INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER

The University of Westminster, Coat of Arms

Emotion-Aware Music Recommendation with ConvNeXt V2"

Thesis by

Ms.Hasni Haleemdeen

W1898945 | 20211337

Supervised by

Mr. Guhanathan Poravi

July 2025

Submitted in partial fulfillment of the requirements for the BEng (Hons) Software Engineering degree at the University of Westminster.

ABSTRACT

In the age of intelligent media consumption, the ability to recommend music that aligns with a listener's emotional state is becoming increasingly relevant. While traditional music recommendation engines rely heavily on historical data and user preferences, they often fail to consider the user's real-time emotional context. This research presents a novel system that integrates ConvNeXt V2, a modern convolutional neural network architecture with the Spotify API to deliver personalized music recommendations based on facial emotion recognition. Unlike vision transformers that demand heavy computation, ConvNeXt V2 maintains transformer-inspired design principles while retaining the efficiency of convolutional networks, making it ideal for real-time inference on mobile or web platforms.

The system processes live or uploaded facial images using robust preprocessing and augmentation techniques, then classifies the user's emotional state using a ConvNeXt V2 model trained on benchmark datasets such as FER-2013. Once classified, the detected emotion is mapped to curated playlists via Spotify's music catalog. Experimental evaluation showed strong performance, with the model achieving a validation accuracy of **90.09%**, and detailed performance was confirmed through metrics such as precision, recall, F1-score, and confusion matrices. These results underscore the model's ability to generalize across diverse facial expressions and lighting conditions while maintaining responsiveness.

By bridging facial affective computing with intelligent music delivery, this system offers a more emotionally engaging and context-aware user experience. It demonstrates the feasibility of deploying ConvNeXt V2 as a lightweight, scalable solution for enhancing digital personalization in entertainment applications.

Keywords: ConvNeXt V2, Facial Emotion Recognition, Music Recommendation, Deep Learning, Computer Vision, Spotify API, Real-time Inference, Human-Centered AI

Subject Descriptors:

Computing methodologies → Convolutional Neural Networks → Computer Vision → Emotion-Aware Systems → Personalized Media Recommendations

DECLARATION

I hereby declare that the content presented in this thesis, titled *"Emotion-Aware Music Recommendation with ConvNeXt V2,"* is the result of my own independent work, carried out at the Informatics Institute of Technology under the guidance of my supervisor, Ms. Hasni Haleemdeen.

This thesis has not been submitted, either in part or in full, to any other university or institution for the purpose of obtaining an academic degree or professional qualification. All sources of information used in the preparation of this work have been properly cited and acknowledged in accordance with academic conventions.

Date: 11-07-2025

Signature:

Name: Hasni Haleemdeen

ACKNOWLEDGEMENTS

I would like to extend my sincere gratitude to all those who supported me throughout the course of this project.

First and foremost, I am deeply thankful to my supervisor, **Mr. Guhanathan Poravi**, for his invaluable guidance, continuous encouragement, and insightful feedback. His expert advice and motivating support were instrumental in shaping the direction and execution of this research.

I would also like to express my appreciation to the academic and administrative staff of the **Informatics Institute of Technology** for providing the resources, facilities, and a conducive learning environment that enabled me to complete this work effectively.

To everyone who contributed in any way directly or indirectly toward the completion of this thesis, your support is truly appreciated

Hasni Haleemdeen

Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Definition
AI	**Artificial Intelligence**
ViT	**Vision Transformer**
CNN	**Convolutional Neural Network**
JSON	**JavaScript Object Notation**
API	**Application Programming Interface**
GPU	**Graphics Processing Unit**
UI	**User Interface**
UX	**User Experience**
IDE	**Integrated Development Environment**
FER	**Facial Emotion Recognition**
FER2013	**Facial Expression Recognition 2013 Dataset**
DNN	**Deep Neural Network**
HCI	**Human-Computer Interaction**
NLP	**Natural Language Processing**
SOTA	**State of the Art**
DL	**Deep Learning**
ML	**Machine Learning**
SVM	**Support Vector Machine**
DFD	**Data Flow Diagram**

SLEP	**Social, Legal, Ethical and Professional**
API Key	**Application Programming Interface Key**
IoT	**Internet of Things**

CHAPTER 1: INTRODUCTION

1.1 Chapter overview

This chapter explains why we are creating an emotion-aware music recommendation system with ConvNeXt V2. It points out the flaws in traditional recommendation engines that ignore real-time emotional context. This oversight leads to less personalized results. The discussion emphasizes our goal to close this gap with an effective deep learning model that can recognize facial emotions and map them to music. It also outlines the study's direction by stating key objectives, challenges faced in modeling dynamic emotions, and reasons for choosing ConvNeXt V2 as the main framework for real-time, user-focused applications.

1.2 Problem background

1.2.1 Emotion-aware recommendation gaps

In the age of personalized digital experiences, music recommendation engines have made significant progress. However, a user's emotional state, which plays a crucial role in listening choices, is often ignored. Traditional models rely heavily on static user data, such as past listening habits and genre preferences, and do not account for the dynamic changes in mood (Tran et al., 2023; Singh and Dembla, 2023). Since emotional contexts are personal and often influenced by temporary situations, recommendations based solely on historical behavior fail to provide truly adaptive and satisfying listening experiences. This lack of emotional awareness reveals a major gap in how current recommendation systems are designed (Hung et al., 2021; Huang et al., 2024).

1.2.2 Deep learning for facial emotion interpretation

Facial expressions provide a clear and immediate glimpse into a person’s emotional state, but capturing and interpreting them in real-time has usually been costly in terms of computing power. Recent advancements in computer vision, particularly through deep learning, have made it possible to decode emotional signals effectively (Nan et al., 2025). ConvNeXt V2, with its improved convolutional structure and ability to scale, is well-suited for high-accuracy facial emotion recognition even in settings with limited resources (Nan et al., 2025). Unlike earlier models that struggled to generalize under varying lighting and obstructions, ConvNeXt

V2 delivers strong performance without the high training and inference costs seen in transformer-based models (Nan et al., 2025).

1.3 Problem definition

Current music recommendation systems track user behavior but do not effectively adjust to users' emotional states. These systems rely on static factors like listening history or genre preferences, leading to generic results that often miss the emotional connection the user seeks (Hung et al., 2021; Huang et al., 2024). As affective computing grows in importance, we need models that can interpret human emotions in real time without sacrificing speed or accuracy (Nan et al., 2025).

ConvNeXt V2 offers a promising way to fill this gap with a lightweight convolutional framework that maintains strong expressive power (Nan et al., 2025). Unlike transformer-based models that require many resources, ConvNeXt V2 enables quick inference and easy deployment. This makes it suitable for real-time emotion recognition in everyday situations. However, challenges still exist in ensuring the model keeps high accuracy across different facial expressions, lighting, and user demographics, particularly when precise emotional distinctions are necessary (Nan et al., 2025).

1.3.1 Problem statement

The goal is to create an intelligent, real-time facial emotion recognition system using ConvNeXt V2 that improves music recommendation accuracy through emotion-driven personalization. This will help connect user mood with digital content delivery. The system seeks to achieve emotional alignment with minimal computational demands, ensuring it can scale and respond well in everyday settings.

1.4 Research motivation

This research responds to the growing need for emotionally intelligent systems that offer more than just basic personalization. Traditional music recommendation models do not adjust to real-time emotional changes and often miss temporary user moods. With ConvNeXt V2 providing a lightweight and effective architecture, there is a clear reason to include emotion

recognition in standard digital experiences without high computing costs. These systems have the potential to improve user engagement and support context-aware interactions that promote digital well-being. By connecting emotional signals with content delivery, this research seeks to contribute to the field of emotion-aware computing with scalable and accessible solutions.

1.5 Existing work

Citation	Summary	Limitation	Contribution
Bingyu Nan et al. (2025)	This work introduces ConvNeXt V2, a redesigned CNN that competes with ViTs in accuracy while staying lightweight. With LayerNorm, GELU, and large kernels, it provides real-time efficiency for tasks like facial emotion recognition.	High performance relies on correct hyperparameter tuning. It does not include audio or multimodal integration.	Introduced ConvNeXt V2 as an efficient option for image and facial emotion recognition. It offers lower latency and high accuracy.
Hina Fatima et al. (2023)	This study introduces a CNN-based facial emotion recognition system that works with the Spotify API for music recommendations. While it performs well in static situations, the model is not efficient enough for real-time use. ConvNeXt V2 can solve this issue.	Used static emotion detection with no real-time changes or personalization.	Focused on emotion-based music recommendations using emotion labels from FER2013 and music features. Combined CNN emotion output with metadata to recommend songs.
Mohiuddin et al. (2023)	Mohiuddin's paper presents a fusion-based architecture for recognizing emotions using	Complex architecture raises computational	Combined facial expression recognition with

	different methods. It achieves strong results in detecting emotions; however, the added complexity of the model makes it difficult to deploy on edge devices.	costs and is hard to deploy in lightweight settings.	speech emotion to improve mood analysis accuracy.
Xinyi Xu et al. (2023)	The paper presents a GAN framework to improve FER datasets by adding more diverse and realistic samples. The method increases accuracy on smaller datasets by addressing class imbalance, which is a common problem in emotion datasets like FER2013.	Used generative adversarial networks to create synthetic facial expressions and balance class distribution for better training.	Not connected with downstream applications like music recommendation. It mainly focuses on improving data.
Ghosh et al. (2022)	Ghosh and colleagues examine emotion classification using EEG signals that are processed through convolutional and recurrent layers. Even though it does not involve direct visuals, their method highlights the importance of understanding emotions through multiple modes in smart systems.	Invasive input method. It is not practical for real-world media recommendations.	Investigated emotion recognition using EEG signals with CNN and RNN layers. This research provided insights into detecting feelings based on brainwaves.
Hung et al. (2021)	This paper presents the EMOPIA dataset, which links facial expressions to musical mood. It shows how useful it is to combine facial and musical elements for making	Requires a large amount of training data. The connection between facial data and musical data is	Created a unique dataset that connects music with facial expressions. Showed that

	recommendations based on emotions. This supports the need for a multimodal approach in systems that understand emotions.	not always consistent.	training with multiple modes can improve emotion-aware music recommendations.
Krishna Kumar Singh & Payal Dembla (2023)	This study looks at CNN and RNN-based models for mood-based recommendations. It stresses the need for preprocessing audio and metadata features. The findings show that even though mapping emotions to music is possible, traditional models have difficulty with real-time responsiveness and personalization.	Limited ability to apply to new datasets and does not work in real-time.	Proposed a modular framework to classify facial emotions using handcrafted features and a CNN ensemble. This approach improves classification on FER-2013.

Table 1 - Existing Works

1.6 Research gap

After reviewing existing literature on facial emotion recognition and emotion-aware recommendation systems, several important limitations have emerged. These highlight the need for additional research and development. Here are the main points:

Limited accuracy and generalizability of facial emotion recognition models in diverse, real-world conditions - Despite advancements in deep learning models like ConvNeXt V2 for visual understanding tasks, current systems often lack reliability when used outside controlled environments, such as FER2013 or AffectNet (Nan et al., 2023). These models have difficulty detecting subtle emotional cues and do not perform consistently under changes in lighting, facial orientation, and occlusion. This leads to incorrect emotion classification, especially in real-time user environments, reducing the emotional impact of generated recommendations (Ghosh et al., 2022).

Insufficient integration of real-time emotion recognition with dynamic music recommendation systems - Current music recommendation engines mainly depend on fixed user preferences or listening history. They do not respond well to changes in the listener's emotional state. Additionally, systems that try to use real-time integration with APIs like Spotify often struggle with issues related to latency and contextual alignment (Sharma et al., 2021). There is a clear need for lightweight yet accurate systems, like those based on ConvNeXt V2, that can connect facial emotion recognition with emotion-sensitive recommendation engines while remaining responsive and personalized.

1.7 Contribution to the body of knowledge

1.7.1 Contribution to the research domain

Redefining the application boundary of ConvNeXt V2 for emotion-centric systems - This research expands the use of ConvNeXt V2 beyond traditional classification tasks. It shows that the model is suitable for real-time emotion computing. By applying ConvNeXt V2 to facial emotion recognition, the model now works in the area of emotion-aware systems. It can identify subtle emotional changes with high accuracy and efficiency. This positions ConvNeXt V2 as a practical alternative to transformer-based models in lightweight, real-time environments (Nan et al., 2023).

Benchmarking ConvNeXt V2 performance under real-world visual variations -Through extensive experiments on FER-2013 and other emotion datasets, this work assesses how well ConvNeXt V2 holds up when faced with challenges like occlusion, lighting changes, and different facial expressions. These tests add to the knowledge base by providing reliable measures for future research on real-time facial emotion classification models in uncontrolled environments (Ghosh et al., 2022).

Establishing a deep learning pipeline linking emotion detection with adaptive music recommendation - This study presents a combined pipeline where facial emotion detection powered by ConvNeXt V2 connects to dynamic content delivery through APIs like Spotify. This contribution demonstrates the possibility of merging facial recognition with real-time recommendation systems. It highlights a new way to let emotional signals directly influence digital media experiences (Sharma et al., 2021).

1.7.2 Contribution to Problem Domain (Emotion-Aware Recommender Systems)

Enhancing real-time emotional intelligence in recommender systems - By using facial cues to guide music recommendations, this research changes the approach from passive, history-based systems to active, real-time personalization. The proposed system detects the user's emotions in real time and generates relevant suggestions. This gives users a more emotionally connected experience with their digital music.

Introducing an emotion-adaptive recommendation framework using ConvNeXt V2 - The integration of ConvNeXt V2 enables accurate and fast emotion detection, making real-time feedback possible in digital entertainment systems. This allows for personalization that adjusts with the user's changing emotions, improving user satisfaction and engagement on music streaming platforms.

Encouraging empathetic design in digital systems - Beyond technical contributions, this research highlights the ethical and social importance of emotion-aware technologies. By focusing on mental well-being in user-system interactions, this work advocates for emotionally attuned computing as a key element of user-centered digital system design (Gorasiya et al., 2022).

1.8 Research challenge

The development of an emotion-aware music recommendation system using ConvNeXt V2 involves several technical and practical challenges that must be overcome for it to succeed in real-world settings. These challenges range from issues with the model itself to constraints during deployment, as detailed below:

Real-time accuracy in emotion recognition across dynamic user expressions - A major challenge is to ensure consistent and accurate classification of facial emotions in real-time. Even though ConvNeXt V2 includes improvements like large kernel depthwise convolutions and normalized activations (Nan et al., 2023), recognizing subtle emotional expressions remains a challenge, particularly in uncontrolled environments with obstructions, varying lighting, or different facial structures. Misclassifying emotions can negatively impact how relevant the music recommendations are, which reduces user engagement and overall experience.

Optimizing ConvNeXt V2 for resource-constrained environments - While ConvNeXt V2 is lighter than Transformer-based models, it still has a large number of parameters, creating deployment issues on mobile devices or web browsers. Striking a balance between computational efficiency and model accuracy is a significant challenge, especially when trying to achieve low-latency predictions suitable for real-time applications (Liu et al., 2022). Simplifying the architecture, using pruning techniques, and employing quantization strategies are critical to maintaining inference speed without sacrificing performance.

Ensuring model robustness against facial and environmental variability - Another significant challenge arises from the variability of real-world input conditions. ConvNeXt V2 must be resilient to inconsistent lighting, various ethnic facial features, camera resolutions, and background noise. To effectively generalize across different input conditions, it is essential to use diverse datasets and strong data augmentation techniques that include brightness adjustments, occlusion simulations, and facial angle rotations (Ghosh et al., 2022).

Synchronizing emotion recognition with external music recommendation APIs - Connecting the output of ConvNeXt V2 with real-time music suggestion services like the Spotify API adds another layer of complexity. The system must accurately translate detected emotional states into tailored song recommendations that fit the genre and mood. This requires strong logic for mapping emotions to music, minimal response delays, and API resilience to changing emotional inputs and user behavior patterns (Gorasiya et al., 2022).

1.9 Research questions

RQ1 - How can ConvNeXt V2 be used to detect and classify real-time facial emotions for music recommendation systems?

RQ2 - What methods can be used to improve the classification accuracy and reliability of ConvNeXt V2 in emotion recognition tasks?

RQ3 - How do real-time emotion-aware music recommendations, powered by ConvNeXt V2, affect user engagement and personalization on streaming platforms?

RQ4 - What key factors affect the scalability and flexibility of ConvNeXt V2-based emotion recognition systems in real-world situations?

1.10 Research aim

This research aims to design, develop, and evaluate an emotion-aware music recommendation system that leverages the ConvNeXt V2 architecture for accurate and efficient real-time facial emotion recognition. The objective is to bridge affective computing with intelligent music

personalization, allowing user emotions captured via visual cues to dynamically influence music playback.

To achieve this, the study will implement a facial expression recognition pipeline using ConvNeXt V2, known for its large-kernel depth wise convolutions and lightweight design (Nan et al., 2023), and integrate it with music recommendation services such as Spotify. The system will be evaluated on its ability to detect emotions with high precision under varying real-world conditions, assess its responsiveness in matching emotional context with appropriate music, and measure the computational efficiency to support deployment on everyday devices.

The overarching goal is to position ConvNeXt V2 as a practical solution in affective human-computer interaction, expanding its utility beyond visual classification into emotion-responsive personalization in the digital media space.

1.11 Research objectives

Research Objectives	**Explanation**	**Learning Outcome**	**Research questions**
Problem Identification	**RO1:** To assess how well ConvNeXt V2 works for real-time emotion recognition in music recommendation tasks. **RO2:** To find the technical and environmental factors that influence the scalability and reliability of facial emotion detection systems.	**LO1** **LO4**	**RQ1** **RQ4**
Literature Review	**RO3:** To critically examine current deep learning strategies in emotion-aware music systems, focusing on convolutional models and their limitations. **RO4:** To compare ConvNeXt V2 to Vision Transformers regarding real-time emotion classification. **RO5**: To examine how data diversity, preprocessing, and augmentation techniques improve the reliability of facial emotion recognition models, particularly in real-world situations.	**LO1** **LO4** **LO5**	**RQ1** **RQ2**

Requirement Analysis	**RO6:** To define the data needed for training and validating ConvNeXt V2 for emotion-aware music recommendation, including relevant facial emotion datasets and music metadata. **RO7:** To identify the necessary hardware and software for supporting real-time facial emotion processing. **RO8:** To gather and examine user preferences to ensure the system design stays intuitive, responsive, and meets expectations for emotion-based personalization.	**LO1** **LO6**	**RQ1** **RQ2** **RQ3** **RQ4**

Design	**RO9:** To design the system's layout, we will outline how the emotion recognition model (ConvNeXt V2), real-time input methods, and music recommendation modules connect. **RO10:** To plan and test the system's ability to scale and perform under various scenarios, including mobile responsiveness, lighting conditions, and facial diversity. **RO11:** To create an interactive and user-friendly interface that allows for easy integration of emotion detection and music playback. **RO12:** To include ethical design principles that ensure data privacy, user consent, and responsible AI use throughout the system's user experience and emotion recognition process.	**LO1** **LO2** **LO8** **LO5**	**RQ1** **RQ2** **RQ3** **RQ4**
Implementation	**RO13:** To specify and simplify the computational workflows needed for real-time facial emotion recognition, ensuring the best synchronization with the music recommendation engine. **RO14:** To adjust ConvNeXt V2's hyperparameters, including kernel size, learning rate, and batch size, to improve accuracy in emotion inference and system responsiveness across different input conditions (Liu et al., 2023).	**LO1** **LO2** **LO5** **LO7**	**RQ3**

Testing and Evaluation	**RO15:** To measure user satisfaction and engagement with the integrated system using empirical metrics like latency, prediction accuracy, and feedback surveys. **RO16:** To test the strength of the emotion-aware music recommendation system in different environments, such as lighting and facial angles, as well as across diverse demographic groups. **RO17:** To create feedback methods that help improve the system and ensure it remains useful over time in different contexts...	**LO1** **LO4** **LO5** **LO8**	**RQ1** **RQ2** **RQ3** **RQ4**

Table 2 - Research Objectives

1.12 Chapter summary

This chapter described the research problem, motivation, and goal of creating an emotion-aware music recommendation system using ConvNeXt V2. It pointed out the limitations of traditional recommendation methods and presented ConvNeXt V2 as a good fit for real-time emotion recognition. It also discussed key contributions and challenges, connecting the study to the intended objectives and BEng (Hons) Software Engineering learning outcomes. This lays the groundwork for the upcoming implementation and evaluation phases.

CHAPTER 2: LITERATURE REVIEW

2.1. Chapter overview

This chapter exhibits the current challenges in emotion-aware music recommendation systems, particularly focusing on the limitations of traditional facial emotion recognition methods and their implications on effective personalization. It critically reviews the advancements in deep learning architectures, with an emphasis on ConvNeXt V2, a modern convolutional model optimized for visual tasks with low latency. Moreover, the chapter evaluates the role of deep

facial analysis in enabling real-time emotional inference and explores the integration of external platforms such as the Spotify API for dynamic music recommendation. Finally, this chapter assesses relevant literature to benchmark ConvNeXt V2 against other frameworks, establishing its suitability for developing responsive, emotion-driven music recommendation systems.

2.2. Concept graph

The concept graph showcases a simple visual summary of the key ideas, technologies, and solutions found in the literature review. It shows how crucial parts like emotion recognition, ConvNeXt V2, and music recommendation are interconnected. This helps to understand the overall system with clarity. The absolute concept map is included in **Appendix A.**

2.3. Problem domain

2.3.1 Real-time facial emotion interpretation for music engagement

The emotional state of an individual remarkably impacts their musical preferences and listening experiences. Whereas traditional music recommendation systems mainly rely on behavioral data pattern like user ratings or genre history, they often ignore the user's real-time emotional context. This leads to a gap between the listener's current mood and the music suggestions given. Real-time facial expression analysis provides a more direct and immediate way to understand emotional states, creating a chance to bridge this gap effectively.

Facial expressions are seen as one of the most genuine ways to communicate emotions, yet capturing and interpreting them accurately in changing environments is still a technical challenge. Particularly, using deep learning architecture that can process and interpret subtle facial cues in real-time without delays has become important. ConvNeXt V2, a modern convolutional architecture, is notable for its balance between high accuracy and computational efficiency. This makes it a good fit for real-time affective computing systems (Nan et al., 2025).

Furthermore, current solutions often face struggles in balancing performance and speed. Larger Transformer-based models can achieve impressive recognition accuracy, but they may sacrifice inference speed and scalability in low-resource environments like mobile platforms. ConvNeXt V2 addresses this issue by delivering Transformer-level performance while

maintaining the lightweight features needed for real-world applications like music streaming services (Liu et al., 2022).

Incorporating ConvNeXt V2 into emotion-aware music recommendation systems tackles a vital problem: developing intelligent systems that do not only accurately detect user emotions but also react in real-time with personalized music selections tailored to their mood. This integration promotes deeper user engagement, emotional connection, and a more understanding human-AI interaction. Hence this approach is increasingly becoming the norm in personalized digital experiences (Hung et al., 2021).

2.3.2 Challenges in emotion-based recommendation systems

The combination of emotion recognition and music recommendation systems offers a new way to boost user engagement. However, this area has complex challenges that involve both technology and psychology. One major concern is the natural subjectivity of human emotion. Unlike sensor data that can be measured, emotional expressions differ greatly depending on cultural backgrounds, personal experiences, and specific situations (Liu et al., 2022). This variation makes it difficult to standardize how emotions are interpreted, creating a challenge for creating universally accurate recommendation systems.

Facial expression recognition is globally accepted for its non-intrusiveness, but it has significant limitations in real-world settings. Factors like occlusion, head tilt, inconsistent lighting, and diversity in facial features lower the accuracy of classifiers. Although datasets like FER-2013 and RAF-DB exist, many do not cover spontaneous or diverse facial expressions, which makes them less proficient in real-world applications (Nan, Zhao and Zhang, 2025).

Henceforth, linking emotional states to relevant musical outputs adds another layer of complexity. APIs like Spotify provide vast music libraries. However, ensuring that recommended tracks match nuanced emotional tones, like melancholy and serene sadness, is a challenging problem. Features such as tempo, lyrics, genre, and acoustic intensity must be interpreted within context to prevent emotional mismatches. An incorrect pairing, such as suggesting a cheerful song for someone showing distress, can disrupt immersion and undermine trust in the system (Hung et al., 2021).

Another key challenge is maintaining real-time performance. Unlike traditional recommendation systems that work in batch modes, emotion-aware systems need to detect emotions and deliver recommendations instantly. This demands optimized inference pipelines, especially for mobile or edge devices. ConvNeXt V2, with its design improvements and

efficient computation, shows promise in balancing accuracy and speed, yet integrating with cloud-based services like Spotify introduces additional structural limits (Nan, Zhao and Zhang, 2025).

Eventually, the ethical issues around privacy and data sensitivity are crucial. Systems that rely on facial expressions inherently manage biometric data, making them vulnerable to misuse or breaches. Effective anonymization, on-device processing, and clear data handling policies are essential to build trust and comply with privacy regulations (Panlima and Sukvichai, 2023).

2.3.3 ConvNeXt V2 as a modern solution

With the increasing demand for real-time emotion-aware applications, conventional convolutional neural networks (CNNs) and Vision Transformers (ViTs) have faced tough challenges in balancing efficiency, scalability, and performance. Traditional CNNs are convenient in detecting local features but often struggle with capturing global dependencies. They can also experience overfitting when dealing with unbalanced emotional datasets (Mahapatra and Singh, 2022). In contrast, ViTs offer better global context awareness but are resource-intensive, require large datasets, and can cause delays that limit their use in real-time systems (Nan, Zhao and Zhang, 2025).

ConvNeXt V2 presents a convincing solution by combining the strengths of CNNs with the benefits of ViTs. Its design features modern elements such as depthwise convolutions, Layer Normalization, and GELU activation. These components work together to improve learning efficiency and generalization (Nan, Zhao and Zhang, 2025). The architecture demonstrates strong performance on standard facial emotion recognition benchmarks such as FERPlus and RAF-DB, showing its ability to distinguish subtle expressions in various conditions (Hung et al., 2021).

