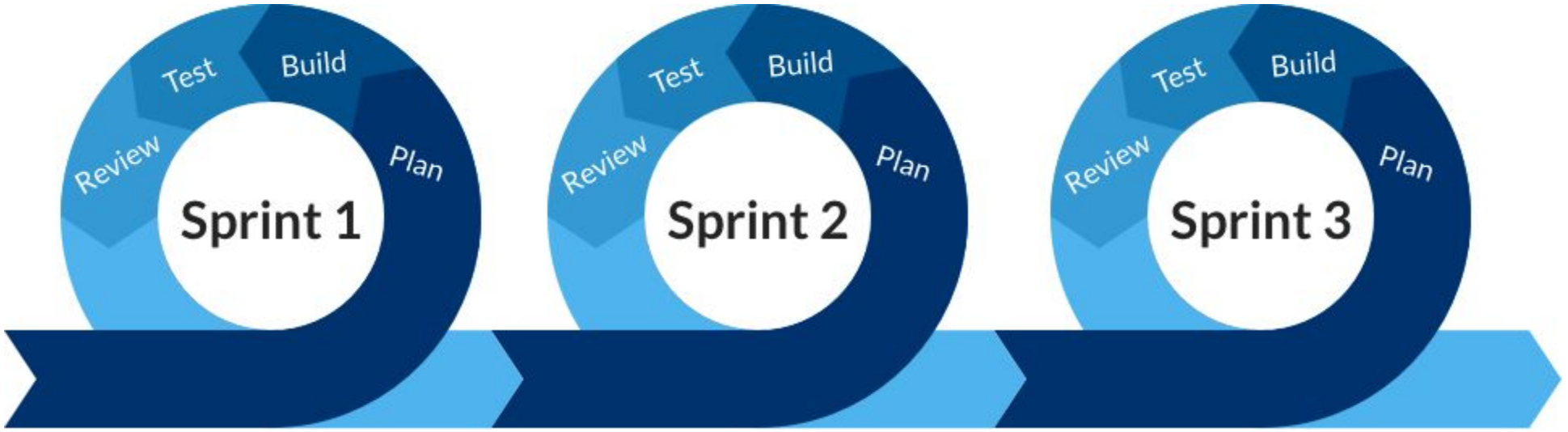
- Capstone-I deliverables
  - Project requirements to be identified
  - Provide the high level design of the project
  - Create the dataset

- Capstone-II deliverables

  - Train the model on a new artist’s voice
  - Maintain the emotion/feel of the song
  - Preserve the musical structure of the song
  - Build an interactive system of the model
  - Enhance and optimizing various aspects of the model

Capstone (Phase-I & Phase-II) Project Timeline

- Follow iterative model
- Deploy every 2 weeks
- Follow SDLC every iteration
- Apply ML model to learn a singers style by training on a dataset
- Apply singers' style to new songs



Any other information

Conclusion

1. The project aims on imposing a style of a singer onto a song. Given the singer(style) and the song(content) from the user, the song will be generated as if it had been sung in that singers style

2. Worked on midi files, separated vocals from instrumental and looked at different methods found that VQ-VAE model works the best for style transfer.

3. Plan to implement the VQ-VAE model by making use of 2 encoders (style and content) and a Wavenet decoder

References

[1]O. Cífka, A. Ozerov, U. Şimşekli and G. Richard, "Self-Supervised VQ-VAE for One-Shot Music Style Transfer," ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 96-100, doi: 10.1109/ICASSP39728.2021.9414235.

[2]J. Wang, C. Jin, W. Zhao, S. Liu and X. Lv, "An Unsupervised Methodology for Musical Style Translation," 2019 15th International Conference on Computational Intelligence and Security (CIS), 2019, pp. 216-220, doi: 10.1109/CIS.2019.00053.

[3]Y. -Q. Lim, C. S. Chan and F. Y. Loo, "ClaviNet: Generate Music With Different Musical Styles," in IEEE MultiMedia, vol. 28, no. 1, pp. 83-93, 1 Jan.-March 2021, doi: 10.1109/MMUL.2020.3046491.

[4]C. -F. Huang and C. -Y. Huang, "Emotion-based AI Music Generation System with CVAE-GAN," 2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), 2020, pp. 220-222, doi: 10.1109/ECICE50847.2020.9301934.

[5]Oore, S., Simon, I., Dieleman, S. et al. This time with feeling: learning expressive musical performance. Neural Comput & Applic 32, 955–967 (2020). https://doi.org/10.1007/s00521-018-3758-9

[6]R. Lang, S. Wu, S. Zhu and Z. Li, "SSCL: Music Generation in Long-term with Cluster Learning," 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2020, pp. 77-81, doi: 10.1109/ITNEC48623.2020.9085207.

[7]en-Yu Liu, Yu-Hua Chen, Yin-Cheng Yeh, Yi-Hsuan Yang, "Score and Lyrics-Free Singing Voice Generation," International Conference on Computational Creativity (ICCC), 2020, doi: 10.48550/arXiv.1912.11747

[8]M. Ragesh Rajan, "Singing Voice Synthesis System for Carnatic Music," 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), 2018, pp. 831-835, doi: 10.1109/SPIN.2018.8474033.

[9]Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, Ilya Sutskever, "Jukebox: A Generative Model for Music," 2020, doi: 10.48550/arXiv.2005.00341

[10]Nachmani, E. and Wolf, L., 2019. Unsupervised singing voice conversion. arXiv preprint arXiv:1904.06590.

[11]Blaauw, M., Bonada, J. and Daido, R., 2019, May. Data efficient voice cloning for neural singing synthesis. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6840-6844). IEEE.

[12]Daher, R., Zein, M.K., El Zini, J., Awad, M. and Asmar, D., 2020, July. Change your singer: a transfer learning generative adversarial framework for song to song conversion. In 2020 International Joint Conference on Neural Networks (IJCNN) (pp. 1-7). IEEE.

Thank You