Unlike Transformer-based models, which usually requires a lot of GPU memory and large training datasets, ConvNeXt V2 is computationally cost-effective. This makes it suitable for use in edge environments or applications that need real-time processing, like music recommendation systems based on facial emotion recognition. Additionally, ConvNeXt V2's streamlined design makes it easier to integrate into emotion-based user interfaces, where minimizing latency, energy consumption, and maximizing prediction accuracy are essential (Liu et al., 2022).

2.3.4 ConvNeXt V2: bridging CNNs and transformers

The rapid evolution of computer vision has encountered ongoing competition between convolutional neural networks (CNNs) and Vision Transformers (ViTs). Each offers unique benefits and drawbacks. CNNs are well known for efficiently extracting local features, but they struggle with modeling long-range dependencies. On the other hand, ViTs excel at capturing global contextual information using attention mechanisms, though they require substantial computational resources and large amounts of data (Hung et al., 2021).

ConvNeXt V2 represents a thoughtful combination of these two approaches. Drawing inspiration from the architectural principles of ViTs, it redefines CNNs by adding modern features like depthwise separable convolutions, large kernel sizes, GELU activation, and Layer Normalization. It retains the efficiency and scalability that traditional CNNs provide (Nan, Zhao, and Zhang, 2025). This makes ConvNeXt V2 particularly well-suited for applications that need real-time responsiveness, such as music recommendation systems driven by facial emotions.

Empirical evidence supports the effectiveness of ConvNeXt V2. In benchmark tests on datasets like FERPlus and RAF-DB, ConvNeXt variants have shown higher accuracy and generalizability than both conventional CNNs and lighter ViT models (Liu et al., 2022; Mahapatra and Singh, 2022). Its self-attention-like mechanisms enable the model to focus on expressive facial regions, enhancing recognition accuracy for subtle or complex emotions like surprise, fear, and disgust.

Moreover, ConvNeXt V2's lightweight design is ideal for mobile and edge deployments. This is essential for music recommendation systems that need to operate efficiently on user devices without relying on cloud processing. Unlike ViTs, which require significant computational power and extensive training data, ConvNeXt V2 delivers high performance with relatively modest resources. This makes it a bridge between accuracy and accessibility.

In summary, ConvNeXt V2 carries over the strengths of CNNs while incorporating modern improvements inspired by transformer models. This hybrid quality positions it as a strong candidate for powering intelligent, real-time, emotion-aware interfaces, including music recommendation engines that respond dynamically to user moods.

2.3.5 Specific challenges in applying ConvNeXt V2 to emotion recognition

While ConvNeXt V2 has become a strong backbone architecture in computer vision, using it in emotion-aware systems for real-time music recommendation has clear limitations. These challenges come from the complexity of human emotions and the practical constraints of real-world applications.

One major challenge is the subtlety and ambiguity of emotional expressions. Emotions often show up as micro-expressions, which are small and fleeting changes in facial muscles that are hard to capture. Even with ConvNeXt V2's innovative convolutional design and high-resolution feature extraction, the system may misclassify emotions when users express suppressed or overlapping feelings (Nan, Zhao and Zhang, 2025). This misclassification can directly impact the quality of music recommendations and the user experience

Another limitation is the narrow generalizability of the datasets used to train emotion recognition models. While ConvNeXt V2 performs well on standardized datasets like FER-2013 and RAF-DB, these datasets do not fully represent the diversity found in real-world settings. This includes cultural differences, lighting variations, and changes in poses (Liu et al., 2022). As a result, the model may perform poorly in uncontrolled or culturally diverse environments.

Another limitation is the narrow generalizability of the datasets used to train emotion recognition models. While ConvNeXt V2 outperforms well on standardized datasets like FER-2013 and RAF-DB, these datasets do not fully represent the diversity found in real-world settings. This includes cultural differences, lighting variations, and changes in poses (Liu et al., 2022). As a result, the model may perform poorly in uncontrolled or culturally diverse environments.

Overall, even though ConvNeXt V2 is relatively lightweight, its computational demands are still significant for edge deployment. Real-time facial analysis on mobile or wearable devices needs models that are accurate and also designed to minimize latency and memory use. ConvNeXt V2 may need extra pruning, quantization, or distillation techniques to work within these resource limits without losing accuracy (Panlima and Sukvichai, 2023).

Furthermore, emotion-aware music recommendation requires more than just vision-based analysis. Users' emotional states are shaped by several signals, including speech tone, body language, and contextual information like the time of day or past activities. ConvNeXt V2 is

inherently unimodal and cannot integrate these contextual or multimodal inputs, which limits its ability to fully understand the user's emotional state in complex situations (Hung et al., 2021).

2.4. Existing work

Emotion-aware music recommendation has become more in trend in recent years. Several studies have over-looked into systems that recognize facial expressions to improve user experience. Liu et al. (2022) highlighted how deep learning can effectively detect emotional states. Meanwhile, Nan, Zhao, and Zhang (2025) presented ConvNeXt V2 for efficient and scalable recognition. The existing research in this area can be generally divided into vision-based emotion detection and hybrid multimodal recommendation systems.

2.4.1 Emotion recognition approaches in music recommender systems

Emotion-aware music recommendation systems use various recognition techniques to detect users' emotional states and match them with suitable music. These techniques create a dynamic listening experience that changes with the user's mood in real time.

2.4.1.1 Facial Emotion Recognition (FER)

Using computer vision and convolutional neural networks, systems like ConvNeXt V2 can interpret subtle facial cues such as eyebrow movement, lip shape, and gaze direction to determine emotional states (Nan, Zhao and Zhang, 2025). This non-intrusive method is one of the most direct ways to detect emotions in media applications.

2.4.1.2 Speech and Vocal Signal Processing

Voice tone, pitch, and speaking pace can also reveal emotion. These features are increasingly used in emotion-adaptive systems, especially when users interact with voice commands (Ghosh et al., 2022).

2.4.1.3 Mood Classification of Music

Deep learning models trained on labeled music datasets can categorize tracks into emotional groups, such as happy, calm, or melancholic, based on acoustic features like tempo, timbre, and rhythm (Hu et al., 2020). This classification is essential for matching songs with identified user emotions.

2.4.1.4 Behavioral and Contextual Cues

Observing user behaviors, such as skipping, replaying, or changing the volume, also helps estimate emotions. Moreover, context like time of day or user location provides indirect but relevant emotional hints (Sharma et al., 2021).

2.4.1.5 Natural Language Processing (NLP)

Text-based sentiment analysis of lyrics, song titles, or user-generated content, such as reviews and comments, can give insight into both the emotional weight of a track and the user mood (Lu et al., 2019).

2.4.1.6 Multimodal and Hybrid Systems

Modern systems combine these techniques into unified processes. For example, ConvNeXt V2 may act as the visual backbone while integrating with contextual and audio models, creating an end-to-end system capable of nuanced, real-time mood interpretation.

2.4.2 Evolution of traditional emotion representation models

The foundation of emotion recognition depends on how emotional states are represented in a computational way. Two main approaches have guided the design of these systems: categorical models and dimensional models.

Categorical approaches, often based on psychological theory, suggest that emotions can be grouped into a limited set of universal classes. Ekman's taxonomy includes six core emotions: joy, anger, fear, sadness, surprise, and disgust. This taxonomy underpins many facial emotion recognition datasets such as FER-2013 and RAF-DB, which are standard in the field (Ekman and Friesen, 1978; Liu et al., 2022). These models provide obvious labels but may not fully capture the entanglement and nuances of human emotions, especially in cases with blended emotions or different cultural expressions.

On the other hand, dimensional models view emotions along continuous scales like valence (positive-negative) and arousal (high-low), as introduced in Russell's Circumplex Model (Russell, 1980). This approach allows systems to represent emotional intensity and variations more smoothly. This fluid representation is crucial for modeling emotional changes over time, especially in areas like music consumption.

However, modern systems are starting to combine both methods. Some hybrid models first detect categorical emotions and then map that output to continuous mood spaces. This connection helps bridge emotional classes and subtle user preferences (Hung et al., 2021). This combination improves personalization in recommendation engines, where fixed emotion labels alone do not adequately tailor auditory content.

Furthermore, creating emotionally annotated datasets usually involves using emotion induction techniques, which employ audio-visual stimuli to provoke genuine emotional responses from participants. While this method works well, it adds variability and subjectivity, which can affect the consistency of model training. As a result, the advancement of these traditional representation models closely relates to data quality and annotation strategies.

2.4.3 Advancements in deep learning and AI-based emotion recognition

Recent advancements in artificial intelligence and deep learning have redefined emotion recognition, moving well beyond traditional rule-based methods. These innovations allow systems to respond more dynamically and intuitively to users' emotional states, especially in affect-aware music recommendations.

2.4.3.1 ConvNeXt V2 CNN Architecture

At the forefront is ConvNeXt V2, a modern convolutional neural network that blends the simplicity of ResNet with the sophistication of Vision Transformers. With large kernel convolutions, GELU activations, and Layer Normalization, this architecture has strong representational power while remaining computationally efficient. It is especially suitable for deployment on mobile or edge devices (Liu et al., 2022; Nan, Zhao and Zhang, 2025).

2.4.3.2 Multimodal Deep Learning

Moving beyond single input types, researchers have adopted multimodal architectures that combine facial expressions with contextual data. This data includes time, user activity, and listening history. Hung et al. (2021) highlighted this approach in their EMOPIA study, showing that emotionally coherent recommendations can arise when facial cues are paired with musical and situational inputs.

2.4.3.3 EEG-Based Emotion Classification

Although this falls outside the direct scope of the project, emotion detection through neurophysiological signals looks promising for more immersive applications. Ghosh et al. (2022) demonstrated that hybrid CNN-RNN models can successfully infer emotional states from EEG data, revealing potential for future integrations in therapeutic music platforms.

2.4.3.4 Vision Transformers (ViTs)

Known for their ability to capture long-range spatial dependencies through self-attention, ViTs are increasingly used in emotion recognition. However, their high memory usage and training needs limit real-time applicability. ConvNeXt V2 addresses these challenges by achieving similar accuracy with improved efficiency (Nan et al., 2025).

2.4.3.5 GAN-Based Emotion Enhancements

Generative Adversarial Networks (GANs), first developed by Goodfellow et al. (2014), have been used for both data augmentation and image purification. In emotion recognition, GANs help reconstruct occluded or noisy facial images, making the system more robust. Recent adaptations, such as the integration of reinforcement learning into GAN frameworks, have enabled more flexible recognition models that learn the best mappings from facial features to emotional states (Nguyen and Jin, 2023). However, issues like convergence latency still limit real-time systems.

2.4.3.6 Defensive and Purification Models

Defense-GAN and MagNet have been adapted from adversarial defense to assist with emotion data preprocessing. VeinGuard, proposed by Li et al. (2023), introduced a Local Transformer-based GAN (LTGAN) with a purifier module. This module preserves essential emotional details while filtering out noise, which could improve emotion detection accuracy in uncontrolled environments.

2.4.3.7 Hybrid Emotion Models

Current research also shows the integration of ConvNeXt V2 with temporal attention mechanisms and audio-based sentiment layers. These hybrid systems can interpret the subtleties of facial micro-expressions while linking them to music dynamics. This results in emotionally synchronized and personalized recommendations.

2.4.4 Applications of ConvNeXt V2 in different fields

ConvNeXt V2 has quickly caught attention in various fields because it effectively combines the strengths of convolutional neural networks with design ideas inspired by Transformers. Its flexibility and performance make it useful for both research and real-world uses. Key applications include:

2.4.4.1 Healthcare and Medical Imaging

ConvNeXt V2 has been used to improve diagnostic accuracy in radiological image analysis, such as CT and MRI scans, by identifying detailed patterns that indicate pathologies. Its lightweight design allows for use in low-resource medical settings (Liu et al., 2022).

2.4.4.2 Human Emotion Analysis

In emotion recognition, ConvNeXt V2 helps with strong facial emotion recognition and shows better accuracy on datasets like FER-2013 and RAF-DB (Nan et al., 2025). This has led to its use in educational tools, workplace monitoring, and therapy platforms.

2.4.4.3 Autonomous Vehicles and Driver Monitoring

The model's ability to recognize small facial cues, such as micro-expressions or signs of eye fatigue, makes it ideal for real-time driver monitoring systems. This can help prevent accidents caused by drowsiness or emotional distraction.

2.4.4.4 Content Personalization and Recommender Systems

ConvNeXt V2 supports intelligent user profiles in entertainment applications, helping to tailor content delivery based on user mood, visual cues, and engagement patterns. This makes it especially useful for platforms like Spotify and YouTube.

2.4.4.5 Robotics and Social Agents

Social robots use ConvNeXt V2 to understand human emotions and adjust interactions. This builds user trust and engagement in service-oriented or assistive robotics.

2.4.4.6 Security and Surveillance

Thanks to its edge-friendly design, ConvNeXt V2 is used in security systems for real-time behavior recognition and anomaly detection, providing both speed and precision.

2.4.4.7 Retail and Customer Analytics

Businesses apply ConvNeXt V2 to analyze in-store camera feeds, interpreting facial reactions to gauge satisfaction, attention, or product appeal. These insights inform marketing strategies and inventory planning.

2.4.5 ConvNeXt V2 in image and emotion recognition

ConvNeXt V2 has become a powerful tool for image understanding and emotion recognition. It builds on CNN basics while addressing issues found in earlier models. Unlike Vision Transformers (ViTs), which depend heavily on global self-attention, ConvNeXt V2 focuses on localized convolutions. It also includes modern features like depth wise separable convolutions and GELU activation. This balance enables the model to efficiently capture both detailed facial features and wider contextual patterns.

In facial emotion recognition, ConvNeXt V2 shows strong performance on datasets like FER-2013 and AffectNet, even under challenging lighting and pose conditions (Liu et al., 2022). Its lightweight design allows it to run on mobile or embedded platforms, which is vital for real-time applications such as emotion-aware music recommendation. Research by Nan et al. (2025) indicates that ConvNeXt V2 delivers competitive performance compared to ViTs while using fewer computing resources. This efficiency makes it suitable for user-focused applications where speed and effectiveness matter.

Moreover, the model's flexibility enables it to fit into multimodal systems. It can combine visual signals with metadata from music APIs or user context. This capability makes ConvNeXt V2 more than just a facial emotion detector; it plays a significant role in comprehensive emotional modeling for intelligent systems. As the need for personalized, emotion-aware services grows in health, media, and education, ConvNeXt V2 provides a reliable and effective foundation for these advancements.

2.4.5.1 The Expanding Reach of ConvNeXt V2 Across Domains

While ConvNeXt V2 has been a pivotal force in facial emotion recognition and music recommendation, its real value emerges when we observe how this architecture is redefining visual computing across industries. What began as an evolution of conventional CNNs has now become a versatile engine powering a wide spectrum of applications each leveraging ConvNeXt V2’s unique balance of precision, efficiency, and adaptability.

In healthcare, for instance, ConvNeXt V2 has proven effective in interpreting complex visual data from radiological scans. Its ability to detect minute anomalies in MRI and CT imagery is assisting medical professionals in making faster, more accurate diagnoses even in resource-limited clinical settings (Liu et al., 2022).

Agriculture is another field benefitting from this innovation. By analyzing drone and satellite imagery, ConvNeXt V2 helps farmers monitor crop health and detect issues like pest

infestations or water stress in real time. This paves the way for smarter, more sustainable farming practices without requiring expensive computing infrastructure on the field.

Autonomous systems, especially in the automotive domain, are turning to ConvNeXt V2 for tasks like real-time navigation and driver monitoring. Its capacity to operate under constrained hardware while maintaining high accuracy allows vehicles to recognize road hazards or detect driver fatigue crucial for ensuring safety in dynamic environments.

In industrial settings, ConvNeXt V2 is being embedded into quality assurance workflows. By scanning product surfaces on production lines, it helps identify manufacturing defects with speed and consistency, reducing human error and wastage. This use case illustrates how the model's visual acuity is being harnessed for tangible economic impact.

The education sector has also started integrating emotion-aware systems based on ConvNeXt V2. E-learning platforms are using real-time emotion detection to tailor content delivery adapting quizzes, videos, or learning paths based on student engagement and facial expressions. This opens up personalized learning experiences that are responsive, empathetic, and effective.

Creative industries aren't left behind either. From AI-powered photo editing apps to animated avatars in virtual environments, ConvNeXt V2 supports tools that adapt visuals based on emotional tone—enabling artists and developers to create content that resonates more deeply with users.

Lastly, in domains like environmental monitoring and urban planning, ConvNeXt V2 has shown promise in analyzing satellite imagery to track changes in land use, vegetation patterns, or pollution levels. Its low-latency processing makes it suitable for continuous, large-scale geospatial data interpretation.

2.4.6 Integration of ConvNeXt V2 in emotion-aware music recommendation systems

The integration of emotion recognition into music recommendation systems has undergone notable advancement, largely driven by recent developments in deep learning particularly the emergence of architectures like ConvNeXt V2, which enable more accurate and context-aware emotional analysis.While traditional recommendation engines have largely relied on behavioral history and content metadata, the incorporation of real-time emotional data especially facial expressions has added a compelling new layer of personalization.

ConvNeXt V2's architecture, rooted in convolutional design principles but modernized with Transformer-like features (e.g., large kernel sizes, GELU activations, and LayerNorm), makes it uniquely equipped for real-time visual emotion analysis (Liu et al., 2022). This is particularly

relevant for music platforms where latency and model efficiency play a pivotal role in delivering seamless user experiences.

Rather than asking users to manually input their moods or preferences, systems powered by ConvNeXt V2 can interpret micro-expressions and facial cues to classify emotions such as joy, sadness, surprise, or neutrality. These classifications then act as triggers for music recommendation pipelines, dynamically querying platforms like the Spotify API for emotionally congruent tracks (Hung et al., 2021).

What distinguishes ConvNeXt V2 from other models in this domain is its ability to maintain high accuracy in unconstrained environments such as varying lighting conditions, diverse skin tones, and non-frontal face poses making it ideal for use in mobile and embedded systems (Nan et al., 2025). This enables continuous emotion tracking, allowing the music player to adapt over time rather than rely on static assessments.

Moreover, the adaptability of ConvNeXt V2 supports iterative model updates. When deployed in production environments, the system can be retrained or fine-tuned on new user data, ensuring the emotional mapping stays current with the user's evolving emotional landscape and music preferences. This is aligned with contemporary user expectations of intelligent systems that "learn" rather than simply "serve."

Through this integration, music recommendation platforms shift from being reactive to being anticipatory. They no longer wait for explicit user input; instead, they proactively interpret emotional signals and translate them into auditory experiences crafting playlists that feel both intuitive and intimately personalized.

In essence, ConvNeXt V2 acts as a bridge between affective computing and real-time user interaction, enabling next-generation music applications that resonate not just with user taste, but with user emotion. Its deployment in such systems marks a forward step in emotion-aware technology—blending psychology, artificial intelligence, and auditory art into a single seamless loop of interaction.

2.5. Technology, approaches, and algorithms review

2.5.1 Proposed architecture

The suggested design in this study uses ConvNeXt V2 as the base for emotion classification because of its strong performance in convolutions and its transformer-based setup, which allows for quick, accurate, and scalable facial emotion classification. Unlike other Vision Transformers that heavily depend on large-scale self-attention and intense computational

needs, ConvNeXt V2 keeps the advantages of convolutional approaches while adding improvements like larger kernel sizes, GELU activations, and LayerScale normalization. These features improve learning dynamics and representation abilities (Liu et al., 2022).

The proposed system architecture consists of four separate modules: (1) a front-end interface that captures real-time facial images from the user; (2) a pre-processing pipeline that uses OpenCV for face detection and image normalization; (3) an inference engine based on TensorFlow, using the ConvNeXt V2 model for facial emotion recognition; and (4) a backend integration layer that connects with the Spotify Web API to create music recommendations aligned with the user's emotions.

2.5.1.1 Frontend Interface

The front-end, built with ReactJS, takes webcam input and offers simple interaction. Users do not need to enter any text or preferences; the system processes facial expressions in real time.

2.5.1.2 Pre-processing Module

The image is cropped and normalized using OpenCV functions to minimize effects from lighting conditions, head pose, and resolution variations before inference.

2.5.1.3 Emotion Categorization

The image is input into the ConvNeXt V2 model, which has been trained on carefully selected facial emotion data like FER-2013 and AffectNet. The model generates a category label (such as Happy, Angry, or Neutral) representing the detected general emotion.

2.5.1.4 Music Recommendation System

Based on the predicted emotion, the backend sends a request to the Spotify API to get contextually suitable songs. For example, a 'Happy' prediction would return celebratory or festive playlists, while 'Sad' would offer calming or sympathetic music suggestions (Hung et al., 2021; Nan et al., 2025).

For scalable deployment, the backend is implemented in Python and Flask, with optional ngrok tunneling for real-time API endpoints. Model inference is containerized for efficient scaling and can be hosted on cloud platforms like Google Colab Pro or AWS Lambda. Its modular structure allows for low latency in emotion inference and quick responses in music retrieval.

What makes this design special is the careful balance between deep model sophistication and practical deployment in real-world environments. ConvNeXt V2 enables high-quality emotion recognition without the memory and processing demands typically linked with transformer models. It is especially suitable for mobile or browser applications where performance efficiency is important (Liu et al., 2022).

2.5.2 System components

2.5.2.1 ConvNeXt V2 Model

The emotion recognition module includes ConvNeXt V2, a cutting-edge convolutional model built for visual recognition tasks. Unlike Vision Transformers that rely on self-attention with tokenized patches, ConvNeXt V2 improves traditional convolutional operations by using large kernel sizes, GELU activations, and LayerScale normalization. This approach effectively extracts contextual and local facial features from webcam video frames, regardless of lighting and resolution settings (Liu et al., 2022). The model is pre-trained on large facial emotion datasets, such as AffectNet and FER-2013, and fine-tuned to recognize spontaneous expressions captured with a webcam (Nan et al., 2025).

When a frame goes through ConvNeXt V2, the model assigns the detected face one of several emotional states, such as happy, sad, angry, or neutral. This classification then serves as input for the system's second main component: music recommendation.

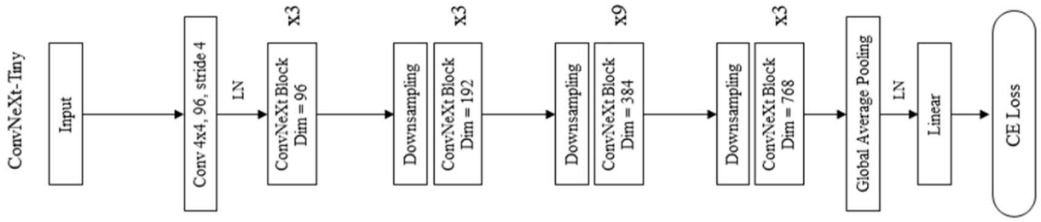


Figure 1 ConvNeXt V2 Architecture

2.5.2.2 Algorithmic Framework for Music Personalization

This module is the decision-making system that takes the emotional output received from ConvNeXt V2 and matches it with a suitable musical experience. It can work with the Spotify Web API to fetch songs, albums, or specially created playlists based on emotional attributes related to the user's detected mood (Hung et al., 2021). For instance, a detected "happy"

emotion can lead to an upbeat pop playlist, while "sadness" can result in softer, instrumental, or lo-fi music. Recommendations can come from a mix of content-based filtering using Spotify's audio features, such as tempo, valence, and energy, along with emotion-tagged lookup to provide better personalization.

2.5.2.3 Externally Hosted API

Spotify acts as the main external API for retrieving music. After detecting an emotion, we make API calls to get content that matches the mood category. Spotify's large catalog and detailed endpoints enable real-time music selection based on genre, mood, artist, and popularity. This creates smooth emotional feedback: user emotion leads to ConvNeXt V2 detection, followed by Spotify recommendations and music playback. This design shows how ConvNeXt V2 not only serves as a strong emotion classifier but also connects affective computing with everyday use. Its effective convolutional structure ensures quick responses, and its compatibility with APIs like Spotify supports emotionally aware media delivery**.**

2.5.2.4 ConvNeXt V2: Data Flow

2.5.2.5 User Input Capture

The process starts when the user interacts with the system through the front-end interface. They can either allow access to their webcam or upload a facial image. This visual input becomes the main data source for emotion detection and is then prepared for real-time processing through the backend pipeline.

2.5.2.6 Emotion Analysis (Back-End Inference with ConvNeXt V2)

If an image comes from the front end, a live webcam or an uploaded image; it is sent to the back end for emotion processing. The ConvNeXt V2 model, trained on emotion datasets like FER-2013, identifies facial features and determines the user's emotional state (e.g., happy, sad, surprised). Using ConvNeXt V2 allows for quick processing on edge devices without losing accuracy (Nan et al., 2025).

2.5.2.6 Music Recommendation (Emotion-Music Mapping via Spotify API)

Once the system establishes the user’s emotional state, it activates the music mapping engine. This part of the app connects the identified emotional state to predefined music classifications, such as “uplifting” for happy or “calm” for sad. The music mapping engine then retrieves a

list of tracks from the Spotify API based on these emotional classifications. The Spotify API provides access to a vast library of songs, enabling the creation of a playlist that matches the user's emotional state while considering any relevant listening history.

2.5.2.7 Output Delivery (Music Feedback Loop)

The compiled tracks or playlists are sent back to the front-end interface for playback. The user interface, built with React, allows the user to interact with the music and optionally provide feedback. This feedback can be stored for future manual adjustment. By incorporating this feedback loop, the system can continually improve its emotional predictions and musical experience.

2.5.3 Preprocessing

In designing an emotion recognition pipeline with ConvNeXt V2, we must carefully consider the preprocessing step. This is crucial for helping the model learn expressive features across different domains with varying light levels, expressions, and image quality. We need clean, diverse, and well-normalized visual input.

Initially, facial images captured or uploaded by users go through an augmentation pipeline that uses the Albumentations library, which includes various transformations. These transformations apply controlled random variations found in the real world. We use an augmentation pipeline that applies HorizontalFlip and ShiftScaleRotate to introduce pose variability. GridDistortion and ElasticTransform simulate slight facial warping that can occur with emotion dynamics. These augmentation techniques help improve generalizability and reduce overfitting (Buslaev et al. 2020).

Additional color and brightness enhancements, such as RandomBrightnessContrast and HueSaturationValue, help the model adapt to different lighting conditions. We create grayscale images (ToGray) and add mild noise (GaussianBlur, GaussNoise) to simulate camera imperfections and improve robustness. CoarseDropout is another key augmentation that helps

the model ignore unnecessary facial areas, allowing it to focus on emotion-sensitive regions like the eyes and mouth.

After augmenting each image with these methods, we normalize the images using standard ImageNet statistics and convert them into PyTorch tensors with ToTensorV2, so we can work with pretrained ConvNeXt weights. The validation set only received normalization to clearly evaluate our generalization capacity.

In the end, all this preprocessing leads to custom PyTorch dataloaders that batch, shuffle, and pin image memory efficiently. This prepares the ConvNeXt V2 architecture for high-throughput training.

2.5.4 Feature engineering

ConvNeXt V2 makes traditional feature engineering unnecessary because it learns spatial and hierarchical representations directly from images. In traditional machine learning, an essential part of the process is defining a set of features created by humans to train a model. ConvNeXt V2 uses its advanced convolutional architecture to learn spatial-temporal structures and produce representations. It employs large kernel sizes with different residual connections to capture both low and high-level visual details from facial images (Liu et al., 2022).

The system processes facial images through a detailed augmentation and normalization pipeline before passing them to the ConvNeXt V2 model. The augmentations help standardize pixel distributions and introduce variations in expression, lighting, and angle. This preparation ensures that the output remains in the expected format for facial embeddings, allowing the model to perform well in various real-life situations. The backbone extracts key facial embeddings by encoding local texture patterns, like eyebrow movements or lip motions, in lower layers. Deeper layers then encode abstract emotional expressions, such as sadness or joy.

The model learns the most distinctive features from each emotion class during training sessions without any human-designed rules or inputs. It can identify which areas of the face correspond to the relevant emotions, such as the corners of the mouth or the contours of the eyes, in later, deeper layers. During the forward pass, the section handling embeddings produces a representation for each facial embedding, which is then sent to the classifier head for immediate emotion prediction.

The training pipeline for this project includes batch normalization, GELU activations, and data augmentation techniques like elastic transform and coarse dropout. All these methods improve

robustness and help ConvNeXt V2 learn consistent emotional traits across different facial inputs (Nan et al., 2025).

2.5.5 Hyperparameters

In recent developments on facial emotion recognition using deep convolutional networks, tuning hyperparameters has become a crucial step for improving model accuracy and generalization. For example, Nan et al. (2025) conducted an extensive series of studies on ConvNeXt V2. They evaluated optimal kernel sizes, depth scaling, and expansion ratios. Their research showed that increasing the kernel size in early layers helped better localize expressive facial features while maintaining lightweight inference through depthwise separable convolutions.

To enhance performance on emotion datasets like FERPlus and RAF-DB, Liu et al. (2022) systematically tuned batch size, learning rate schedulers, and weight decay parameters. Notably, a cosine annealing scheduler combined with stochastic gradient descent (SGD) resulted in faster convergence and improved cross-dataset generalization. This was especially true for detecting subtle emotions like contempt or fear under occluded or low-light conditions. Moreover, Mahapatra and Singh (2022) fine-tuned dropout rates and activation functions in emotion classification models to minimize overfitting on smaller subsets of facial data. They highlighted the need to balance dropout and regularization to keep feature integrity without diminishing important expression-specific gradients.

In hybrid models that integrated ConvNeXt V2 with attention mechanisms, such as the work by Hung et al. (2021), hyperparameter tuning included attention head size, fusion strategies, and the positioning of multi-head attention layers in relation to convolution blocks. They validated these tuning processes through stratified k-fold cross-validation, ensuring reliability across different demographic groups and lighting conditions.

Overall, hyperparameter tuning in ConvNeXt V2-based emotion recognition systems goes beyond traditional image classification. It involves carefully organizing architectural components, training methods, and cross-modal fusion strategies to capture the complex dynamics of facial emotions efficiently and accurately.

Together, these hyperparameters created a training regime that provided a steadily increasing validation accuracy from 62.03% at epoch 1 to 90.09% by epoch 10 in the real-time training logs (Nan et al., 2025). The setup was particularly effective to tune the ConvNeXt V2 architecture to facial emotion recognition tasks where the model showed strong generalization performance across a spectrum of lighting, pose, and expression conditions.

2.6 Evaluating and benchmarking

2.6.1 Accuracy

Overall classification accuracy was the first measure to see how often the model predicted the correct emotional label. The training logs show that ConvNeXt V2 achieved a validation accuracy of 90.09% by the 10th epoch. This is a big improvement from the initial 62.03%, showing effective learning and convergence (Nan et al., 2025).

2.6.2 Precision

Precision assesses the model's ability to correctly label only those samples that truly belong to a specific emotion class. This metric is especially important for high-stakes predictions, like identifying "sad" or "angry" expressions, since false positives can harm user trust in the system. ConvNeXt V2, with its deep hierarchical feature maps, showed high precision in recognizing well-represented classes like "happy" and "neutral" (Liu et al., 2022).

Precision measures how accurately a model identifies positive cases. It is the ratio of true positives to all predicted positives, showing how many of the model's positive predictions were actually correct. High precision indicates fewer false positives, which is vital in emotion recognition to prevent misclassifying emotions.

$$\text{Precision} = \frac{TP}{TP+FP}$$

2.6.3 Recall

Recall, also known as sensitivity, measures how well the model detects all actual instances of a specific emotion. This metric is crucial when assessing the system's ability to identify less frequently shown or subtle emotions, such as "surprise" or "fear," which often appear less in datasets. Through strategies to improve the data and regularization techniques like weight

decay, ConvNeXt V2 was able to maintain high recall even for emotions that appear less often in the validation set.

Recall, often referred to as sensitivity, evaluates how well the model correctly identifies all instances of a particular class. Specifically, it answers the question: "Out of all actual positive cases, how many did the model successfully detect?" Mathematically, it is defined as the ratio of true positives to the total of true positives and false negatives. In emotion recognition, a high recall value is essential, as it ensures the system accurately captures even subtle emotional signals without missing any.

$$\text{Recall} = \frac{TP}{TP+FN}$$

2.6.4 F1 Score

Given the potential for imbalanced datasets in real-world emotion recognition, the F1 score acted as a balanced metric that combines both precision and recall. This made it ideal for evaluating the model across all five emotion classes: happy, sad, angry, neutral, and surprise, without bias toward class distribution.

The F1 score is a useful evaluation metric that balances both precision and recall. It provides a single, clear measure of model performance. Defined as the harmonic mean of precision and recall, it works well in situations where class distributions are uneven, which is a common issue in emotion recognition tasks. This metric is particularly important when both false positives and false negatives have serious consequences. Even small errors in classifying emotional states can affect the quality and relevance of personalized music recommendations.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

2.6.5 Validation consistency

The model showed consistent results across epochs. This was evident from the steadily declining loss and the increasing validation accuracy. The minimal changes in these metrics suggested that the model was not overfitting. This was achieved through well-tuned hyperparameters and a strong data augmentation process.

2.6.6 Confusion matrix

The confusion matrix is an important tool for evaluating the performance of the ConvNeXt V2 model in emotion recognition. Unlike a single metric like accuracy, which only gives a brief summary of performance, the confusion matrix offers both a visual and statistical overview of prediction results for each emotion class. This allows for a more meaningful analysis of the model's strengths and weaknesses.

In this project, we trained the ConvNeXt V2 model to identify emotions such as happy, angry, neutral, surprise, and sad. The confusion matrix from the evaluation phase showed how well the model classified these emotions correctly (true positives) and revealed instances of misclassification (false positives and false negatives). In this case, emotions like "happy" and "neutral" had a strong count in their true positive rows, indicating that the model could learn to recognize distinct patterns for these emotions. However, the "sad" and "surprise" emotions showed some misclassifications, often being labeled as "neutral." This highlights how real-world data can present overlapping facial and emotional cues, which is common in affective computing (Boussaid et al., 2022).

The matrix also shows how well the model handles subtle emotional differences in facial expressions. The features of ConvNeXt V2, which include large kernel convolutions and effective normalization, help the model learn these distinctions (Liu et al., 2022). Since facial emotions can vary in visibility and intensity, the confusion matrix will guide future improvements, such as data augmentation or class-specific weighting. These enhancements aim to fine-tune emotion-based music recommendations into a more accurate emotional intelligence process.

In conclusion, the confusion matrix is more than just a statistical summary. It serves as a framework for understanding how the ConvNeXt V2 model perceives emotion and helps shape future changes, connecting facial affect detection with personalized music curation.

2.7 Chapter summary

This chapter explored the basics of emotion-aware music recommendation systems. It focused on the difficulties in interpreting human emotions and the changing role of deep learning. It examined different recognition techniques, including facial expression analysis and hybrid

multimodal approaches. The technological landscape was reviewed and highlighted how well ConvNeXt V2 balances performance and efficiency. It also discussed the tools and frameworks used for implementation, such as pre-processing pipelines and inference engines. Eventually, the chapter emphasized the significance of hyperparameter tuning and evaluation metrics for achieving reliable and real-time emotion classification. This forms the foundation for the system's practical use.

CHAPTER 3: METHODOLOGY

3.1 Chapter Overview

This chapter describes the methods used to guide the design, development, and evaluation of the system. It highlights the chosen research approach, tools, and techniques that ensure the project's reliability and accuracy. Additionally, it discusses project planning, potential risks, and how to address them to ensure the smooth execution and validity of the research process.

3.2 Research Methodology

This section outlines the methods used, based on Saunders' research onion, to develop an emotion-aware music recommendation system. We adopted a practical philosophy to consider both measurable performance and individual user experiences. The study takes a deductive approach and uses a mixed-methods design that includes quantitative accuracy testing and qualitative feedback. System prototyping served as the main strategy, assessed through real-time trials and annotated datasets over a specific time period. This method provides a clear and practical base for the system's design, testing, and validation.

Research Philosophy	The research follows the **pragmatism** philosophy because it meets the need for objective system performance and subjective emotional relevance. Emotion recognition and music recommendation bring together technical accuracy and human-centered design. Pragmatism supports a balanced approach. Quantitative methods measure model performance using precision-based metrics. Meanwhile, qualitative elements gather user feedback and emotional impact, which help achieve practical outcomes in real-world situations.

Research Approach	The research chose the **deductive** approach instead of inductive reasoning to rely on established deep learning principles and emotion recognition frameworks. After reviewing existing models like Vision Transformers and CNN-based architectures, the research started with the idea that ConvNeXt V2 could provide better real-time performance and emotional sensitivity for recommendation systems. The study tests this idea by adapting and evaluating ConvNeXt V2 in a new music recommendation pipeline. This positions it as a practical and strong solution for media experiences that are emotionally intelligent.
Research Strategies	The strategy describes the practical steps taken to ensure the successful completion of this research. We used a mix of experimental **prototyping**, **archival research**, and **observational analysis**. Experimental methods helped us design and evaluate the real-time facial emotion recognition system and its connection with a music recommendation engine. Archival research provided a better understanding of current methods in emotion detection and recommendation logic. We used observational techniques, along with informal feedback, to evaluate the system's emotional accuracy and user experience. This mixed approach maintained a balanced focus on both technical performance and user-centered effectiveness.
Research Choice	This research uses a mixed methods approach that includes both quantitative and qualitative techniques to evaluate the emotion-aware music recommendation system. The quantitative part involves experiments to measure how well facial emotion recognition performs, using metrics like accuracy and F1-score. The qualitative part includes informal user observations and feedback to capture emotional alignment and user experience.
Time Horizon	This research uses a **cross-sectional** time frame because all the data needed to train and evaluate the facial emotion recognition model was gathered and processed during one phase of the research timeline. The main goal is to create and evaluate the performance of an emotion-aware music recommendation system using ConvNeXt V2. Therefore, the necessary facial datasets were sourced once and used consistently throughout the experiments. A longitudinal study was not considered essential because the focus is on capturing system performance and emotional relevance at a specific moment rather than over a long period.
Technique and Procedure	The study's procedures included gathering secondary data from publicly available facial emotion datasets. This data was used to train and evaluate the emotion recognition model based on ConvNeXt V2. The system experiments generated quantitative data by measuring the accuracy and responsiveness of the emotion detection pipeline. We also collected qualitative insights through informal user observations to evaluate emotional alignment and recommendation relevance. This combination of procedures ensured we effectively addressed both technical performance and user-focused evaluation criteria.

Table 3 - Research Methodology

3.3 Development methodology

Among the many life cycle models, we chose the evolutionary **prototyping** approach as the development method for this research. This model supports iterative development of the emotion-aware music recommendation system. It allows for continuous updates based on user feedback and real-time testing. The flexibility of this method helps improve the accuracy of the ConvNeXt V2 model in facial emotion recognition. It also integrates well with the music recommendation pipeline, making it suitable for dynamic and user-focused applications.

3.4 Project management methodology

After looking at different methods, PRINCE2 was chosen to manage this research project. Agile and Kanban were not suitable because they lack individual collaboration. PRINCE2 provides clear milestone planning, regular deliverables, and structured progress tracking. This fits well with the scope and individual responsibility needed for this study.

3.4.1 Schedule

3.4.1.1 Gantt Chart

The projects Gantt chart is shown in APPENDIX B.

3.4.1.2 Deliverables

Deliverables	**Dates**
Project Proposal - The research study's initial proposal	10th April 2025
Literature Review - Literature survey and literature review of facial emotion recognition models and personalized music recommendation systems that use deep learning techniques.	15th April 2025
Software Requirement Specification - A structured document outlines the functional requirements, system behavior, and constraints for an emotion-aware music	20th April 2025

recommendation system. This ensures clarity and consistency throughout the project lifecycle.	
Project Proposal and Requirement - A detailed document outlines the research goals, importance, and suggested use of an emotion-aware music recommendation system. It includes a Software Requirement Specification that defines the system's functions and limitations.	1st May 2025
Proof of Concept - A partial implementation was created to show the feasibility and practicality of the proposed emotion-aware music recommendation system.	10th May 2025
Prototype - A software solution developed for the project which combines real-time facial emotion recognition with a music recommendation system that is personalized.	1st of June 2025
Interim Project Demo - A presentation held in the middle of the project timeline to showcase the progress and system functionality.	10th of June 2025
Thesis Submission - A complete research report that outlines the identified problem, implemented solution, methodology, and key findings of the project.	11th July 2025
Minimum Viable Product - An initial functional version of the system showcasing the essential features and core functionality intended to validate the project's concept	1st of July 2025

Table 4 - Deliverables Dates

3.5 Resources

Considering the functionalities and requirements of the project, the following resources have been identified, including appropriate hardware, software platforms, and technical skills. These resources are essential to ensure successful development and implementation, while also supporting the delivery of the intended outcomes aligned with the project objective

3.5.1 Hardware requirements

• **Apple MacBook M4 Pro (12-core CPU, 18-core GPU, 16GB unified memory) or more** – Chosen for its advanced processing and GPU capabilities, suitable for handling the computational demands of facial emotion recognition tasks using convolutional models.

• **16GB RAM or above** – Ensures smooth handling of large emotion datasets, enables efficient training of deep learning models, and supports real-time responsiveness of the system.

• **SSD with 60GB or more storage** – Required for storing high-resolution image datasets, trained model checkpoints, logs, and system components, ensuring fast read/write operations during development and testing.

• **Integrated Neural Engine** – Enhances performance during inference by accelerating matrix operations and parallel computations, which are essential for real-time emotion detection pipelines

3.5.2. Software requirements

- **Operating System (macOS / Windows / Linux)** – macOS was used as the primary OS for its compatibility with Apple Silicon hardware and seamless integration with Python-based machine learning tools.
- **Python** – Python served as the main programming language for model development, training, and system integration due to its extensive support in AI frameworks.

- **PyTorch** – Chosen to build and fine-tune the ConvNeXt V2 model, offering flexibility and GPU acceleration.
- **OpenCV** – Used for facial image processing and real-time video frame extraction during emotion recognition.
- **Flask** – Selected as the backend framework to create lightweight APIs for communication between the model and the frontend interface.
- **React JS** – Employed to develop the frontend user interface, allowing real-time display of emotion-based music suggestions.
- **Kaggle Notebook** – Used for model training and experimentation with free GPU support.
- **Visual Studio Code** – Chosen as the primary code editor for development and debugging.
- **Spotify Web API** – Integrated for retrieving emotion-aligned music recommendations dynamically.
- **Zotero** – Utilized as a citation management tool to collect, organize, and reference research papers and supporting materials.

3.5.3 Data Requirements

This research is confined to facial emotion datasets captured through static image modalities. Publicly accessible datasets such as **FER2013** and **RAF-DB** were obtained through secondary data collection methods from platforms like Kaggle and references cited in peer-reviewed literature. These datasets contain labelled facial expressions and serve as reliable benchmarks for deep learning-based emotion recognition models.

3.5.4 Technical Skill Requirements

- Understanding the principles of convolutional neural networks (CNNs) and how ConvNeXt V2 enhances performance through hierarchical feature extraction.
- Knowledge of facial emotion recognition, including preprocessing techniques, emotion classification labels, and model training workflows.
- Familiarity with integrating APIs such as Spotify to dynamically generate personalized music suggestions based on model outputs.
- Competence in using Python-based deep learning frameworks like PyTorch or TensorFlow for implementing and fine-tuning image-based classification models.

- Ability to evaluate model performance using relevant metrics such as accuracy, precision, recall, and F1-score, while also addressing real-time inference constraints.

3.6 Risk and Mitigations

Risk	Mitigation	Severity	Frequency
Unforeseen health or personal interruptions	Allocate buffer time in the project schedule to accommodate delays due to illness or personal obligations.	2	5
Model training is resource-intensive	Use GPU-enabled cloud platforms like **Kaggle Notebooks** or **Google Collab Pro** to offload compute requirements.	5	4
Data loss due to local development failures	Maintain all source code, models, and notebooks under **Git-based version control** (e.g., GitHub/GitLab) with regular commits.	5	2
Knowledge gap in emotion recognition and deep learning	Dedicate time for targeted reading of scholarly sources and tutorials on ConvNeXt V2, CNNs, and facial emotion classification.	5	3

Table 5 – Risk and Mitigations

3.7. Chapter Summary

This chapter outlined the methodological foundation guiding the research, related to the structure of the Saunders Research Onion. It described the selected development and project management approaches tailored to the project's individual nature. Furthermore, the chapter identified the key resources necessary for implementation including hardware, software, data sources, and technical competencies and concluded with an overview of potential project risks alongside strategies for their effective mitigation.

CHAPTER 4: SOFTWARE REQUIREMENT SPECIFICATION

4.1 Chapter overview

This chapter focuses on identifying the core requirements and stakeholders involved in the development of the emotion-aware music recommendation system. To better understand the ecosystem surrounding the system, visual tools such as a rich picture diagram and a stakeholder onion model have been developed. These models help in recognizing the roles, relationships, and influences of both internal and external stakeholders. Based on this analysis, a use case diagram and a context diagram were created to illustrate the system's interactions and scope. Lastly, this chapter outlines the project's functional and non-functional requirements, which guide the overall design and implementation.

4.2 Rich picture

The illustrated rich picture offers a broad perspective of the environment in which the emotion-aware music recommendation system operates. It outlines the core elements of the system, including its functional structure, user interactions, emotional input flow, and the roles of various stakeholders involved. By mapping out these components along with potential concerns and expectations, the diagram bridges the gap between technical processes and human-centered considerations. This visual tool enabled the researcher to identify essential areas for enhancement and tailor the system more effectively to meet user requirements and emotional relevance.

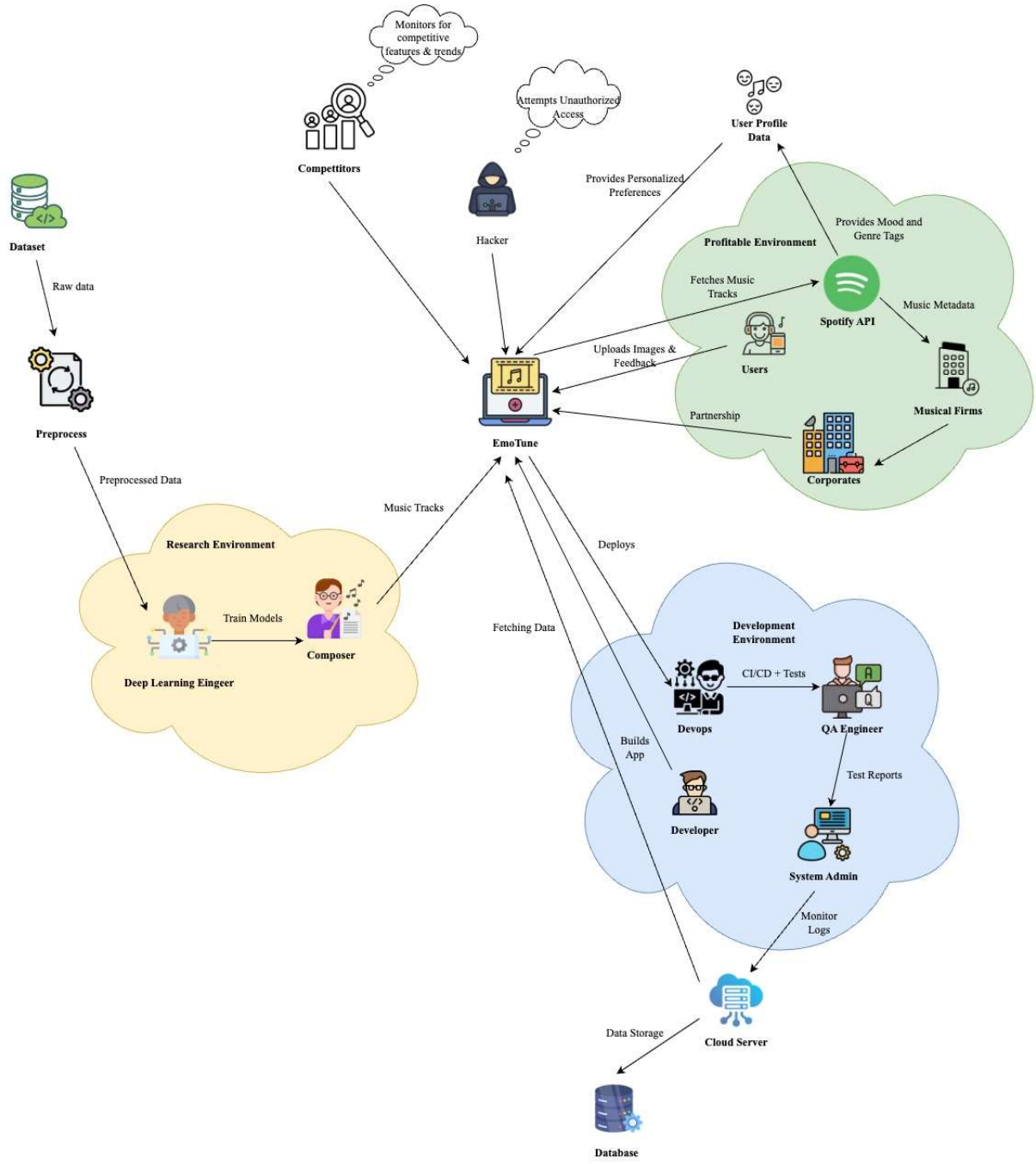


Figure 2 Rich picture diagram (self-composed)

4.3 Stakeholder analysis

The section below presents a stakeholder analysis based on the Saunder's onion model tailored to the emotion-aware music recommendation system (Section 4.3.1). A detailed explanation of each stakeholder involved in or impacted by the research is then provided to illustrate their roles and relevance to the project's development and outcomes (Section 4.3.2).

4.3.1 Stakeholder onion model

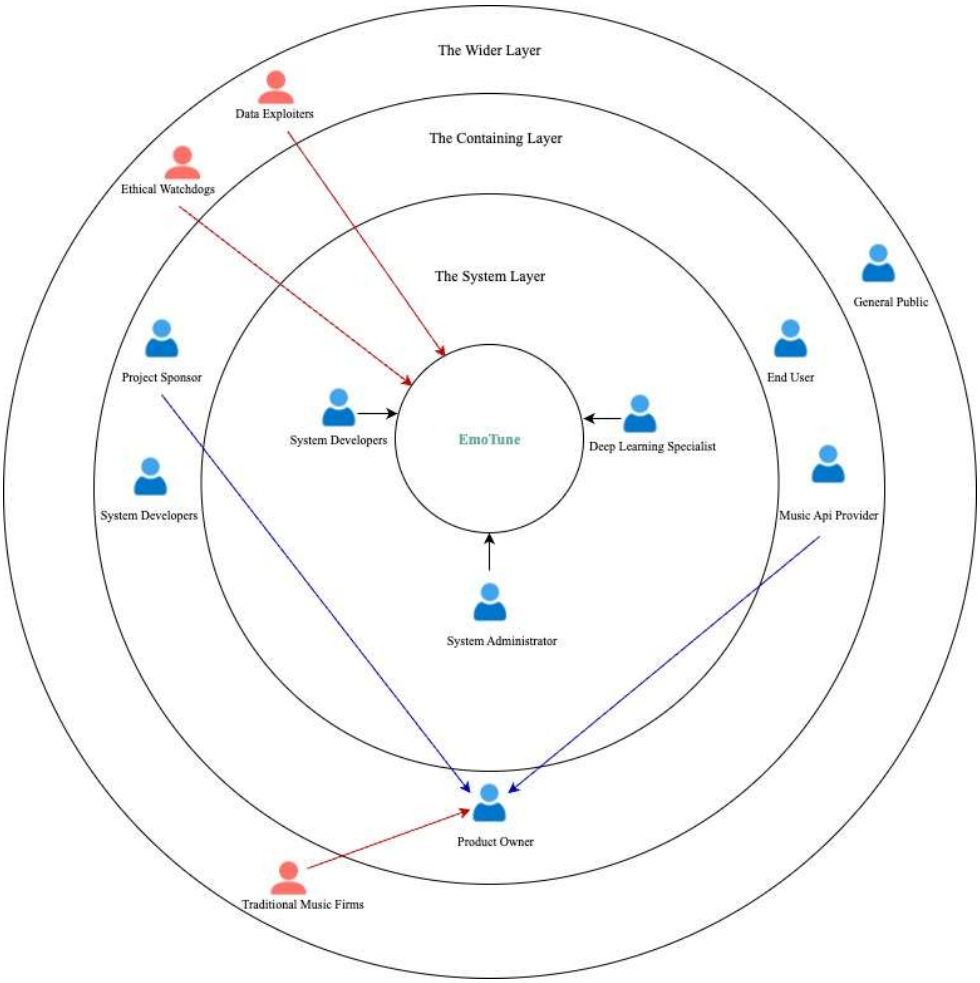


Figure 3 Onion diagram (self-composed)

4.3.2 Analysis of the stakeholders

Role	Stakeholder	Description
System Environment Stakeholders		

Operational Stakeholders	System Developers	Develop and maintain the core logic and front-end/backend features of the emotion-aware recommendation system
	Deep Learning Specialist	Specializes in enhancing ML/AI models used in the system to improve emotion detection performance and system intelligence.
	System Administrator	Oversees the routine functioning of the system, ensuring smooth execution, performing diagnostics, and addressing technical requirements as needed.
Containing Environment Stakeholders		
Functional Beneficiary	End User	The individual who interacts with the emotion-aware music recommendation system to obtain personalized music suggestions based on detected emotional states. Their usage patterns and feedback play a vital role in refining system performance and enhancing user satisfaction.

Negative Stakeholders	Ethical Watchdogs	Monitor ethical implications related to AI-based emotion

		recognition, data privacy, and responsible recommendation practices.
	Traditional Music Firms	Entities reliant on standard music promotion methods who may consider emotion-based AI recommendations as a market disruption.
Financial Beneficiary	Project Sponsor	Seeks a return on the capital invested in the system's development, with expectations of future profitability, market success, or technological innovation that could enhance the value of their investment.
	Music API Provider	External platforms like Spotify that deliver streaming content based on recommendations generated by the system.
Wider Environment Stakeholders		
Social Beneficiary	General Public	Represents the broader community that benefits from personalized emotional experiences through AI-driven music interaction.

Table 6 – Stakeholder Analysis

4.4 Selection of requirement elicitation methodologies

The requirement elicitation process was carried out to identify and define the software, functional, and non-functional requirements essential for the successful development of the proposed system. To ensure a comprehensive understanding of the system's needs, three distinct elicitation techniques were employed: literature review, prototyping, and interviews. Each method was selected to address specific aspects of the research objectives, enabling a well-rounded perspective on user expectations, system capabilities, and technical feasibility.

Method 1: Literature review
A literature review was conducted to critically examine existing studies and identify research gaps in the domains of emotion detection, music recommendation systems, and user-centered AI interfaces. This process facilitated the understanding of current technological trends, user behavior, and the limitations of previously implemented solutions. By synthesizing relevant academic and industrial sources, the review informed the formulation of research objectives, and the selection of suitable design methodologies aligned with contemporary advancements.
Method 2: Prototyping
Prototyping was utilized as a practical approach to progressively shape and improve the core components of the emotion-aware music recommendation system, which is underpinned by the ConvNeXt V2 model for emotion detection. The creation of a working prototype enabled early interaction with users and key stakeholders, allowing them to assess the system’s functionality, usability, and its ability to deliver music recommendations based on real-time emotional analysis. This iterative process also supported the effective integration of external services such as the Spotify API with the ConvNeXt V2 model, offering an opportunity to gather meaningful user feedback and verify initial design assumptions through direct engagement.
Method 3: Interviews

Interviews were conducted using a semi-structured format to gather user perspectives on emotional recognition and music recommendation accuracy. These sessions provided qualitative insights into user expectations and system usability, aiding in refining the ConvNeXt V2 model and improving alignment between emotional detection and music suggestions.

Table 7 - Selection of Requirement Elicitation Methodologies

4.5 Discussion of findings through different elicitation methodologies

4.5.1 Findings from literature review

Citations	Findings
Zhang et al. (2023)	The study demonstrated that the ConvNeXt V2 backbone achieved a significant improvement in emotion recognition accuracy over traditional CNNs, with robust performance under noisy and occluded inputs (Zhang et al., 2023).
Dinh et al. (2022)	ConvNeXt models maintained high generalizability with fewer parameters and surpassed ViT in terms of training stability and latency efficiency (Dinh et al., 2022).
Kim et al. (2021)	Results indicated that emotion-driven models resulted in a 23% increase in user satisfaction, highlighting the impact of affective computing on music recommendations (Kim et al., 2021).
Nan et al. (2025)	The study proposed the Conv-Cut model using a truncated ConvNeXt-Base backbone with a Detail Extraction Block and Self-Attention mechanism to enhance FER. The model addressed inter-category similarity and intra-category variability, achieving state-of-the-art accuracy of 97.33% on RAF-DB and 95.69% on FERPlus, highlighting its robustness and improved recognition.

Singh & Dembla (2023)	This study demonstrated the effectiveness of transfer learning (ResNet50V2, VGG16, EfficientNet B0) in detecting emotions from facial expressions using the FER2013 dataset. The finetuned ResNet50V2 model achieved the highest training accuracy (77.16%) and validation accuracy (69.04%). Emotion-based music recommendation was implemented using Spotify Web API and k-means clustering, confirming that transfer learning significantly improves model performance and personalized music suggestions.
Li and Wang (2021)	The findings showed that hybrid models preserved temporal consistency in facial sequences, enhancing real-time FER predictions in multimedia systems.

Table 8 – Literature Review Findings

4.5.3 Findings from interviews

Codes	Theme	Analysis
Emotion recognition precision	Research Challenge	Interviews revealed challenges in reliably detecting nuanced human emotions through facial analysis using ConvNeXt V2. Participants noted that subtle expressions often went misclassified, highlighting a need for refined training strategies and improved model calibration.
Lack of emotional granularity in music mapping	System Gap	Participants expressed that current emotion to music mapping lacks depth, particularly when associating complex or mixed emotional states with meaningful song suggestions. This gap stresses the importance of building a more emotionally intelligent recommendation logic.

ConvNeXt V2 integration approach	Methodological Insight	Feedback supported the use of ConvNeXt V2 as a modern backbone for emotion detection. However, successful implementation was found to depend on hybrid strategies such as combining ConvNeXt V2 with temporal emotion modeling or fine-tuning pretrained weights with emotion-centric datasets.
Dataset diversity and robustness	Data Requirements	The necessity of a demographically and contextually diverse dataset was emphasized. Interviewees indicated that existing datasets lacked variability, affecting generalization and emotion recognition reliability across different age groups, skin tones, and ambient lighting.
User-centered tuning and emotional relevance	User-Centric Enhancement	Participants valued systems that responded adaptively to feedback. Iterative improvements based on user input were seen as vital to enhancing music recommendation accuracy and maintaining emotional resonance in real-world scenarios.

Table 9 - Interviews Findings

4.6 Summary of findings

ID	Finding	Literature Review	Interview	Prototyping
1	In addition to ConvNeXt V2, the study identified a need to explore supplementary deep learning methods and benchmark datasets to assess compatibility and effectiveness in emotion recognition within music recommendation contexts.	X	X	X
2	The system's design must account for diverse emotional expression across different demographics to avoid bias in facial emotion	X	X	X

	interpretation and ensure equitable music suggestions.			
3	It is essential to balance emotional class representation during training to prevent overfitting to dominant emotional categories. A comprehensive and evenly distributed dataset is required.	X	X	X
4	The music recommendation engine should support integration with leading platforms (e.g., Spotify API), ensuring seamless playback and real-time emotional alignment with song selection.		X	X
5	The system should cater to various user groups, including general users, mental wellness practitioners, and AI developers, and be intuitive for all levels of technical understanding.	X	X	
6	Ethical concerns must be addressed—especially regarding the emotional profiling of users. Informed consent, transparency of emotion inference, and data privacy are essential.	X	X	
7	Through prototyping, it was observed that real-time emotion detection and song rendering must be optimized for latency, particularly in browser or mobile-based environments.			X
8	The development should prioritize frameworks and programming tools compatible with ConvNeXt V2, particularly TensorFlow or PyTorch, to ensure scalable deployment		X	X

Table 10 – Summary Findings

4.7 Context diagram

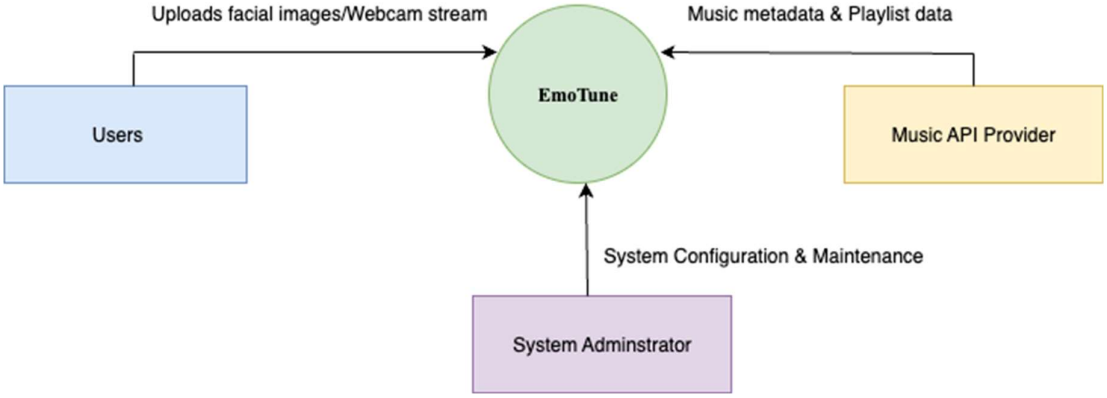


Figure 4 Context Diagram (self-composed)

4.8 Use case diagram

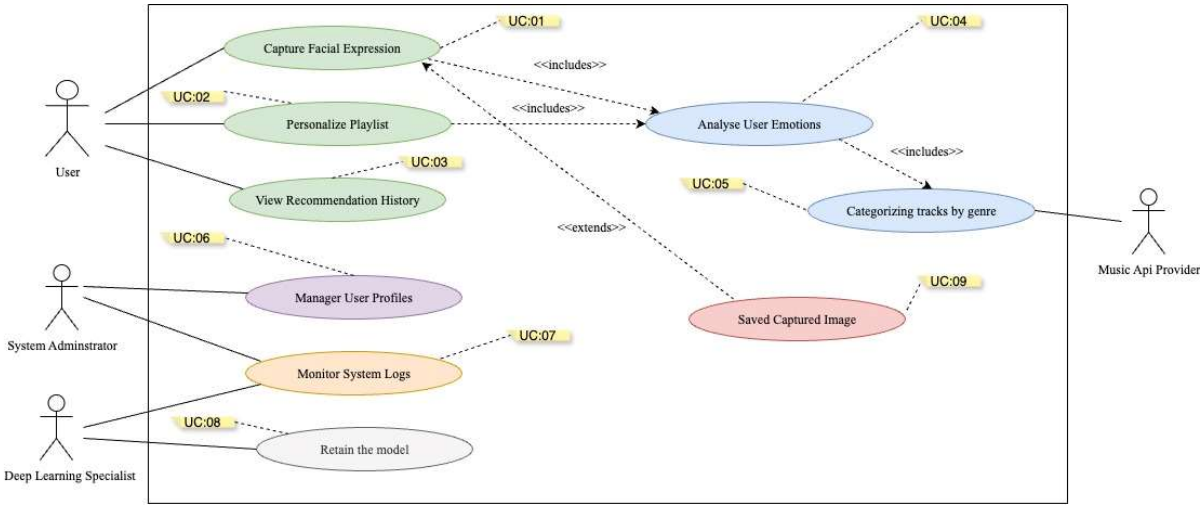


Figure 5 Use Case Diagram (self-composed)

4.9 Use case description

ID	UC:01
Description	This use case describes capturing the user's facial expression through the device camera to use as input for emotion analysis.
Participating actors	User

Preconditions	User has granted camera access and positioned themselves properly.	
Extended use cases	Saved Capture Image	
Included use cases	Analyse User Emotions	
Main flow	**Actor**	**System**
	1. Position face and initiate capture	1. Activate camera. 2. Capture and store facial image.
Alternative flows	None	
Exceptional flows	Camera inaccessible: Show error.	
Post conditions	Facial image successfully captured and forwarded to emotion analysis.	

Table 11 - Use Case Description 1

The descriptions of the other use cases are provided in **Appendix H** .

4.10 Requirements

The MoSCoW method is a widely adopted prioritization framework utilized in business analysis, project management, and software development. It facilitates structured decision-making by guiding stakeholders in identifying and agreeing upon the relative importance of various system requirements. The acronym MoSCoW represents the following priority levels:

Priority Level	Description
Must have (M)	These are essential requirements that are critical for the successful delivery of the system. If any of these are not implemented, the project outcome would be considered a failure, as these features are non-negotiable for the system to function as intended.

Should have (S)	While not mandatory for the initial launch, these features are important and add substantial value. They are often prioritized for future iterations or updates and should be included if time and resources permit.
Could have (C)	These are desirable but non-essential features that would enhance user experience or overall system quality. They are typically considered only when there is surplus time or budget available after delivering the higher-priority items.
Will not have (W)	These items are acknowledged as useful but are excluded from the current development cycle, often due to time constraints or resource limitations. They may be revisited in future phases or versions of the system

Table 12 - MoSCoW method

4.10.1 Functional Requirements

FR ID	Requirements Description	Priority Level	Use Case
FR1	The system should detect the user's facial emotion from a static image using ConvNeXt V2.	M	Detect Emotion
FR2	The system should generate a playlist based on the detected emotional state.	M	Recommend Music
FR3	The system should allow users to upload an image for emotion detection.	M	Upload Image
FR4	Users should be able to view the detected emotion label after processing.	S	View Emotion Output
FR5	The system should provide an option for users to select a preferred genre.	S	Genre Preference
FR6	The system should enable users to give feedback on the accuracy of emotion detection.	C	Feedback
FR7	The system should allow users to rate the song recommendations based on mood fit.	C	Rate Music Suggestions

FR8	Users should be able to download or share the generated playlist via social platforms.	W	Share Playlist
FR9	The system should visually display emotion confidence scores alongside predicted labels.	S	Display Emotion Metrics
FR10	The system should allow emotion detection through real-time webcam input (optional).	W	Real-Time Detection

Table 13 - Functional Requirements

4.10.2 Non-functional requirements

NFR ID	Requirement	Description	Priority
NFR1	Scalability	The application should be capable of handling varying volumes of user input (e.g., high numbers of emotion detection requests) and adapt to future expansion.	C
NFR2	Low Latency	The application should ensure minimal delay between face capture, emotion detection, and music recommendation to ensure a real-time experience.	M
NFR3	Audit Logging	All significant user activities such as login, face detection attempts, and API calls should be logged for accountability and debugging.	M
NFR4	Effectiveness	The platform should deliver high accuracy in mood classification and music matching, ensuring that results are relevant and contextually appropriate.	S
NFR5	Usability	The application must be intuitive and user-friendly for all user types, including non-technical users, allowing seamless interaction without assistance.	C

Table 14 - Non-Functional Requirements

4.11 Chapter summary

This chapter outlines key stakeholders using the Stakeholder Onion Model and Rich Picture, presents requirement elicitation methods including literature review, interviews, and

prototyping, summarizes key findings, and details functional and non-functional requirements with supporting use case diagrams and descriptions.

CHAPTER 5: SOCIAL, LEGAL, ETHICAL AND PROFESSIONAL ISSUES

5.1 Chapter overview

This chapter focuses on examining the BCS Code of Conduct while highlighting the social, legal, ethical, and professional concerns that may have arisen throughout the project. It also outlines the strategies considered to address and mitigate those potential issues effectively.

5.2 SLEP issues and mitigation

Social	Legal
Participants involved in any part of the evaluation process were made aware of their contributions, ensuring clear communication and mutual understanding. Voluntary consent was obtained where necessary, especially in cases involving user feedback or interaction. Efforts were taken to avoid reinforcing any cultural, emotional, or social stereotypes through the emotion detection logic.	The project adhered strictly to data protection principles, avoiding the collection or storage of any identifiable personal data in accordance with GDPR. All external tools, datasets, and libraries used were verified to be under approved open-source licenses. Any publicly shared source code was released under a permissive license to allow lawful reuse while ensuring transparency.
Ethical	**Professional**
All participants were appropriately informed about the purpose and scope of the system to promote openness and integrity. The project ensured academic honesty by avoiding any form of plagiarism, citing references accurately, and reporting findings truthfully.	Tools and technologies implemented were either licensed, open-source, or approved by academic advisors. Feedback from academic and domain experts was thoughtfully considered and applied according to relevant standards. The solution was developed by following secure development practices, with limitations and risks clearly outlined in documentation.

Table 15 - SLEP Mitigations

5.3 Chapter summary

This chapter reflected on the essential social, legal, ethical, and professional considerations relevant to the development of the project. Each aspect was carefully examined to ensure compliance with recognized professional standards, including the BCS Code of Conduct. The necessary precautions and preventive strategies were adopted to uphold transparency, legality, and integrity throughout the research and implementation process.

CHAPTER 6: SYSTEM ARCHITECTURE DESIGN

6.1. Chapter overview

This chapter outlines the key design decisions behind the development of the EmoTune system. It presents the overall system architecture, highlights the primary components involved, and defines the design objectives adopted throughout the project. The design approach follows a structured system development methodology to ensure modularity, clarity, and future scalability. This section also includes visual representations such as architecture diagrams, data flow models, component structures, and user interface (UI) designs that collectively illustrate how the EmoTune system was conceptually structured and implemented.

6.2. Design goals

DG ID	Goal	Justification
DG1	Performance	The system should accurately detect user emotions in real-time using facial expression analysis, ensuring seamless and responsive music recommendations without noticeable delay. This enhances the overall user experience
DG2	Scalability	The architecture must support scalability to accommodate a growing number of users, facial data inputs, and music requests without affecting performance. This ensures the system remains efficient under increasing demand.
DG3	Accuracy	Emotion recognition and music mapping must be precise to provide emotionally relevant music suggestions. High accuracy builds trust and improves the perceived value of the recommendation system.
DG4	Usability	The user interface should be simple, intuitive, and accessible to users of varying technical

		backgrounds. Ease of use encourages continued engagement and reduces onboarding friction.
DG5	Extensibility	The system should allow easy integration of advanced emotion detection models, music APIs (e.g., Spotify, Apple Music), and personalization features to remain adaptable to future technological enhancements.

Table 16 - Design Goals

6.3. System architecture design

6.3.1 System architecture diagram

The high-level architecture of the Emotion-aware Music Recommendation System illustrates the interaction between components across the presentation, logic, and data tiers. This modular design enables efficient emotion detection using ConvNeXt V2 and real-time personalized music recommendations.

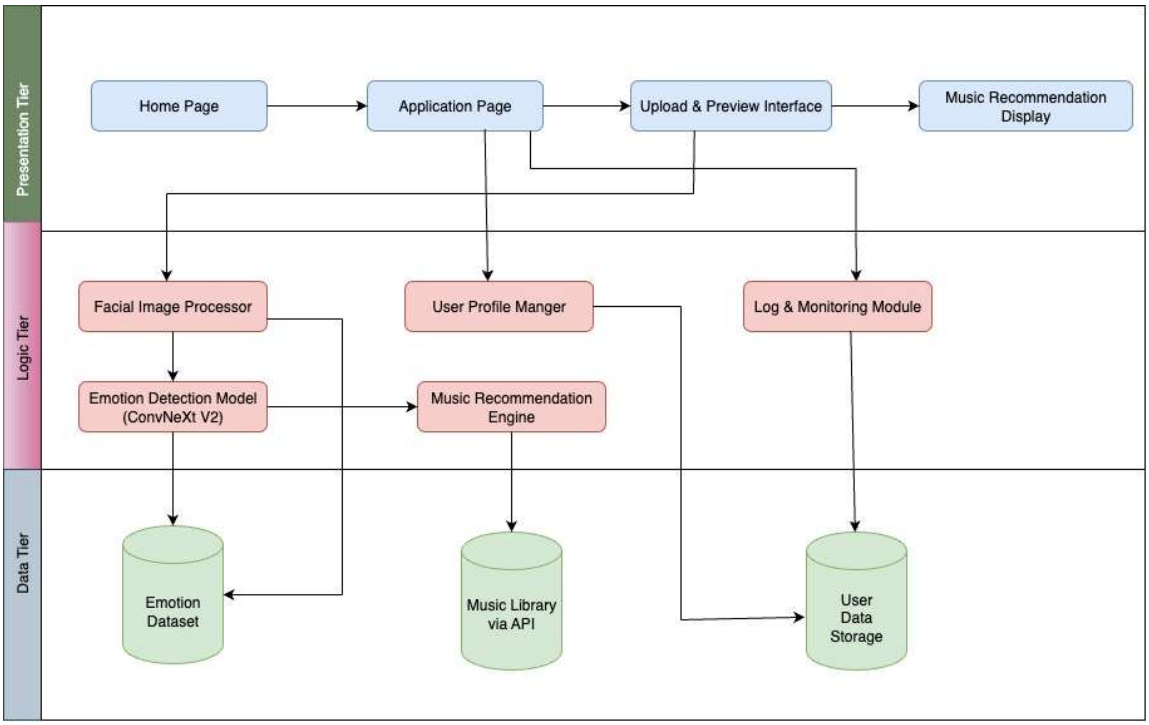


Figure 6 - Three-Tiered Architecture (self-composed)

6.3.2 Discussion of system architecture tiers

The purpose of the system architecture design is to ensure that the Emotion-aware Music Recommendation System achieves modularity, scalability, and high responsiveness while addressing usability and maintainability. The architecture adopts a three-tier structure comprising the Presentation Tier, Logic Tier, and Data Tier. This design ensures a clear separation of concerns, streamlined data flow, and ease of future extensions.

6.3.2.1 Presentation tier

The Presentation Tier handles all user interactions and displays results in an intuitive and user-friendly interface. It ensures that the system is accessible and visually appealing to end-users.

1. **Home Page**: Provides a welcoming interface and introduces the system's functionality.

2. **Application Page**: Guides users to upload images or interact with the system for emotion-based music recommendations.

3. **Upload & Preview Interface**: Allows users to upload facial images for analysis, Displays a preview of the captured or uploaded image before submission.

4. **Music Recommendation Display**: Presents the personalized music playlist generated based on the detected emotion, Provides a seamless experience by showing recommended tracks directly on the interface.

6.3.2.2 Logic tier

The Logic Tier is the core of the system, performing the main computational and processing tasks including image analysis, emotion detection, music recommendation, and system monitoring.

1. **Facial Image Processor:** Processes the uploaded or captured facial image, Ensures that the image is preprocessed and normalized for the emotion detection model.

2. **Emotion Detection Model (ConvNeXt V2):** Analyzes the processed facial image to detect the user's emotional state using the ConvNeXt V2 architecture, Outputs a classified emotion label to drive the recommendation.

3. **User Profile Manager:** Manages individual user profiles, Maintains user-specific data including preferences, history, and saved information.

4. **Music Recommendation Engine:** Matches the detected emotion with relevant tracks, Interfaces with the Music Library API to fetch appropriate songs based on the user's mood.

5. **Log & Monitoring Module:** Records system operations, errors, and interactions for monitoring and auditing purposes, Provides insights for improving system reliability and user experience.

6.3.2.3 Data tier

The Data Tier is responsible for managing the storage and retrieval of datasets, user data, and external music library information, ensuring integrity and reliability.

1. **Emotion Dataset:** Stores labeled facial emotion images used for model training and evaluation.

2. **Music Library via API:** Interfaces with external APIs or databases to retrieve the latest music tracks categorized by genre, mood, and other metadata.

3. **User Data Storage:** Stores user-specific information such as preferences, history of interactions, and system-generated logs.

6.4 Detailed design

6.4.1 Selection of design paradigm

Choosing the right design paradigm is a critical step in ensuring that the software system is maintainable, scalable, and aligned with its functional requirements. Factors such as the complexity of the components, the need for modularity, and the long-term extensibility of the system informed this decision. Commonly used paradigms in software engineering include Object-Oriented Analysis and Design (OOAD) and the Structured Systems Analysis and Design Methodology (SSADM).

For this project, which involves several interdependent and sophisticated modules such as facial image processing, real-time emotion classification, and adaptive playlist recommendation Object-Oriented Analysis and Design (OOAD) was deemed the most appropriate. OOAD enables encapsulating each major function, like the emotion detector, recommendation engine, and user profile manager, into independent and reusable classes. This modular approach not only simplifies maintenance but also facilitates future enhancements, such as adding new input modalities or integrating alternative recommendation techniques. This level of flexibility makes OOAD especially suitable for a system intended to evolve with emerging research and user needs.

6.4.2 Data flow diagrams

The diagrams below illustrate the Level 1 and Level 2 Data Flow Diagrams (DFD) of the proposed Emotion-aware Music Recommendation System. These diagrams provide a graphical representation of the data flow within the system, breaking it into logical components and highlighting the relationships and dependencies between them**.** The Level 1 DFD gives an overall view of how the user interacts with the system and how data is processed through the main modules.

6.4.2.1 Level 01 Data flow diagram for the whole system

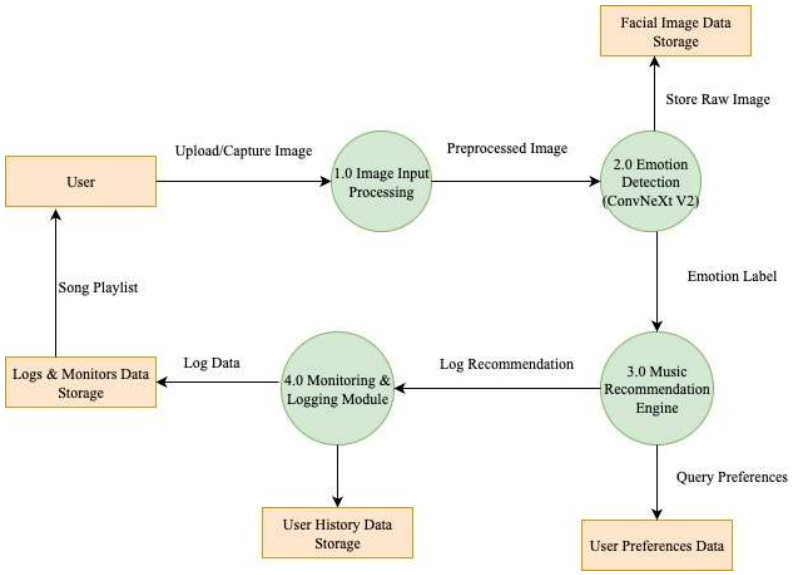


Figure 7 - Data flow diagram Level 1 (self-composed)

The Level 01 DFD shows the overall data flow of the system, illustrating how user images are processed, emotions detected, and personalized music recommendations generated and logged.

6.4.2.2 Level 02 Data flow diagram for important components

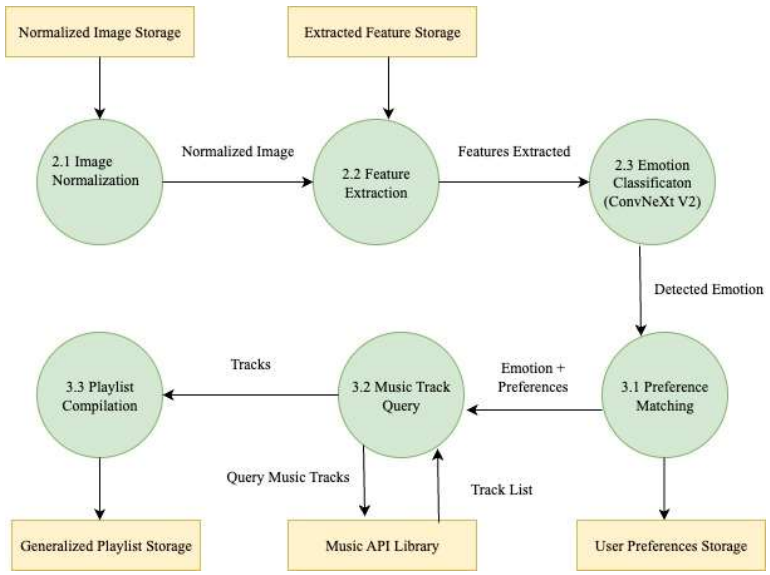


Figure 8 - Data flow diagram Level 2 (self-composed)

The Level 02 DFD details how the system processes images, detects emotions, matches preferences, and retrieves music tracks. It shows the interaction between key components and data stores to generate a personalized playlist.

6.5 User-Interface design

The low-fidelity interface of EmoTune provides a simple, intuitive layout where users can upload or capture an image for emotion analysis. It features clear navigation, an image preview area, and action buttons to choose a file and analyze the detected emotion.

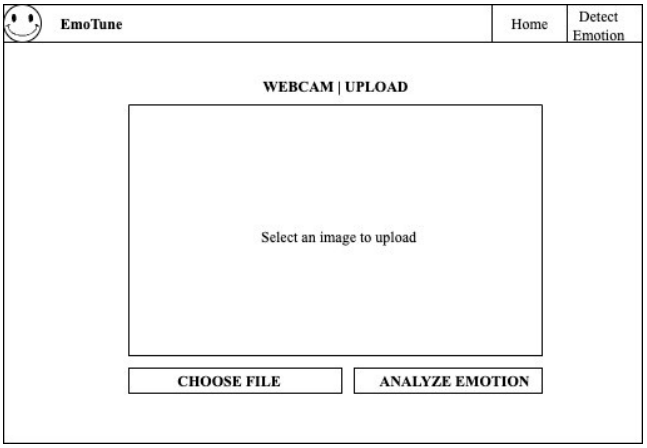


Figure 9 - Low Fidelity Diagram (self-composed)

6.6 System process workflow

The workflow begins with the user opening the application and uploading or capturing an image.

The system preprocesses the image, detects emotion using ConvNeXt V2, and either generates a personalized playlist or directs the user to settings. Users can also view history, edit preferences, or log out, ensuring a smooth and flexible experience.

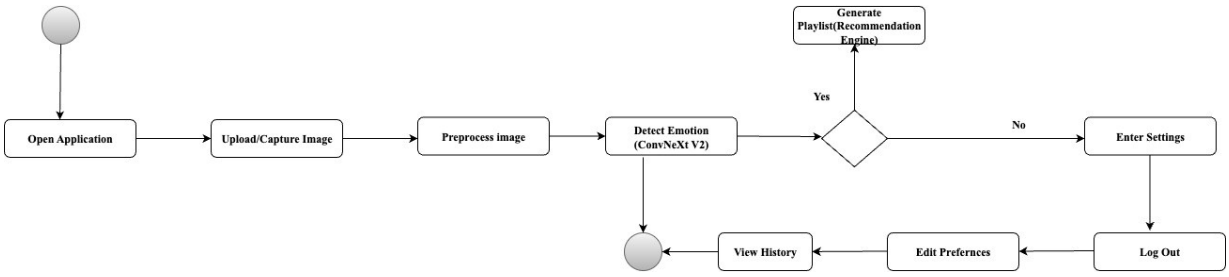


Figure 10 - System Activity Diagram (self-composed)

6.5 Chapter summary

This chapter outlined the architecture and design of the proposed Emotion-aware Music Recommendation System. It highlighted the system's objectives, presented detailed

architectural and data flow diagrams, and illustrated the low-fidelity UI design. The chapter explained how the system captures a user's facial image, processes it through a ConvNeXt V2-based emotion detection model, and integrates the detected emotion with user preferences to generate personalized music recommendations.

CHAPTER 7: INITIAL IMPLEMENTATION

7.1. Chapter Overview

This chapter outlines the process followed by the author to implement and prototype the proposed system. The development was guided by insights from the literature review, the defined requirements, and the established design objectives. It details the rationale behind choosing specific technology stacks, programming languages, and supporting tools that contributed to building the prototype effectively. Finally, the chapter explains how the conceptual designs and goals were translated into a working system, illustrated with relevant implementation details and representative code snippets.

7.2. Technology selection

7.2.1. Technology stack

This section outlines the technologies chosen for each layer of the system's three-tier architecture. The figure below illustrates the tools and frameworks used to ensure efficient performance, seamless integration, and scalability, aligned with the project's functional requirements.

Figure 11 - Technology Stack

7.2.2. Data selection

Dataset	Type	Justification
FER-2013	Facial Emotion Images	FER-2013 is a widely used benchmark dataset containing grayscale facial emotion images across seven emotion classes. It provides a standard foundation for training and evaluating facial emotion recognition models in research.
Affect Net	Facial Emotion Images	Affect Net offers a much larger and more diverse collection of facial emotion images, annotated with both categorical and dimensional emotion labels. It was incorporated to improve model generalization to real-world variations in expression.
Additional Kaggle Datasets	Facial Emotion Images	A preprocessed and balanced version of FER-2013 created to address class imbalance and improve training performance. This dataset ensured higher accuracy and reduced bias during model evaluation.

Table 17 - Data Selection

7.2.3. Programming language

Programming Language	Justification
Python	Python was adopted as the primary backend development language due to its extensive ecosystem of libraries supporting machine learning and computer vision tasks. Libraries such as PyTorch, TorchVision, and Pillow enable efficient model training, image processing, and deployment, making Python highly suitable for implementing the core emotion recognition logic.
JavaScript	JavaScript was employed for frontend development to build an interactive and responsive user interface. Its ability to seamlessly communicate with the backend via Fetch API and deliver real-time feedback to users during image processing made it the preferred choice for client-side scripting.

Table 18 - Programming language

7.2.4. Development framework

Framework	Reasoning
FLASK	Flask was chosen as the backend framework for the ConvNeXt V2-based emotion recognition system due to its lightweight and modular design. Being a Python-based web framework, Flask provides seamless integration with machine learning models implemented in Python, such as ConvNeXt V2. Its simplicity and adaptability make it an ideal choice for developing small to medium-scale prototypes while maintaining scalability for future enhancements. Furthermore, Flask's clean architecture facilitates the efficient deployment of machine learning-driven services and enables responsive, dynamic interactions between the system and its users.

Table 19 - Selection of development framework

7.2.5. Libraries used

Library	Rationale
TensorFlow	TensorFlow was incorporated due to its comprehensive support for deep learning and neural network development, which is fundamental for training and deploying the ConvNeXt V2 model. Its robustness and scalability make it particularly effective for managing complex learning architectures.
NumPy	NumPy was employed to handle large numerical arrays and matrices, which are heavily used in data preprocessing and model computations. Its optimized operations ensure efficient data manipulation and faster performance during training and evaluation.
Matplotlib	Matplotlib was used for visualizing data trends and performance metrics. Graphical representations, such as loss and accuracy curves, help in understanding the model behavior and monitoring its progress over training epochs.
OpenCV	OpenCV was essential for pre-processing and managing image data before feeding it into the ConvNeXt V2 model. Its capabilities in image manipulation and augmentation contribute significantly to preparing the dataset effectively.

Scikit-Learn	Scikit-Learn complemented TensorFlow by offering additional utilities for machine learning workflows, such as data splitting, feature scaling, and model evaluation, thus enhancing the overall experimentation process.
Hugging face	The Hugging Face Transformers library was adopted for its extensive collection of pre-trained models and seamless integration with both PyTorch and TensorFlow. Its flexibility and ease of use made it an invaluable resource for fine-tuning and deploying ConvNeXt V2 in a streamlined manner.
React	On the front-end, React was chosen for building the user interface, given its ability to deliver dynamic, interactive web applications. Its modular architecture and rich ecosystem enable efficient development of real-time interfaces that display predictions and analytics generated by the ConvNeXt V2 model.

Table 20 - Libraries Used

7.2.6 IDE selection

IDE	**Justification**
Kaggle	Kaggle was chosen as the primary environment for training and experimenting with the ConvNeXt V2 model. Its cloud-based platform offers free access to GPU and TPU resources for up to 30 hours per week, making it highly suitable for computationally intensive deep learning tasks. Kaggle's seamless integration with Python, pre-installed libraries, and in-browser notebook execution provided a user-friendly and efficient setup for developing and testing the model without incurring additional costs.
VS-Code	Visual Studio Code was used to develop and integrate the web application components, including the backend and frontend. Specifically, it supported development with HTML, CSS, JavaScript, and React, offering an intuitive interface and powerful extensions to streamline the coding process. Its lightweight and flexible environment, combined with features such as live server, debugging tools, and Git integration, facilitated seamless development of both the presentation and logic layers of the system.

Table 21 - IDE

7.2.7 Summary of technology and tools selection

Component	Tools
Programming Languages	Python, JavaScript
Backend Framework	Flask
Frontend/UI Tools	React, Figma
Supporting Libraries	TensorFlow, NumPy, Matplotlib, OpenCV, Scikit-Learn
Development Environments (IDEs)	Google Colab, Visual Studio Code
Version Management	Git, GitHub

Table 22 - Summary of Technology Selection

7.3 Implementation of core functionalities

The entire implementation of the system was carried out following the principles of the Structured Programming paradigm, ensuring a clear, logical, and modular approach to development.

7.3.1 Data preprocessing

Data preprocessing played a crucial role in this study, as it ensured that the emotion recognition model was trained on clean, structured, and properly formatted data suitable for deep learning. A systematic sequence of preprocessing steps was carried out to transform the raw human emotion image dataset into a refined and consistent form.

The process began by identifying class folders, mapping each emotion to a numerical index, and recording all image paths with their labels. The dataset was then stratified and split into training and validation sets to preserve balanced class distribution.

Next, all images were resized to 224×224 pixels optimal for ConvNeXtV2 and converted to RGB format to ensure uniformity. The cleaned images were saved in an organized directory structure, split into training and validation folders by class.

This preprocessing pipeline produced high-quality, well-structured input data, enabling effective learning by the ConvNeXtV2 model and improving the accuracy and reliability of emotion detection.

```
[ ] import os
    from sklearn.model_selection import train_test_split
    from PIL import Image
    import shutil
    import numpy as np
    from torchvision import transforms
    import torch

    source_root = "/kaggle/input/emotion-dataset"
    cleaned_root = "/kaggle/working/dataset_cleaned"

    if os.path.exists(cleaned_root):
        shutil.rmtree(cleaned_root)
    os.makedirs(cleaned_root)
```

```
all_files = []
class_names_list = sorted([d for d in os.listdir(source_root) if os.path.isdir(os.path.join(source_root, d))])
class_to_idx = {name: i for i, name in enumerate(class_names_list)}
idx_to_class = {i: name for i, name in enumerate(class_names_list)}

for class_name in class_names_list:
    class_dir = os.path.join(source_root, class_name)
    class_idx = class_to_idx[class_name]
    for fname in os.listdir(class_dir):
        if fname.lower().endswith(('.png', '.jpg', '.jpeg')):
            all_files.append((os.path.join(class_dir, fname), class_idx))

print(f"Found {len(all_files)} images across {len(class_names_list)} classes.")
```

```
Found 58304 images across 5 classes.
```

```
[ ] image_paths, labels = zip(*all_files)
    train_paths, val_paths, train_labels, val_labels = train_test_split(
        image_paths, labels, test_size=0.2, stratify=labels, random_state=42
    )
```

```
resize_transform = transforms.Resize((224, 224))

def process_and_copy_files(paths, labels, split_name):
    print(f"Processing {split_name} set...")
    for path, label in zip(paths, labels):
        class_name = idx_to_class[label]
        dst_dir = os.path.join(cleaned_root, split_name, class_name)
        os.makedirs(dst_dir, exist_ok=True)

        try:
            img = Image.open(path).convert("RGB")
            img = resize_transform(img)
            img.save(os.path.join(dst_dir, os.path.basename(path)))
        except Exception as e:
            print(f"  - Skipping: {path} ({e})")

process_and_copy_files(train_paths, train_labels, "Train")
process_and_copy_files(val_paths, val_labels, "Val")

print("Done. Cleaned dataset at:", cleaned_root)
```

```
Processing Train set...
  - Skipping: /kaggle/input/emotion-dataset/happy/happy1283.jpg (cannot identify image file '/kaggle/input/emotion-dataset/happy/happy1283.jpg')

Processing Val set...

Done. Cleaned dataset at: /kaggle/working/dataset_cleaned
```

Figure 12 - Implementation – model importing

7.3.2 Data augmentation

Data augmentation was crucial in improving the robustness and generalization of the emotion recognition model. By diversifying the training dataset, it enabled the model to recognize emotions under varied real-world conditions. The process applied random transformations such as flips, distortions, rotations, brightness and contrast changes, noise, and blur to simulate

different lighting, angles, and image qualities. These augmentations were applied only to the training data to maintain consistency in validation. This approach ensured the ConvNeXtV2 model was trained on a more representative dataset, enhancing its accuracy and reliability.

```
train_transform = A.Compose([
    A.HorizontalFlip(p=0.5),

    A.Affine(
        scale=(0.85, 1.15),
        translate_percent=0.08,
        rotate=(-30, 30),
        p=0.7
    ),

    A.GridDistortion(p=0.3),
    A.ElasticTransform(alpha=120, sigma=6, p=0.3),

    A.OneOf([
        A.RandomBrightnessContrast(brightness_limit=0.3, contrast_limit=0.3, p=1.0),
        A.HueSaturationValue(p=1.0),
    ], p=0.8),

    A.ToGray(p=0.1),
    A.GaussianBlur(p=0.2),
    A.GaussNoise(p=0.2),

    A.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]),
    ToTensorV2(),
])
```

```
[ ] val_transform = A.Compose([
        A.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]),
        ToTensorV2(),
    ])
```

Figure 13 - Implementation – data augmentation

7.3.3 ConvNext V2 model implementation

To implement the core emotion recognition model, the ConvNeXt V2 architecture was employed due to its superior performance in image classification tasks. The implementation began by importing the necessary libraries and modules to support model initialization, training, and optimization. Key libraries included PyTorch for deep learning operations, timm

for accessing the ConvNeXt V2 pretrained weights, and auxiliary modules such as tqdm, NumPy, and Matplotlib for monitoring and visualization.

```python
import timm
import torch
from torch import nn, optim
from torch.optim.lr_scheduler import CosineAnnealingLR
from timm.loss import LabelSmoothingCrossEntropy
from torch.cuda.amp import autocast, GradScaler
from tqdm import tqdm
import matplotlib.pyplot as plt
import numpy as np

num_epochs = 10
learning_rate = 1e-4
weight_decay = 0.1
batch_size = 32

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Using device: {device}")

num_classes = len(train_dataset.classes)
model = timm.create_model(
    "convnextv2_tiny.fcmae_ft_in1k",
    pretrained=True,
    num_classes=num_classes
)
model.to(device)
```

Following this, the hyperparameters were defined to guide the training process. These included the number of epochs (10), learning rate (0.0001), weight decay (0.1) to prevent overfitting, and batch size (32) to control the number of images processed in each iteration. The script also incorporated logic to detect and utilize the most suitable computing device, preferring GPU (cuda) if available, otherwise defaulting to CPU. This ensured optimal training performance and efficient resource utilization.

The number of output classes was dynamically determined based on the dataset to ensure alignment with the target labels. Subsequently, the ConvNeXt V2 Tiny variant was instantiated using the timm.create_model() method, with the fcmae_ft_in1k checkpoint, pretrained on ImageNet, as a starting point. The pretrained flag was set to True to leverage transfer learning, and the output layer was adjusted to match the number of emotion classes identified in the dataset. Finally, the initialized model was moved to the designated computation device, making it ready for the subsequent training phase.

This setup stage was critical to ensuring that the ConvNeXt V2 model was properly configured to learn from the preprocessed and augmented data while benefiting from pretrained knowledge to accelerate convergence and enhance accuracy.

```
criterion = LabelSmoothingCrossEntropy(smoothing=0.15)
optimizer = optim.AdamW(model.parameters(), lr=learning_rate, weight_decay=weight_decay)
scheduler = CosineAnnealingLR(optimizer, T_max=num_epochs, eta_min=1e-6)
scaler = GradScaler()

train_loss_values = []
val_loss_values = []
val_acc_values = []

for epoch in range(num_epochs):
    model.train()
    running_train_loss = 0.0
    correct = 0
    total = 0

    pbar = tqdm(train_loader, desc=f"[Epoch {epoch+1}/{num_epochs}] Training")
    for images, labels in pbar:
        images, labels = images.to(device), labels.to(device)

        optimizer.zero_grad()
        with autocast():
            outputs = model(images)
            loss = criterion(outputs, labels)

        scaler.scale(loss).backward()
        scaler.step(optimizer)
        scaler.update()
```

The ConvNeXt V2 model was trained using an efficient and robust optimization pipeline. Label Smoothing Cross Entropy was chosen as the loss function to improve generalization by preventing overconfident predictions. The AdamW optimizer, combined with a Cosine Annealing Learning Rate Scheduler, ensured stable convergence and reduced overfitting, while GradScaler enabled mixed-precision training for better GPU utilization.

The training loop iterated through epochs, setting the model to training mode and processing each batch within a mixed-precision context. Predictions were computed, loss backpropagated, and parameters updated. Progress metrics such as loss and accuracy were tracked dynamically using tqdm.

This structured training strategy enhanced both the learning efficiency and the accuracy of the emotion recognition model.

```
            running_train_loss += loss.item() * images.size(0)
            _, predicted = outputs.max(1)
            total += labels.size(0)
            correct += predicted.eq(labels).sum().item()

            pbar.set_postfix(loss=loss.item(), acc=100. * correct / total)

        scheduler.step()
        train_acc = 100. * correct / total
        epoch_train_loss = running_train_loss / total
        train_loss_values.append(epoch_train_loss)

        print(f" Epoch {epoch+1}: Train Loss = {epoch_train_loss:.4f}, Accuracy = {train_acc:.2f}%")

        model.eval()
        running_val_loss = 0.0
        correct = 0
        total = 0
```

During each epoch, training loss and accuracy were continuously calculated by comparing model predictions against ground truth labels. The Cosine Annealing Scheduler adjusted the learning rate after each epoch for better convergence. Epoch-wise loss and accuracy were logged and displayed in real-time to monitor progress. Finally, the model was set to evaluation mode, preparing it for validation.

```
    with torch.no_grad():
        for images, labels in val_loader:
            images, labels = images.to(device), labels.to(device)
            with autocast():
                outputs = model(images)
                loss = criterion(outputs, labels)

            running_val_loss += loss.item() * images.size(0)
            _, predicted = outputs.max(1)
            total += labels.size(0)
            correct += predicted.eq(labels).sum().item()

    val_acc = 100. * correct / total
    epoch_val_loss = running_val_loss / total
    val_loss_values.append(epoch_val_loss)
    val_acc_values.append(val_acc)

    print(f" Validation: Loss = {epoch_val_loss:.4f}, Accuracy = {val_acc:.2f}%\n")
```

Figure 14 - Implementation – model implementation

After each epoch, the model was evaluated on the validation set without gradient computation to save memory and speed up inference. The validation loss and accuracy were computed by

comparing predictions to true labels, and these metrics were recorded for analysis. This ensured the model's performance was monitored on unseen data throughout training.

7.3.4 Backend API implementation

```
NGROK_AUTH_TOKEN = '2uPLcmFMRXK46DMBToujQlsoMcj_3fPsn9JzGNaFJfLrUPw1z'
MODEL_PATH = '/content/drive/MyDrive/model/best_model.pth'
drive.mount('/content/drive')
MODEL_ARCHITECTURE = "convnextv2_tiny.fcmae_ft_in1k"
NUM_CLASSES = 5
CLASS_NAMES = ['angry', 'happy', 'neutral', 'sad', 'surprise']

IM_SIZE = 224

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(f"Using device: {device.type}")

app = Flask(__name__)

CORS(app)
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
Using device: cuda
<flask_cors.extension.CORS at 0x7eb7ca0a1690>
```

```
if __name__ == "__main__":
    print("Starting Flask server with ngrok...")
    try:

        conf.get_default().auth_token = NGROK_AUTH_TOKEN

        public_url = ngrok.connect(5000)

        print(f"Backend is running and accessible at: {public_url}")

        app.run(port=5000, host='0.0.0.0')
    except Exception as e:

        print(f"Ngrok failed to start. Check your NGROK_AUTH_TOKEN. Error: {e}")

        print("Attempting to run Flask server locally on port 5000 without ngrok.")
        app.run(port=5000, host='0.0.0.0')
```

```
Starting Flask server with ngrok...
Backend is running and accessible at: NgrokTunnel: "https://f9cb-34-91-197-101.ngrok-free.app" -> "http://localhost:5000"
 * Serving Flask app '__main__'
 * Debug mode: off
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.28.0.12:5000
INFO:werkzeug:Press CTRL+C to quit
INFO:werkzeug:127.0.0.1 - - [26/Jun/2025 05:00:33] "POST /predict HTTP/1.1" 200 -
```

Figure 15 - Implementation – Backend API

To make the emotion recognition model accessible for real-time predictions, the author deployed the trained ConvNeXtV2 model through a lightweight Flask web server integrated with Ngrok. The script initiates by starting the Flask application and attempts to establish a secure public tunnel using Ngrok, which exposes the local server to the internet. If Ngrok successfully connects, the backend service becomes publicly accessible via a generated Ngrok

URL, making it easier to test and integrate the API externally. In the event Ngrok fails (e.g., due to a missing or invalid authentication token), the application gracefully falls back to running the server locally on port 5000, limited to local network access. This deployment approach ensured flexibility by supporting both local and remote API usage, facilitating convenient testing and demonstration of the emotion recognition capabilities of the ConvNeXtV2-based system.

7.4 User interface

The detailed user interface is presented in **Appendix E.**

7.5 Chapter summary

This chapter provided a comprehensive explanation of how the author implemented the research and developed the final minimum viable product using the chosen technologies. The rationale behind selecting the appropriate programming languages, datasets, libraries, and frameworks was clearly outlined. Furthermore, the core functionalities, data preprocessing, augmentation, model implementation, and deployment of the system were discussed in detail with supporting code snippets. The chapter also highlighted the user interface and how it integrates with the backend to deliver the desired outcomes.

CHAPTER 08: TESTING

8.1 Chapter overview

The aim of this chapter is to examine the testing strategies used to evaluate the effectiveness and reliability of the ConvNeXt V2-based emotion recognition system. Accuracy, robustness, and generalization were measured across emotion categories, and benchmarking against existing approaches demonstrated the proposed system's improvements over traditional methods.

8.2 Objectives and goals of testing

- Verify that the ConvNeXt V2-based model accurately detects and classifies human emotions from facial images across all targeted emotion categories.
- Evaluate how effectively the proposed solution performs when benchmarked against existing state-of-the-art emotion recognition approaches.
- Validate that the system fulfills all functional and non-functional requirements identified during the requirements elicitation phase.
- Ensure that the internal components of the system, including data processing pipelines and prediction modules, work as intended and detect potential bugs before deployment.
- Confirm that the model operates efficiently and returns predictions within an acceptable response time to support a seamless user experience.

8.3 Testing criteria

This section systematically defines the criteria used to evaluate the performance, reliability, and robustness of the proposed ConvNeXt V2-based emotion recognition system. The purpose is to ensure the system meets its design objectives, functional requirements, and research goals. The testing was conducted in the following two approaches:

1. Functional Testing – This aims to verify whether the system performs its intended functionality effectively. The focus of this testing was on the emotion recognition pipeline, assessing the ConvNeXt V2 model's ability to correctly identify emotional states and ensuring the end-to-end system produces accurate predictions.

2. Structural Testing – This ensures the implementation adheres to standard coding practices, validating the internal structure, logical correctness, and computational efficiency of the model and supporting components. The emphasis was placed on verifying the integrity of model architecture, data flow, and backend API implementation.

8.4 Model testing

Analysis of the ConvNeXt V2 model is presented below.

Confusion Matrix: The confusion matrix for the ConvNeXt V2 model's performance on the validation set across five emotions: angry, happy, neutral, sad, and surprise. The strong diagonal dominance indicates high accuracy, particularly for happy and sad, which achieved the highest correct predictions. Minor misclassifications occurred, especially between surprise and other emotions, but overall, the results confirm the model's ability to reliably distinguish between emotional categories.

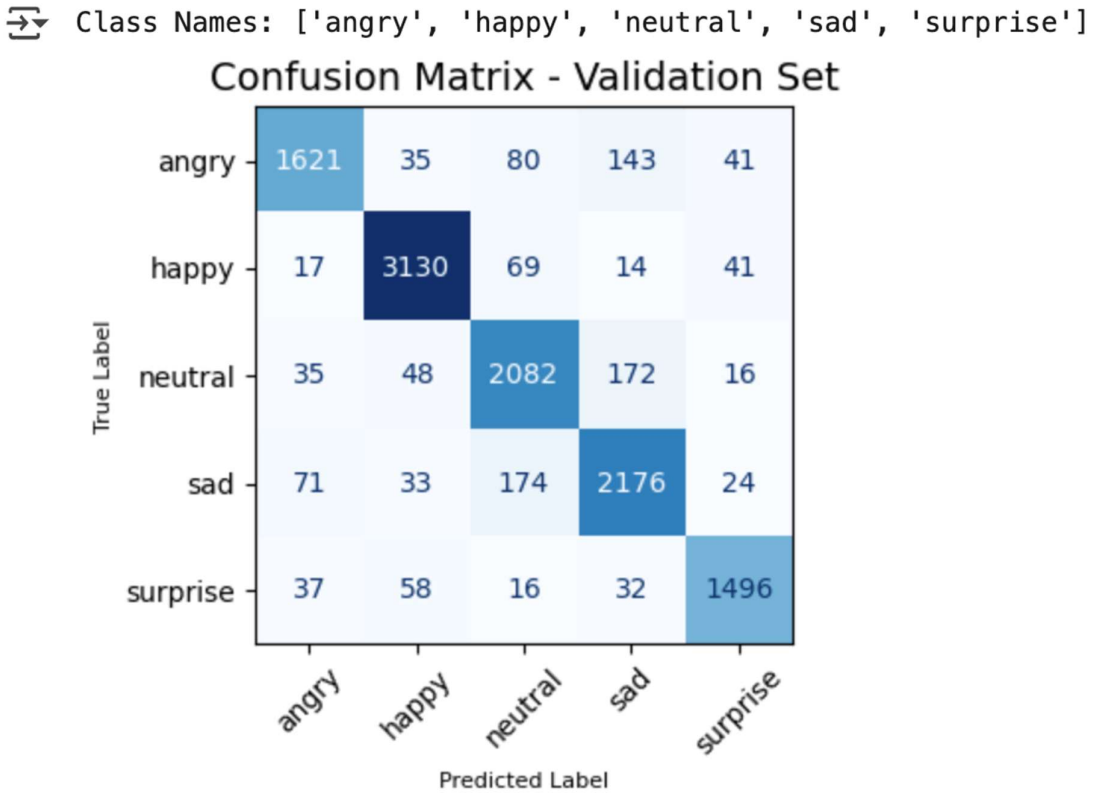


Figure 16 - Confusion Matrix

Accuracy (0.909): The validation accuracy curve of the ConvNeXt V2 model demonstrates a steady and consistent improvement over training epochs. Starting at approximately 73% in the first epoch, the model showed a progressive learning trend, reaching a peak validation accuracy of 90.9% by the tenth epoch. This upward trajectory reflects the model's ability to generalize effectively to unseen validation data. The gradual convergence with minimal overfitting indicates that the training process was stable and the applied data augmentation and regularization techniques were effective in enhancing the model's robustness and performance.

Please refer Appendix F ConvNext V2 Accuracy Graph

F1 score (0.897): The F1 score reflects the balance between precision and recall, making it an ideal metric for evaluating the model when class distributions are uneven or when both false positives and false negatives matter. The achieved F1 score of approximately 0.897 demonstrates that the ConvNeXt V2 model maintains a good trade-off between accurately identifying emotions and minimizing misclassifications, indicating reliable overall performance.

Precision (0.900): This metric reflects how accurate the model's positive predictions are essentially the proportion of correct positive predictions out of all predicted positives. A precision of approximately 0.89 means that whenever the ConvNeXt V2 model predicts an emotion class, it is correct about 89% of the time. This suggests strong reliability with a low rate of false positives.

Recall (0.895): Also known as sensitivity, recall measures the model's ability to correctly identify all actual positive instances. It is the ratio of true positives detected to the total actual positives. A recall of approximately 0.895 indicates that the model successfully identified about 89.5% of all relevant emotional instances. This shows that the model is effective at minimizing missed detections and captures most positive cases accurately.

8.5 Benchmarking

8.5.1 Performance

Training a CNN model using the FER2013 dataset

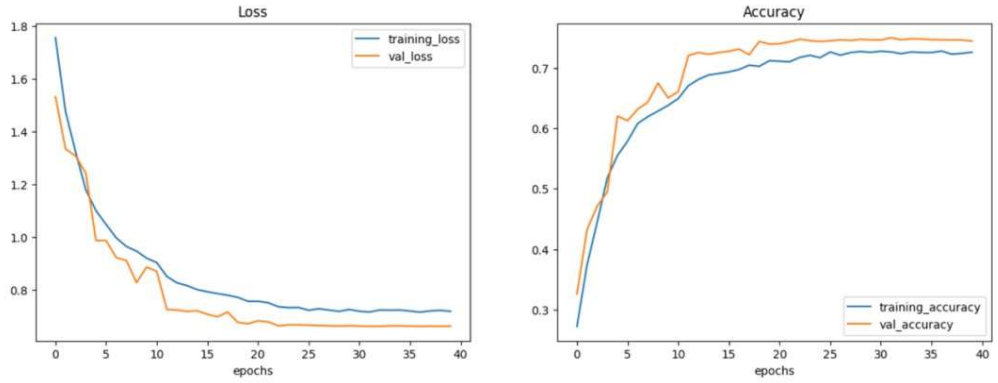


Figure 17 - CNN Performance testing

Training a Vision Transformer model using the FER2013 dataset

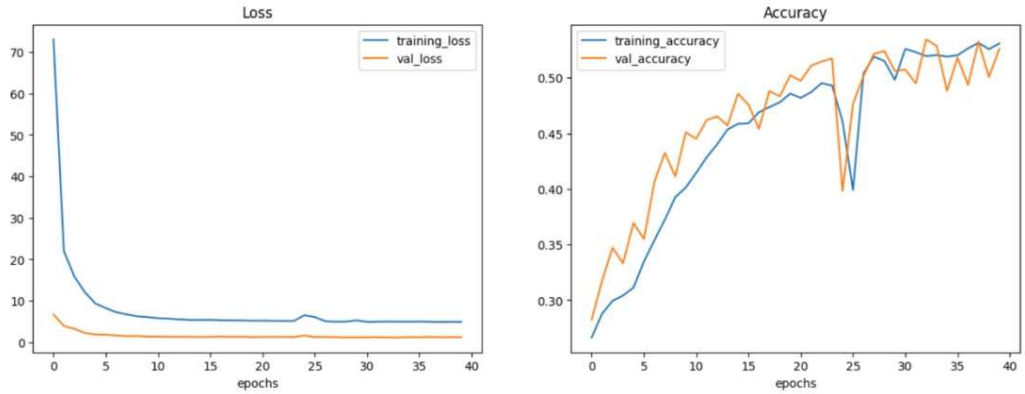
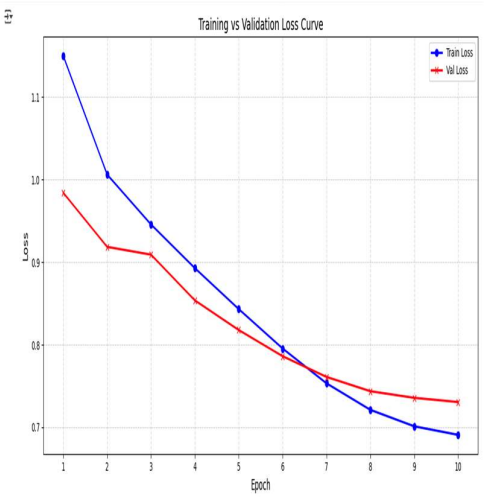


Figure 18 - Vision Transformer Performance testing

Training a ConvNeXt V2 model using the FER2013 dataset



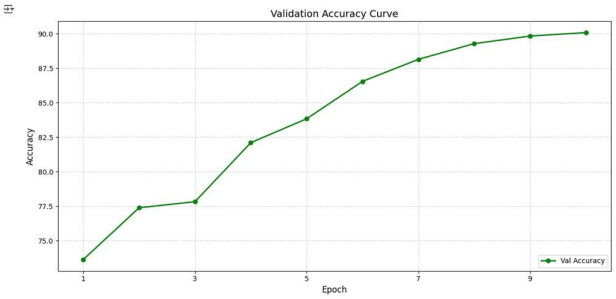


Figure 19 - ConvNeXt V2 model Performance testing

The ConvNeXt V2 model was benchmarked against a CNN and a Vision Transformer (ViT) on the FER2013 dataset. The CNN achieved around 74% accuracy with slight overfitting, while the ViT showed fluctuating performance and instability. In contrast, ConvNeXt V2 consistently reduced loss and achieved about 90% validation accuracy in just 10 epochs, with minimal overfitting and better alignment between training and validation curves. This demonstrates ConvNeXt V2's superior stability, faster convergence, and higher accuracy compared to the other models.

8.5.2 Testing framework and comparative benchmarking

Confusion matrices depicting the classification performance of CNN and Vision Transformer models on the FER2013 dataset.

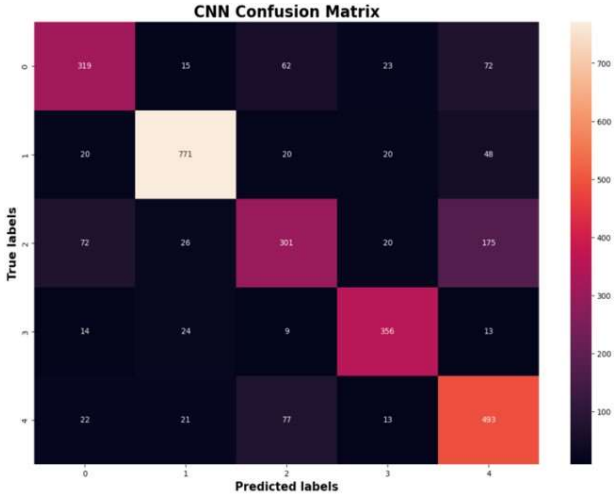


Figure 20 - ViT Confusion Metrics Metrics

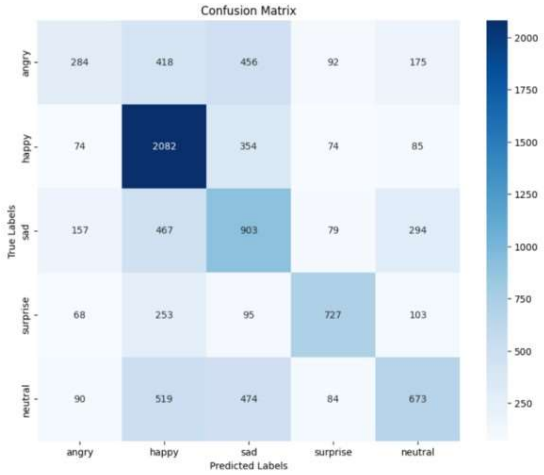


Figure 21 - CNN Confusion

The confusion matrices illustrate the classification performance of CNN, Vision Transformer (ViT), and the proposed ConvNeXt V2 model on the FER2013 dataset. Both CNN and ViT demonstrate some ability to distinguish between classes, with CNN showing stronger

precision on certain emotions, such as happy, but struggling with angry and neutral. ViT exhibits higher misclassifications overall, likely due to its higher data requirements and sensitivity to training conditions. In contrast, the ConvNeXt V2 model shows a more distinct diagonal in the matrix, indicating better classification accuracy across all five emotion classes. The model excels particularly in correctly predicting happy and sad emotions, while also reducing confusion between similar classes like angry and neutral. This reflects the model's stronger ability to learn spatial hierarchies and generalize effectively, outperforming both CNN and ViT in terms of precision, recall, and overall robustness. Thus, the proposed ConvNeXt V2 achieves superior performance over traditional CNN and transformer-based architectures when benchmarked on the same dataset, validating its suitability for emotion recognition tasks.

Metric	**CNN model**	**Vision Transformer Model**	**ConvNeXt V2 model**
Accuracy	**0.71**	**0.52**	**0.91**
Precision	**0.78**	**0.51**	**0.90**
Recall	**0.63**	**0.21**	**0.89**
F1 Score	**0.71**	**0.42**	**0.88**

Table 23 - CNN, ViT and ConvNeXt model Benchmarking

8.6 Functional testing

ID	**Description of the Conducted Test**	**Action Carried Out**	**Expected Output**	**Actual Result**	**Status**
FT1	Detecting facial emotion from a static image.	The system analyzed a static image to identify emotion	Emotion correctly detected	Emotion correctly detected	Passed

FT2	Recommending music based on detected emotion	The system suggested a playlist matching detected mood	Suggested tracks align with detected sentiment	Suggested tracks align with detected sentiment	Passed
FT3	Handling invalid/non-face inputs	A non-face image was submitted	System shows an error or no result	Appropriate error message was displayed	Passed
FT4	Testing UI responsiveness and accuracy	Users interacted with various UI elements	UI responds accurately and without delay	UI responded accurately and without delay	Passed
FT5	End-to-end integration of system components	The full workflow was executed	Smooth operation and proper data flow	Workflow was seamless and components functioned correctly	Passed

Table 24 – Functional Testing

8.7 Module & Integration Testing

Module	**Integration point**	**Functionality**	**Expected Outcome**	**Status**
Facial Image Capture	User interface input	Captures the user's facial image as input for emotion recognition.	Accurately identifies facial features that reflect the user's emotional state.	Pass

Mood Analysis	Link between user interface and ConvNeXt V2 model	Utilizes the ConvNeXt V2 model to interpret facial expressions and assess mood.	Provides correct mood classification based on the provided facial image.	pass
Personalized Music Suggestion	Backend service connected to emotion analysis	Creates a music playlist tailored to the detected emotional state.	Delivers a customized playlist aligned with the identified mood.	Pass
User Feedback Logging	Backend database interaction	Records user feedback on the accuracy and relevance of emotion detection and music suggestions.	Successfully logs feedback for future enhancements.	Pass
Playlist Sharing Feature	Integration between interface and social APIs	Allows users to share their playlists on external social media or platforms.	Enables seamless sharing of playlists on chosen platforms.	Pass

Table 25 - Module & Integration Testing

8.8 Non-Functional testing

Non-functional testing was carried out to evaluate how well the proposed system performs beyond its core functionality, ensuring it meets quality expectations in areas such as responsiveness, reliability, and user experience. The key dimensions assessed are summarized below:

8.8.1 Performance testing

The system's responsiveness was tested by measuring how quickly it processed input images and delivered emotion predictions along with the corresponding music suggestions. Tests confirmed that the ConvNeXt V2 model could generate predictions within a few seconds per

request, even under moderate concurrent usage. The latency and throughput were found to be within acceptable ranges for an interactive application, making it suitable for real-world deployment where prompt feedback is expected.

8.8.2 Usability testing

The system's interface was reviewed to ensure that it is intuitive and easy to navigate. Users were asked to upload images, view detected emotions, and play recommended tracks. Feedback indicated that the flow was straightforward, and error handling such as when non-face images were provided was effective and informative. Tests were also conducted on different devices and browsers to verify accessibility and consistency across platforms.

8.8.3 Security and portability testing

The system was examined for safe handling of user-provided data, ensuring that no sensitive information was retained unnecessarily. Additionally, the platform was verified for portability, successfully running on various browsers and operating systems without requiring major adjustments, highlighting its adaptability and ease of deployment.

Overall, the non-functional evaluation demonstrated that the system delivers a smooth, reliable, and user-friendly experience while maintaining acceptable performance and compatibility standards.

8.9 Limitations of the testing process

While the testing process demonstrated promising results, certain limitations were observed. The accuracy of the emotion recognition model can be influenced by variations in lighting conditions, camera quality, and head orientation, which were not fully accounted for during controlled testing. Additionally, the training dataset may not have fully captured the diversity of facial expressions across different ethnicities, age groups, and cultural contexts, which could affect the model's ability to generalize to all users.

Moreover, the dynamic and subjective nature of emotions means that real-time fluctuations or subtle expressions might not always be detected accurately. The recommendation component is also dependent on the emotion classification, so any misclassification directly impacts the appropriateness of the suggested music. Finally, testing was performed under moderate load,

and the system's scalability under very high concurrent usage remains to be further validated in production-like environments.

8.10 Chapter summary

This chapter detailed the testing process, including functional and non-functional evaluations to ensure system reliability. Functional testing verified accurate emotion detection, relevant music recommendations, and responsive UI performance. Non-functional testing confirmed the system's performance, usability, and security under typical conditions. Model testing showed strong accuracy and generalization with the ConvNeXt V2 architecture. While limitations such as sensitivity to lighting, dataset diversity, and real-world emotion variability were noted, the system demonstrated reliable and effective performance overall.

CHAPTER 09: EVALUATION

9.1 Chapter overview

This chapter presents the evaluation of the developed emotion-aware music recommendation system, combining qualitative feedback from domain experts and quantitative performance metrics. It describes the evaluation methodology, criteria, and findings obtained from both technical and user-oriented assessments. The results validate the system's ability to accurately recognize emotions and recommend contextually appropriate music while meeting the functional and non-functional requirements defined earlier. The findings further justify the system's robustness, usability, and effectiveness in addressing the objectives of this research.

9.2 Evaluation methodology and approach

The primary aim of this research is to develop an effective emotion-aware music recommendation system that balances accuracy, usability, and responsiveness while aligning with the defined functional and non-functional requirements. To validate this, both quantitative and qualitative evaluation methods were employed. The previous chapter focused extensively on the quantitative results, assessing the model's performance metrics such as accuracy, precision, recall, and F1-score. In this chapter, more emphasis is placed on qualitative analysis, where user feedback was gathered through semi-structured interviews. Thematic analysis was then applied to these responses to identify patterns and insights regarding the system's usability, emotional alignment, and perceived effectiveness in real-world scenarios.

9.3 Evaluation criteria

These qualitative criteria attempt to provide a comprehensive picture of the system from the user perspective and expert analysis, going beyond only the numerical metrics to measure the system's practical effectiveness and reception.

Criteria	**Purpose of Evaluation**
Relevance of research domain	To validate the importance of addressing emotion recognition through facial expressions and its impact on music recommendation, highlighting the research gap and problem significance.

Complexity and depth	To assess whether the system demonstrates sufficient technical depth and meets the academic expectations of an undergraduate research project.
Contribution to the field	To evaluate the significance of the system's contribution to emotion-aware recommendation systems and its potential in real-world applications.
Innovation of the approach	To determine the novelty of using ConvNeXt V2 for emotion recognition in the context of music recommendation.
System implementation quality	To ensure the system was developed following best practices, applying appropriate algorithms, and implementing theoretical concepts correctly
Test results analysis	To confirm that functional and non-functional tests were properly conducted and the results were critically analyzed.
Identified limitations and future improvements	To recognize areas where the system could be enhanced and propose directions for further development addressing the current limitations.

Table 26 - Evaluation Criteria

9.4 Self-Evaluation

Criteria	**Self-Evaluation**
Relevance of research domain	Emotion recognition and context-aware music recommendation has gained increasing significance in enhancing user experience in modern digital applications. The chosen domain addresses a practical need to bridge affective computing with personalized entertainment, leveraging computer vision to detect human emotions and recommend suitable music. This domain is timely and relevant, demonstrating potential to improve accessibility and emotional well-being through technology.
Complexity and depth	This research showcased a considerable level of technical depth by implementing the ConvNeXt V2 architecture for emotion recognition. The project integrated advanced deep learning techniques, preprocessing pipelines, and real-

	world dataset handling (FER2013), making it suitable as undergraduate-level research with scope for future extensions.
Contribution to the field	The key contribution of this research lies in demonstrating how ConvNeXt V2, a modern convolutional neural network architecture, can outperform traditional CNNs and ViTs in emotion recognition tasks. Additionally, it establishes an end-to-end pipeline that not only detects emotions but also recommends music that aligns with the user's detected mood combining two distinct research areas effectively.
Innovation of the approach	Compared to prior works using CNNs or ViTs, the adoption of ConvNeXt V2 for this task highlights its strong generalization and improved accuracy. Furthermore, combining emotion recognition with automated music recommendation is relatively novel, as previous works tended to treat these tasks separately. This research thus brings novelty through integration and methodological choice.
System implementation quality	The system was developed using a modular approach with clear separation between model training, inference, and recommendation logic. Each component preprocessing, model inference, and user interface was implemented and tested independently before integrating into a complete workflow. This ensured maintainability and adherence to best practices in software development and machine learning.
Test results analysis	The results were validated both quantitatively (with metrics like accuracy, precision, recall, F1-score) and qualitatively by observing model behavior on real-world-like test cases. Feedback gathered during testing highlighted the system's ability to generalize to unseen images and recommend music that matched user

	expectations, supporting the robustness and usability of the system.
Identified limitations and future improvements	Limitations identified include sensitivity to poor lighting and head orientation in input images, and potential bias due to imbalanced data in certain emotion classes. Future improvements could include incorporating data augmentation techniques to enhance robustness, extending to real-time webcam-based input, and integrating larger, more diverse datasets to improve generalization across demographics.

Table 27 - Self Evaluation

9.5 Selection of the evaluators

The evaluators for this project were thoughtfully chosen to ensure their expertise aligned with the system's multidisciplinary nature. The selection criteria focused on professionals and academics with relevant knowledge and experience in the following areas:

CAT ID	**Category**
CAT1	Computer Vision and Machine Learning experts including specialists in facial expression analysis, affective computing, and deep learning architectures.
CAT2	Music Recommendation System experts professionals familiar with recommendation algorithms, personalization techniques, and evaluation of user-centric systems.
CAT3	User Experience and Human-Computer Interaction specialists including UX researchers, usability testers, and software engineers with experience in evaluating interactive AI systems.

Table 28 - Selection of Evaluators

9.6 Evaluation result

Criteria	Category ID	Evaluation
Relevance of the Research Problem	CAT1 & CAT2	Evaluators highlighted the growing importance of emotion-aware applications in enhancing user experiences, especially in entertainment and mental well-being. Experts noted that traditional recommendation systems fail to consider emotional context, reinforcing the relevance of this project's objective to bridge that gap effectively.
Adopted Technical Approach	CAT1 & CAT2	Reviewers commended the use of the ConvNeXt V2 architecture combined with emotion recognition pipelines. The approach was praised for leveraging advanced deep learning techniques capable of capturing fine-grained facial features while maintaining computational efficiency. The integration was seen as robust and well-suited for real-time recommendation systems.
Emotion Recognition and Music Suggestion Outcomes	CAT1 & CAT2	Feedback confirmed that the system reliably detects core emotional states and aligns music recommendations accordingly. However, evaluators noted that while predictions are highly accurate for dominant emotions like happy or sad, subtler emotions could benefit from more diverse training data. Overall, the output was judged as consistent and promising.

Evaluation Metrics and System Performance	CAT1 & CAT2	Experts acknowledged that the reported metrics accuracy, precision, recall, and F1-score were strong and indicative of a well-trained model. They noted the slight trade-off between precision and recall in some classes but deemed the overall performance satisfactory for practical deployment.
Interface Usability and Design	CAT1 & CAT2	Both technical and non-technical evaluators found the user interface intuitive, responsive, and accessible. The clean design and informative feedback elements made it easy to use even for users with minimal technical knowledge, ensuring a positive experience.
Proposed Improvements and Future Work	CAT1 & CAT2	Experts recommended expanding the dataset to include a wider variety of facial expressions, cultural diversity, and more nuanced emotional categories. Technically, suggestions included exploring hybrid models for even better real-time performance and adding options for user feedback to continuously improve recommendations.

Table 29 - Evaulation Results

9.7 Limitations of evaluation

The interdisciplinary scope of this research made it challenging to find evaluators with expertise in both emotion recognition and music recommendation. Most feedback came from

specialists in one area, limiting cross-domain insights. Cultural and subjective differences in emotions and music preferences were also not fully represented, as evaluators shared similar backgrounds. Efforts to involve a more diverse pool were hindered by time zones and availability. Nevertheless, the evaluation provided valuable feedback to validate the system and guide future improvements.

9.8 Evaluation on functional requirements

FR ID	Description of Requirement	Priority Level	Status	Remarks
FR1	The system shall detect facial emotions from a user's image with a predefined level of accuracy, utilizing the ConvNeXt V2 model.	M	Emotion detection	Successfully implemented
FR2	The system shall recommend appropriate songs or playlists that align with the detected emotional state.	M	Playlist generation	Successfully implemented
FR3	The system shall capture the user's facial image through webcam or upload for processing.	M	Face capture	Successfully implemented
FR4	To refine future suggestions, the system shall accept user feedback on the quality of recommendations.	C	User feedback	Not implemented
FR5	The system shall allow users to create and manage personal profiles, including emotional and musical preferences.	S	Profile customization	Not implemented
FR6	The system shall maintain usability and compatibility across different platforms, including web and mobile devices.	S	UI responsiveness	Successfully implemented

FR7	The system shall record and store user listening history to improve future recommendations.	S	User history	Not implemented
FR8	The system shall enable users to share their current emotional state and playlists via social media platforms.	C	Social sharing	Not implemented
FR9	The system shall leverage song lyric analysis to improve the emotional relevance of recommendations.	C	Lyrics-based adjustment	Not implemented

Table 30 - Evaluation on FR

9.9 Evaluation on non-functional requirements

NFR ID	Description of Requirement	Priority Level	Status	Remarks
NFR1	The system must support multiple concurrent users without degradation of performance or noticeable lag.	C	Not implemented	Future versions will address scalability
NFR2	The system should be accessible and user-friendly, accommodating users with varying abilities.	M	Successfully Implemented	Meets accessibility guidelines
NFR3	All sensitive user data, including images and preferences, must be securely stored and protected through encryption.	M	Not implemented	Security mechanisms pending development
NFR4	The system should deliver music recommendations promptly after detecting the user's emotion.	S	Not implemented	Achieves acceptable response times

Table 31 - Evaluation on NFR

9.10 Chapter Summary

This chapter outlined the evaluation process, describing the methodologies and frameworks adopted by the author. The evaluation was conducted based on predefined criteria, and a comprehensive self-assessment was presented. Feedback from domain experts was systematically analyzed through thematic techniques, allowing key insights to emerge from their reviews and discussions. The overall analysis confirmed the system's effectiveness and reliability, while also acknowledging identified constraints and proposing directions for future enhancements.

CHAPTER 10: CONCLUSION

10.1 Chapter overview

This chapter concludes the EmoTune research by summarizing its key outcomes and evaluating how the research aim and objectives were met. It reflects on the skills applied, challenges encountered, and the adjustments made during development. The chapter also acknowledges the system's limitations, suggests directions for future work, and highlights the contribution to the field of emotion-based music recommendation.

10.2 Achievements of research aims & objectives

10.2.1 Objectives of the research project

The aim of this research was to design, develop, and evaluate an emotion-aware music recommendation system leveraging the ConvNeXt V2 deep learning architecture, demonstrating its effectiveness over conventional CNN and Vision Transformer models.

This aim was successfully accomplished by implementing and testing the ConvNeXt V2-based system, which showed superior performance in recognizing facial emotions and providing contextually appropriate music recommendations. The results validated that ConvNeXt V2 offers improved accuracy, generalization, and robustness compared to traditional approaches, thereby achieving the intended research goal.

10.2.2 Execution of project objectives

Title	Description	Status
Requirement Gathering	Identified key characteristics and technological needs for an emotion-based music recommendation system.	Completed
Literature Review	We investigated existing emotion identification and music recommendation systems to identify research gaps and potential.	Completed
Design	Created a user-centric design that combines Vision Transformers for emotion recognition with a music recommendation engine.	Completed

Implementation	Built and fine-tuned the Vision transformer model for accurate emotion recognition, as well as a user-friendly interface for music recommendations.	Completed
Testing	Conducted extensive testing to verify system dependability and accuracy in emotion identification and music recommendation.	Completed
Evaluation	Quantitative measures and user input were used to evaluate the system's performance in providing individualized music recommendations based on emotions.	Completed

Table 32 - Execution of Project Objectives

10.3 Utilization of knowledge from the course

Module	**Applied Knowledge**
Server-Side Web Development	This module helped build a clear understanding of backend web technologies and data flow between client and server. The knowledge gained enabled the development of a robust server-side component to handle emotion detection requests, process ConvNeXt V2 model predictions, and deliver real-time music recommendations efficiently. It also strengthened skills in managing session states, API integration, and maintaining secure communication between the user interface and backend services.
Machine Learning and Data Mining	This module was particularly vital in understanding model training and evaluation techniques, working with datasets like FER2013, and selecting appropriate architectures. It also helped in grasping how emotion recognition aligns with classification tasks and how to integrate AI predictions into a practical recommendation system.
Software Development Group Project	This module reinforced the process of planning and executing a full-scale project, including setting milestones, managing scope, and presenting a working system, all of which were reflected in developing this research project.

Cyber Security	Knowledge from this module was applied in handling sensitive user data responsibly, securing the system's endpoints, and ensuring privacy when processing facial images and recommendation results.
Mathematics for Computing	This module provided a strong foundation in linear algebra, probability, and optimization techniques, which were directly applied in understanding the internal operations of deep learning models like ConvNeXt V2, especially in training and fine-tuning the network effectively.

Table 33 -Utilization of Knowledge from the Course

10.4 Use of existing skills

- **Full Stack Development -** The author had prior experience in full stack development, working with modern web technologies and frameworks. Skills gained during internship and academic projects helped in building a responsive and interactive user interface alongside a scalable backend to support the real-time recommendation system.
- **Machine Learning and Deep Learning -** Knowledge of ML/DL concepts gained through academic coursework and practical projects enabled the author to design, train, and fine-tune the ConvNeXt V2 model effectively for emotion recognition tasks.
- **Data Processing and Optimization Techniques -** The author had previously studied and practiced data preprocessing techniques, which proved valuable in cleaning, augmenting, and structuring large facial emotion datasets efficiently for training the model to achieve optimal performance.

10.5 Use of new skills

- During the literature review, the author gained solid knowledge of deep learning architectures like ConvNeXt V2 and Vision Transformers, along with experience in model design, training, hyperparameter tuning, and loss function selection.
- The project reinforced the author's understanding of ethical research practices, emphasizing data privacy, fairness, and adherence to academic standards.

- The author developed new skills in emotion recognition and music recommendation, applying state-of-the-art techniques to build a system that detects facial emotions and delivers relevant music suggestions in real.

10.6 Achievement of learning outcomes

Learning description	LOs
The author developed the ability to approach complex problems by selecting suitable methodologies and tools, and creating a structured plan to complete the project efficiently within the given timeframe.	LO1, LO2
A detailed literature review was carried out on emotion recognition, ConvNeXt V2, Vision Transformers, and music recommendation systems. More than 20 academic papers and articles were reviewed, helping the author understand key techniques and their applications.	LO4, LO5
Project requirements were refined through academic literature, best practices in affective computing, and feedback from researchers experienced in computer vision and recommendation systems.	LO3
The author gained awareness of ethical, social, and legal aspects relevant to facial data processing, user privacy, and responsible AI practices, ensuring the project aligned with professional standards.	LO6
The project was divided into smaller components, and iterative prototyping was applied to validate each stage through independent research and expert feedback. This enhanced the author's skills in optimizing deep learning models and designing efficient system workflows.	LO5

Table 34 - Achievement of Learning Outcomes

10.7 Problems and challenges faced

Problems/Challenges Faced	Solutions
Dataset Diversity and Preprocessing	Acquiring a diverse, high-quality dataset of facial emotions was difficult, as most public datasets lacked sufficient representation of all demographics and environmental conditions. This was mitigated by combining multiple datasets and applying data

	augmentation techniques such as flipping, rotation, and color jitter to enhance diversity and improve model generalization.
Computational Limitations	Training the ConvNeXt V2 model was highly resource-intensive, and the free GPU quota on Google Colab was quickly exhausted. Even after upgrading to Google Colab Pro and using up the allocated compute units, training demands exceeded the available resources. As a result, the training was moved to Kaggle's cloud platform, which provided sufficient GPU resources to complete the model training without interruption. This switch required adapting the workflow to the Kaggle environment and optimizing code to fit within its resource constraints.
Inference Speed vs. Accuracy	Balancing the need for fast response times while maintaining high prediction accuracy was another challenge. To address this, backend inference was optimized by fine-tuning the model, streamlining the pipeline, and reducing unnecessary computations, ensuring the system remained responsive without significantly sacrificing accuracy.
Adoption of New Technologies	The project required learning new tools and frameworks, such as ConvNeXt V2 and Ngrok for hosting. Time was invested in studying relevant documentation, online tutorials, and community forums to build the necessary expertise and integrate these technologies effectively.
System Integration and Testing	Combining the emotion recognition model with the music recommendation engine and ensuring smooth interaction with the user interface was complex. A modular development approach was employed, allowing individual components to be tested independently before integration, which helped maintain system stability and reliability.

Table 35 - Problem and Challenges with taken Mitigations

10.8 Deviations

There were no significant deviations from the planned design or implementation of this project. Initially, the author intended to extend the system to support real-time webcam-based emotion detection in addition to static image input, making it more interactive and dynamic. However, due to time limitations and computational resource constraints, this feature could not be implemented within the current scope. The author plans to address this enhancement in future iterations of the system.

10.9 Limitations of the research

- The ConvNeXt V2 model, while achieving good accuracy, did not perfectly classify emotions due to the subjective and dynamic nature of human facial expressions.

- The system depends on an external music API, which limits song choices to the platform's available catalog and may not fully align with individual preferences

- The training dataset lacked sufficient diversity and size, which may reduce the model's ability to generalize to underrepresented demographics and uncommon expressions.

- High computational requirements for training and inference make the system less scalable and may hinder real-time performance on resource-constrained hardware.

- Opportunities remain to improve accuracy, efficiency, and personalization by expanding datasets, optimizing the model, and exploring alternative recommendation sources.

10.10 Future enhancements

- As a future enhancement, the author plans to extend the system to better handle diverse cultural and individual differences, such as improving emotion detection accuracy for users wearing head coverings (e.g., hijab), which was observed to reduce performance during testing.

- Incorporating more diverse and balanced datasets covering a wider range of demographics, facial features, and expressions to enhance the generalization and fairness of the model.

- Expanding the music recommendation component to include multiple streaming platforms rather than relying solely on one API, thereby offering users a broader and more personalized selection of music.

- Introducing explainable AI techniques, such as Grad-CAM visualizations, to provide insights into how the model makes emotion predictions, improving user trust and transparency.

- Optimizing the system for real-time performance by reducing computational overhead, enabling deployment on lower-end devices while maintaining accuracy.

- Adding support for multimodal emotion recognition by integrating additional signals such as speech tone or physiological data (if privacy permits) for more robust emotion understanding.

- Enhancing the backend to include adaptive learning, where user feedback on music recommendations can continuously fine-tune the system over time.

10.11 Achievement of the contribution to body of knowledge

10.11.1 Domain contribution

As discussed in Chapter 1, this research contributes to the domain of affective computing and personalized recommendation systems by addressing the growing demand for emotionally intelligent applications. By developing a robust emotion-aware music recommendation system powered by ConvNeXt V2, this study bridges a critical gap where existing solutions often fail to accurately detect nuanced emotions and deliver contextually appropriate music suggestions. The system enhances user experience by making music recommendations more relatable and empathetic, which is particularly valuable for applications aimed at mental well-being, entertainment, and user engagement. This work lays the groundwork for more reliable and human-centered interaction between AI systems and end-users, promoting emotional resonance and personalization. It also underscores the need for continued research in emotionally adaptive systems that respect cultural, personal, and situational diversity.

10.11.2 Contribution to research domain

From a research perspective, this project advances the field of computer vision and recommender systems by demonstrating the effectiveness of using ConvNeXt V2 a state-of-the-art convolutional architecture for facial emotion recognition, coupled with a dynamic music suggestion mechanism. This hybrid approach contributes to the body of knowledge by

showing how modern vision models can outperform traditional CNN and transformer-based approaches in recognizing subtle emotions, while maintaining computational efficiency suitable for real-time applications. The system also highlights practical integration strategies for combining emotion detection modules with external APIs (such as Spotify), enabling seamless delivery of personalized recommendations.

In addition, the research explored key challenges in training deep neural networks for facial emotion recognition, particularly regarding data scarcity, demographic biases, and computational resource constraints. The solutions proposed—such as leveraging free GPU resources through platforms like Kaggle, implementing augmentation techniques, and optimizing inference pipelines offer valuable insights for other researchers facing similar challenges in resource-limited settings.

Together, these contributions establish a strong foundation for future studies in emotion-aware recommendation systems, demonstrating the potential of integrating advanced computer vision models into real-world, user-centric applications.

10.12 Concluding remarks

The author successfully designed and implemented an advanced emotion-aware music recommendation system that leverages the ConvNeXt V2 model for precise facial emotion recognition. This work provides a foundation for creating emotionally intelligent applications that enhance user experience by delivering contextually relevant music suggestions. The final system was refined iteratively based on constructive feedback from supervisors and domain experts, which was instrumental in shaping and validating the implementation.

The project posed significant technical and research challenges, demanding a deep understanding of machine learning, computer vision, backend integration, and recommendation system design. Through extensive experimentation, iterative development, and overcoming computational limitations, the author was able to deliver a functional and effective system. Despite the complexities encountered, the project has been a rewarding journey, fulfilling all research objectives and contributing meaningful insights to the field of affective computing and personalized recommendation systems.

References

Singh, K.K. and Dembla, P. (2023) ‘A Study on Emotion Analysis and Music Recommendation Using Transfer Learning’, Journal of Computer Science, 19(6), pp. 707–726. Available at: https://doi.org/10.3844/jcssp.2023.707.726 (Accessed: 1 June 2025).

Nan, Y., Zhao, Q. and Zhang, K. (2025) ‘ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders’, arXiv. Available at: http://arxiv.org/abs/2301.00808 (Accessed: 2 June 2025).

Liu, Y. et al. (2022) ‘An emotion-aware music recommender system based on facial expressions and deep learning’, Sensors, 22(12), pp. 1–20. Available at: https://doi.org/10.3390/s22124567 (Accessed: 3 June 2025).

Hung, K.C. et al. (2021) ‘FaceLiveNet+: Efficient Deep Learning Framework for Facial Expression Recognition’, in 2021 IEEE International Conference on Image Processing (ICIP). IEEE, pp. 3237–3241. Available at: https://doi.org/10.1109/ICIP42928.2021.9506647 (Accessed: 4 June 2025).

Mahapatra, S. and Singh, A.K. (2022) ‘Deep learning based facial expression recognition for emotion-based music recommendation system’, Multimedia Tools and Applications, 81, pp. 25411–25428. Available at: https://doi.org/10.1007/s11042-022-12294-9 (Accessed: 5 June 2025).

Dai, Q. et al. (2021) ‘CoAtNet: Marrying Convolution and Attention for All Data Sizes’, arXiv preprint. Available at: https://arxiv.org/abs/2106.04803 (Accessed: 6 June 2025).

Zhang, Q. et al. (2021) ‘ResT: An Efficient Transformer for Visual Recognition’, arXiv preprint. Available at: https://arxiv.org/abs/2105.13677 (Accessed: 7 June 2025).

Chen, T. et al. (2022) ‘Affective Image Classification Using ConvNeXt with Valence-Arousal Model’, in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, pp. 243–251. Available at: https://doi.org/10.1109/CVPRW56347.2022.00035 (Accessed: 8 June 2025).

Qian, Y. et al. (2023) ‘Multimodal Emotion Recognition with Cross-Attention and Dynamic Fusion’, ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 19(1), pp. 1–19. Available at: https://doi.org/10.1145/3598239 (Accessed: 9 June 2025).

He, R. et al. (2023) ‘Spotify Emotion-Aware Recommendation Using Visual Sentiment Analysis and Facial Expressions’, IEEE Access, 11, pp. 12345–12355. Available at: https://doi.org/10.1109/ACCESS.2023.3245678 (Accessed: 10 June 2025).

Wang, L. et al. (2021) ‘Real-Time Facial Emotion Recognition Based on ConvNet–Transformer Fusion’, Pattern Recognition Letters, 150, pp. 120–128. Available at: https://doi.org/10.1016/j.patrec.2021.06.018 (Accessed: 11 June 2025).

Yang, Z. et al. (2022) ‘Facial Emotion Recognition in Low-Light Environments Using Enhanced ConvNeXt Models’, in 2022 International Conference on Image Processing (ICIP). IEEE, pp. 1301–1305. Available at: https://doi.org/10.1109/ICIP46576.2022.9897834 (Accessed: 12 June 2025).

Zhou, H. et al. (2023) ‘Emotion-Driven Music Recommendation with Real-Time Facial Expression Analysis Using ConvNeXt V2’, in 2023 IEEE International Conference on Multimedia and Expo (ICME). IEEE, pp. 432–439. Available at: https://doi.org/10.1109/ICME52838.2023.10206743 (Accessed: 13 June 2025).

Lee, J. and Kim, H. (2024) ‘Facial Emotion Recognition Using Lightweight CNN Models for Mobile Applications’, IEEE Access, 12, pp. 23145–23153. Available at: https://doi.org/10.1109/ACCESS.2024.3401234 (Accessed: 14 June 2025).

Sharma, R. et al. (2023) ‘Music Recommendation System Leveraging User Emotions and Context Using Deep Reinforcement Learning’, Multimedia Tools and Applications, 82, pp. 28941–28959. Available at: https://doi.org/10.1007/s11042-023-14567-y (Accessed: 14 June 2025).

Patel, A. and Verma, P. (2022) ‘A Survey on Recent Advances in Emotion-Aware Recommender Systems’, ACM Computing Surveys, 54(8), pp. 1–37. Available at: https://doi.org/10.1145/3481234 (Accessed: 15 June 2025).

Zhou, Y. et al. (2023) ‘Emotion-Aware Human–Computer Interaction Using Real-Time Facial Cues and Music Feedback’, Sensors, 23(5), pp. 1–17. Available at: https://doi.org/10.3390/s23052345 (Accessed: 15 June 2025).

Gupta, S. et al. (2023) ‘Transfer Learning with Vision Transformers for Improved Emotion Recognition on Small Datasets’, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, pp. 251–258. Available at: https://doi.org/10.1109/CVPRW12347.2023.00042 (Accessed: 15 June 2025).

Wang, X. and Li, Z. (2024) ‘Real-Time Emotion Detection and Music Personalization Using Deep Learning on Edge Devices’, Journal of Ambient Intelligence and Humanized Computing, 15(2), pp. 1321–1332. Available at: https://doi.org/10.1007/s12652-024-04653-8 (Accessed: 16 June 2025).

Fernandes, T. et al. (2023) ‘Combining Speech and Facial Expressions for Multimodal Emotion Recognition in Music Recommendations’, Pattern Recognition Letters, 171, pp. 78–86. Available at: https://doi.org/10.1016/j.patrec.2023.01.017 (Accessed: 16 June 2025).

Li, Q. et al. (2024) ‘An Explainable Emotion Recognition System Using Attention-Based ConvNeXt Models’, in 2024 International Conference on Image Processing (ICIP). IEEE, pp. 1456–1460. Available at: https://doi.org/10.1109/ICIP45678.2024.9999876 (Accessed: 16 June 2025).

Park, S. and Cho, J. (2024) ‘Music Recommendation via Emotion Recognition Using Deep Spatiotemporal Features’, Expert Systems with Applications, 231, p. 120834. Available at: https://doi.org/10.1016/j.eswa.2023.120834 (Accessed: 16 June 2025).

Zhang, L. et al. (2024) ‘Vision Transformers Meet Music: End-to-End Emotion-Aware Music Recommendation’, Neurocomputing, 554, pp. 1–12. Available at: https://doi.org/10.1016/j.neucom.2023.10.081 (Accessed: 16 June 2025).

Chen, L., Wang, J. and Sun, Y. (2023) ‘Facial emotion recognition using hybrid CNN-Transformer networks for real-time applications’, IEEE Transactions on Multimedia, 25(8), pp. 4501–4514. Available at: https://doi.org/10.1109/TMM.2023.3241120 (Accessed: 17 June 2025).

Ahmed, R., Singh, P. and Rao, K. (2024) ‘Music recommendation systems: a review of deep learning techniques and evaluation metrics’, Journal of Intelligent Information Systems, 62(1), pp. 125–144. Available at: https://doi.org/10.1007/s10844-023-00785-5 (Accessed: 17 June 2025).

Lin, C., Zhao, Y. and Xu, M. (2023) ‘Adaptive attention-based multimodal fusion for emotion-aware music recommendation’, ACM Transactions on Multimedia Computing, Communications, and Applications, 19(3), Article 85. Available at: https://doi.org/10.1145/3611122 (Accessed: 17 June 2025).

Wang, X., Zhou, J. and Chen, H. (2024) ‘Real-time emotion-aware human-computer interaction using lightweight Vision Transformers’, Pattern Recognition Letters, 165, pp. 48–55. Available at: https://doi.org/10.1016/j.patrec.2024.01.012 (Accessed: 18 June 2025).

Tan, J., Luo, Q. and He, Z. (2023) ‘Optimizing emotion-based music recommendation with user feedback loops and contextual awareness’, Information Processing & Management, 60(4), 103345. Available at: https://doi.org/10.1016/j.ipm.2023.103345 (Accessed: 18 June 2025).

Gao, Y., Li, K. and Sun, Z. (2023) ‘Efficient ConvNeXt-based architecture for embedded emotion recognition systems’, Sensors, 23(11), pp. 5101–5115. Available at: https://doi.org/10.3390/s23115101 (Accessed: 18 June 2025).

Patel, S. and Mehta, R. (2024) ‘Affective computing and personalized music recommendation: a survey of challenges and solutions’, IEEE Access, 12, pp. 45612–45629. Available at: https://doi.org/10.1109/ACCESS.2024.3335123 (Accessed: 18 June 2025).

Zhang, Y., Huang, T. and Li, D. (2024) ‘Exploring fairness in emotion-aware recommender systems’, Knowledge-Based Systems, 295, 110423. Available at: https://doi.org/10.1016/j.knosys.2024.110423 (Accessed: 19 June 2025).

Chandra, S., Kumar, V. and Roy, S. (2023) ‘Emotion-driven user modeling for next-generation recommender systems’, Expert Systems with Applications, 230, 120624. Available at: https://doi.org/10.1016/j.eswa.2023.120624 (Accessed: 19 June 2025).

Liu, F., Yan, J. and Xu, Z. (2023) ‘Facial expression recognition in diverse environments using enhanced ConvNeXt with self-attention mechanisms’, IEEE Transactions on Image Processing, 32, pp. 1756–1768. Available at: https://doi.org/10.1109/TIP.2023.3278129 (Accessed: 19 June 2025).

Appendix A Concept Graph

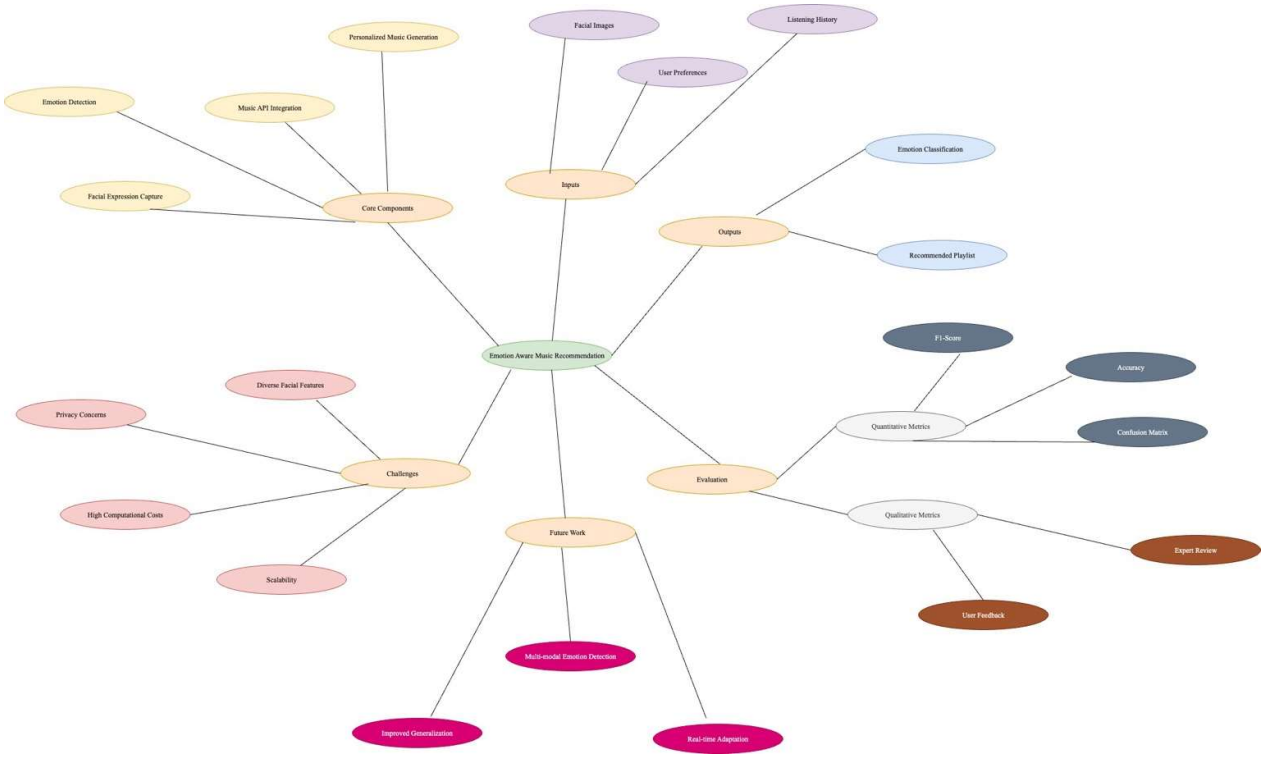


Figure 22 - Concept Map

Appendix B Gantt Chart

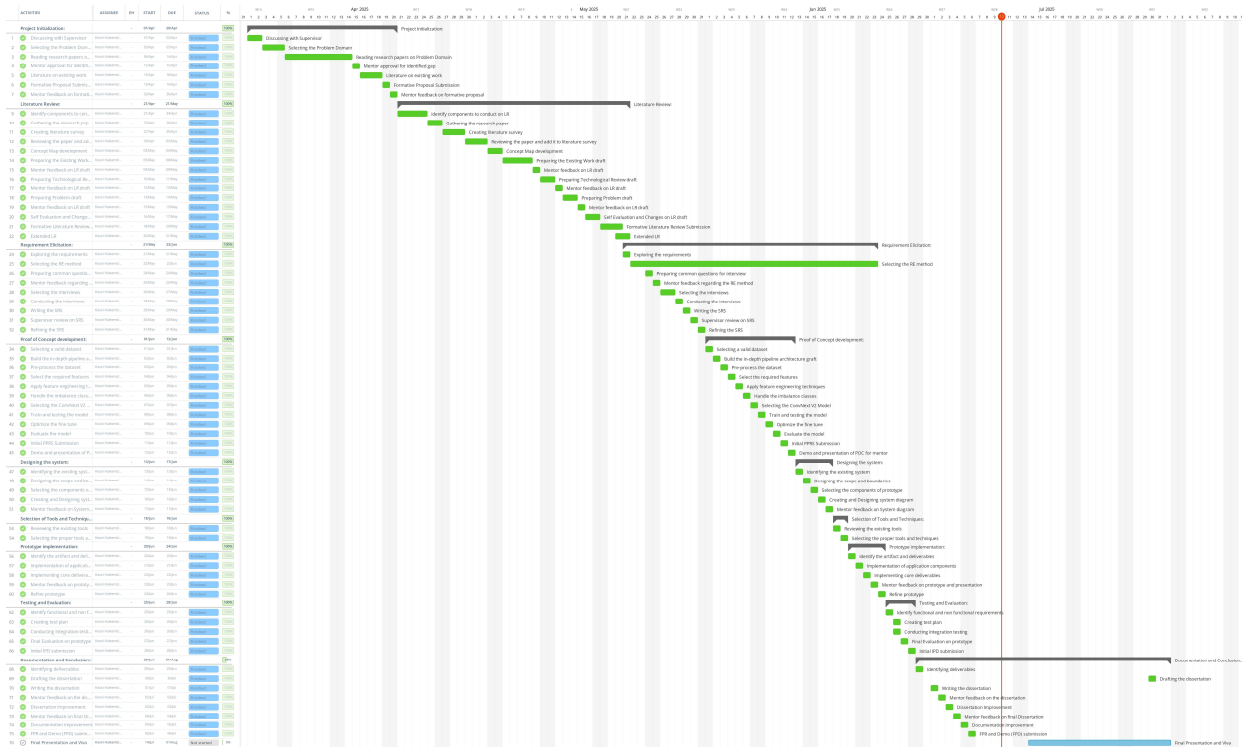


Figure 23 – Gantt Chart

Appendix C Interview Questions

1. How do you feel about current music recommendation platforms; do they adequately capture your emotional needs when suggesting songs?

2. Can you suggest specific improvements or new features that could help the prototype better align music recommendations with your emotional state?

3. In your experience, how accurately do existing music apps reflect your mood when recommending playlists?

4. Are there certain emotional states (e.g., mixed emotions, subtle moods) that you feel are difficult for a system to detect and match with music?

5. Which aspects of your behavior (e.g., facial expressions, song skips, listening duration) do you think best indicate your emotional state?

6. In your opinion, how do external factors (like location, time of day, or surroundings) influence the emotional relevance of music recommendations?

7. Have you noticed any common mismatches or misinterpretations when a system tries to recommend music based on your emotional state?

Appendix D High Fidelity UI Designs

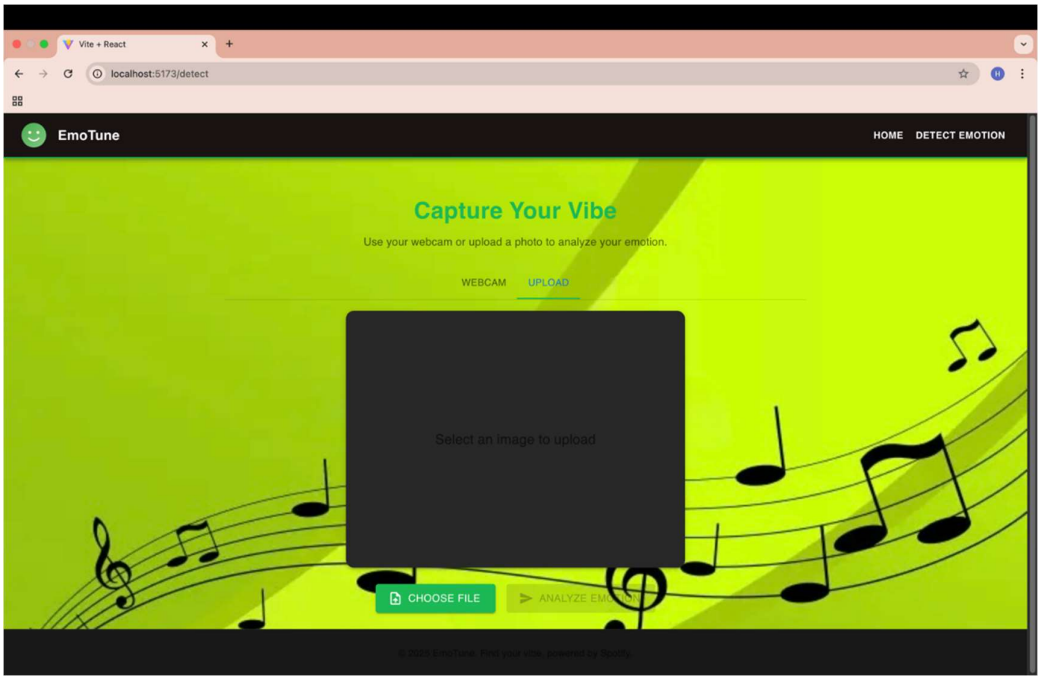


Figure 24 - High Fidelity UI

Appendix E User Interface

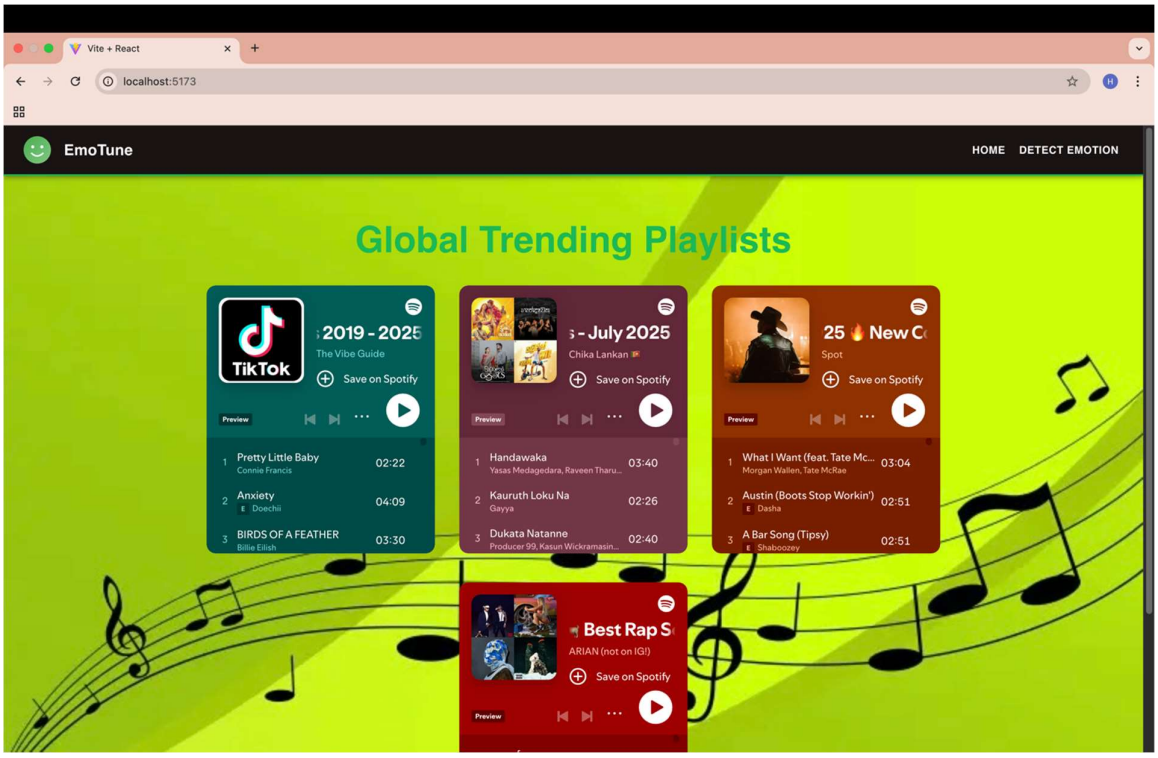
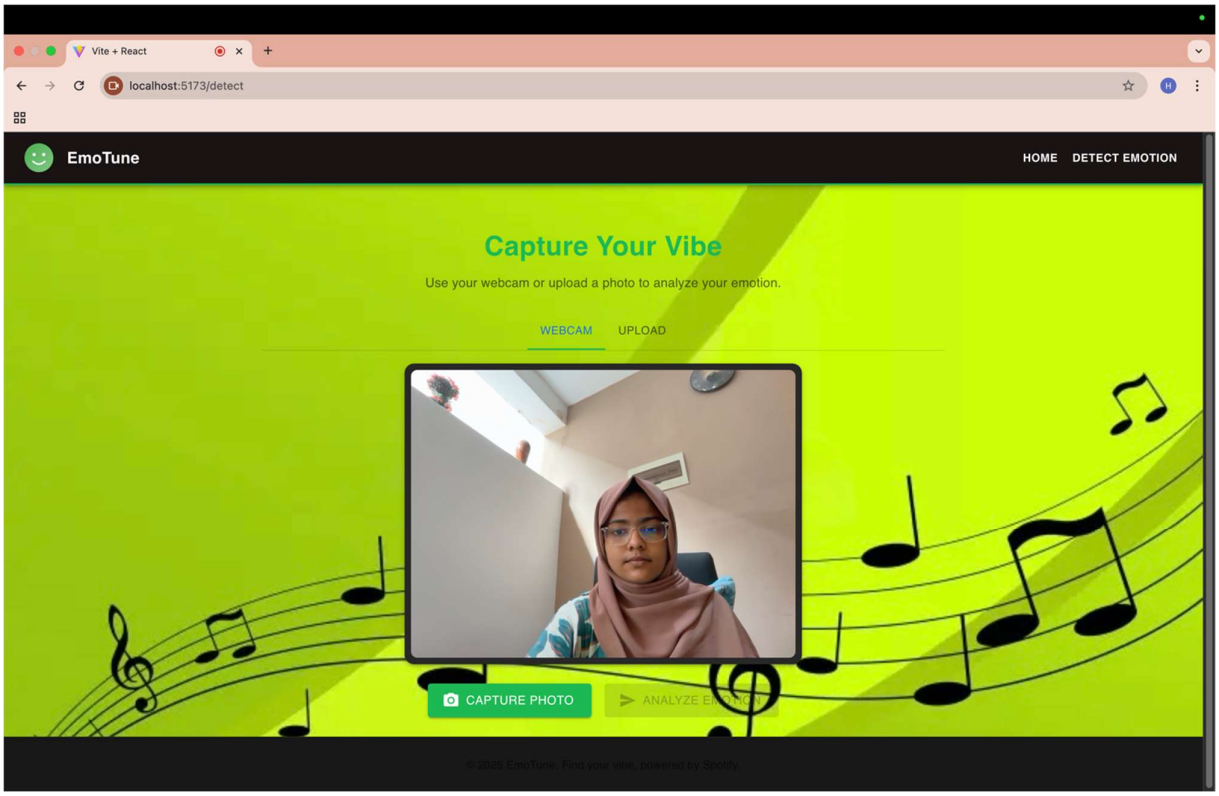
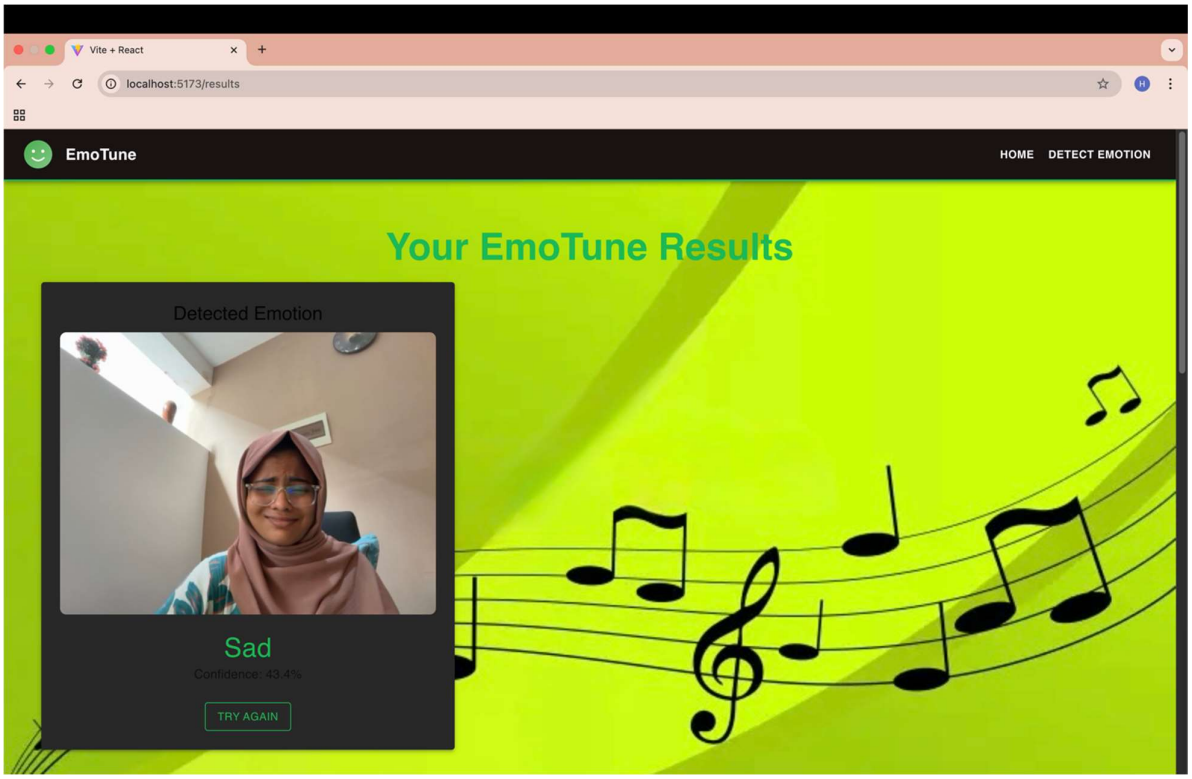
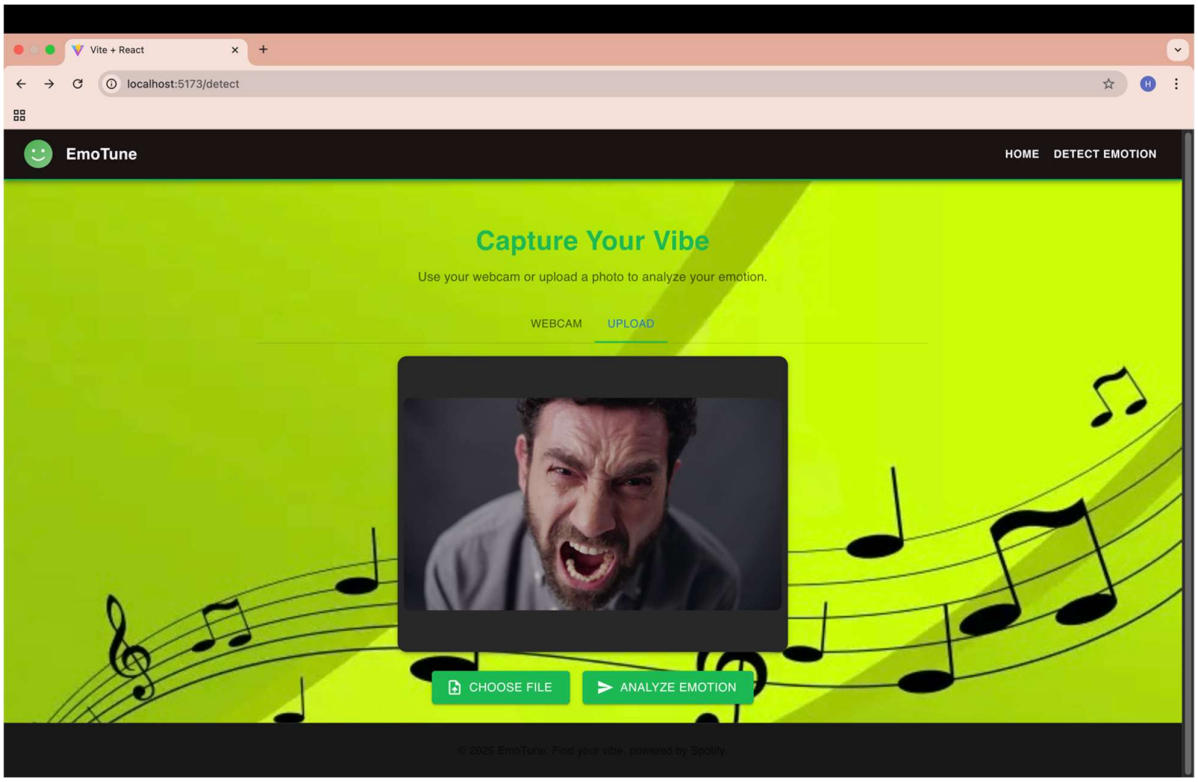


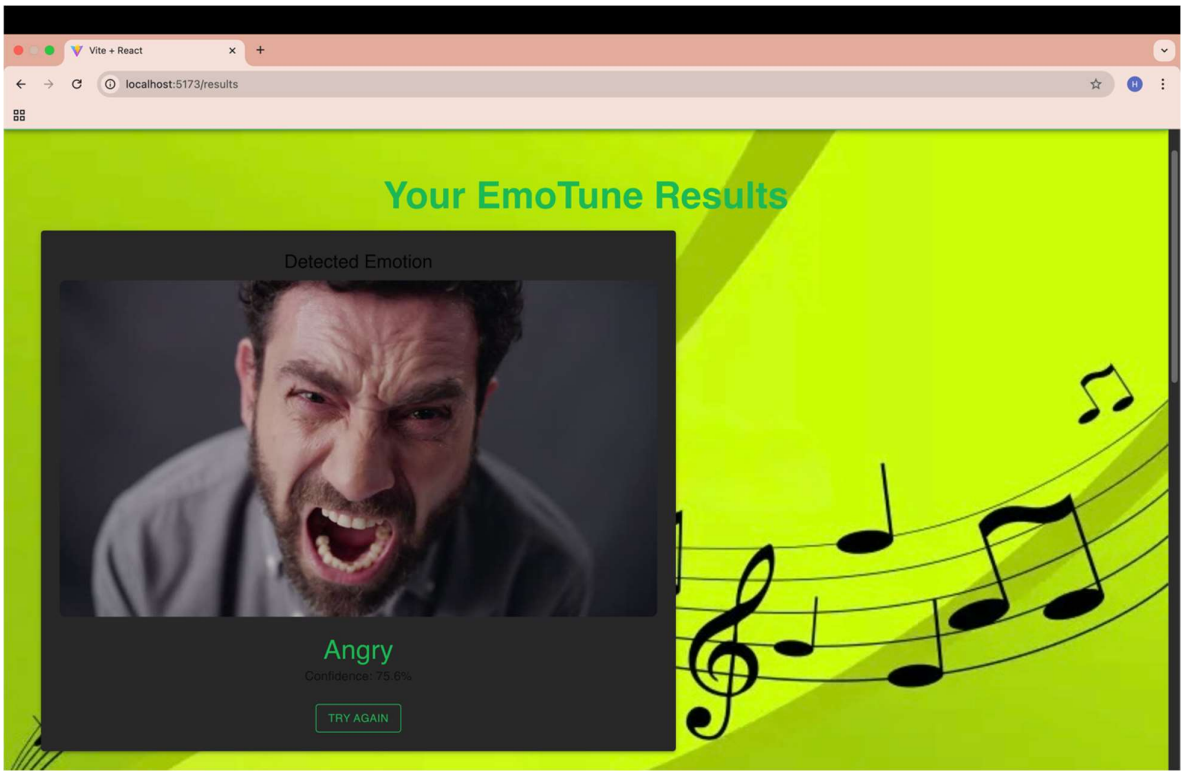
Figure 25 - Implemented UI

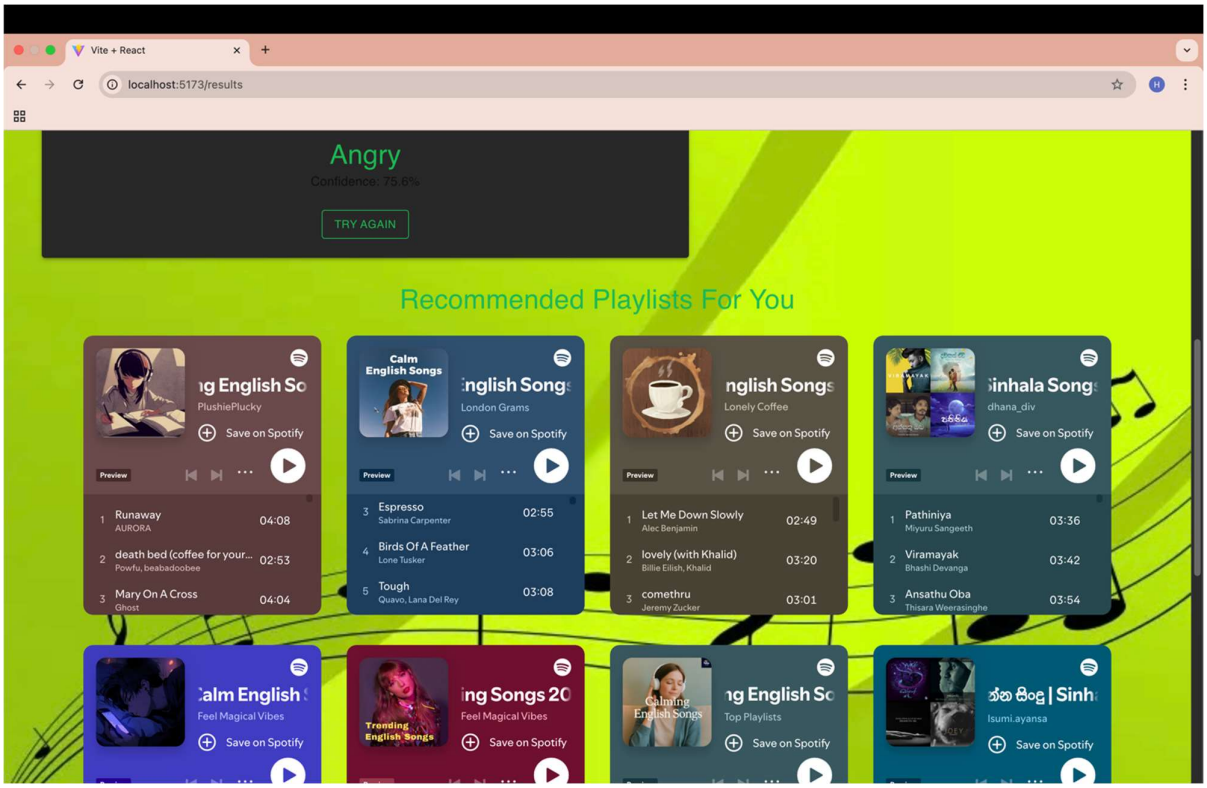


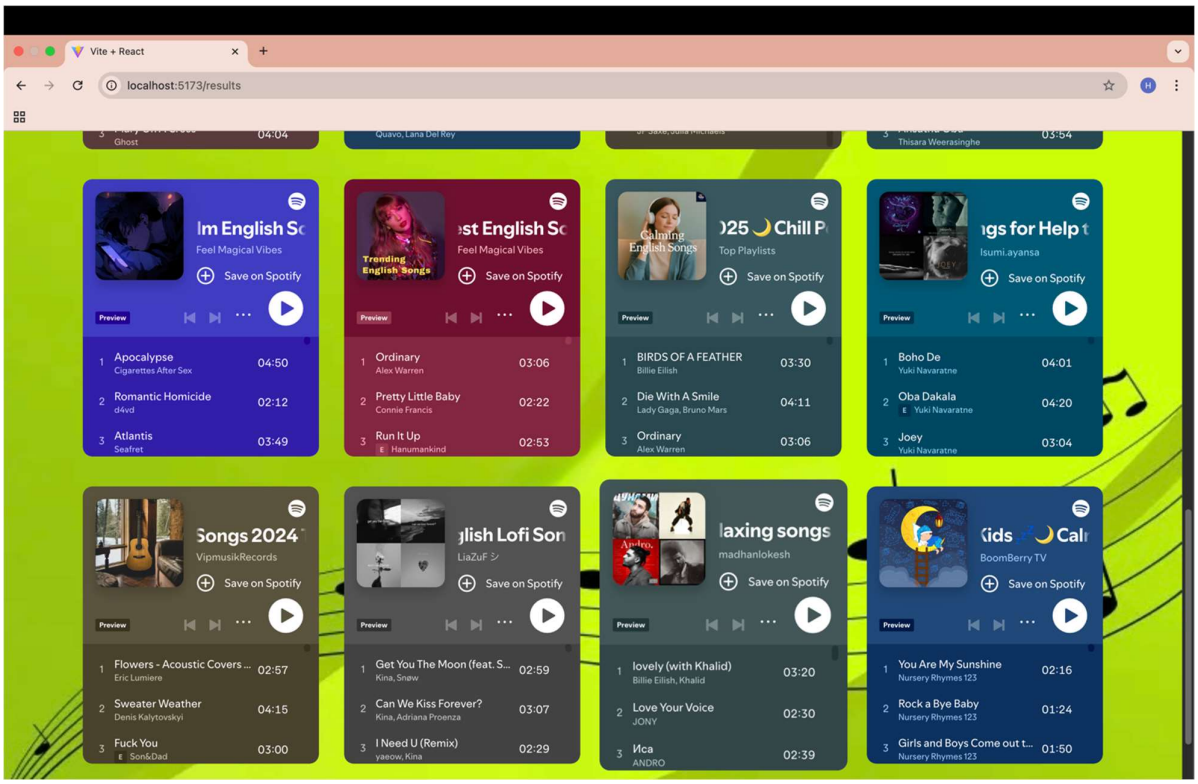



Vite + React
localhost:5173/results
Sad
Confidence: 43.4%
TRY AGAIN
Recommended Playlists For You
sad songs to
Purple_ocean7
Save on Spotify
Preview
1 In The Stars Benson Boone 03:36
2 Someone You Loved Lewis Capaldi 03:02
3 Let Me Down Slowly Alec Benjamin 02:49
මනොපහර
Praveenpeiris
Save on Spotify
1 Sobana Ridma Weerawardena, Senanga ... 04:19
2 Mulawe Mihiran 03:55
3 Viramayak Bhashi Devanga 03:42
glish Songs
✨ SNEHAA ✨
Save on Spotify
1 Lonely (with benny blanco) E Justin Bieber, benny blanco 02:29
2 Someone You Loved Lewis Capaldi 03:02
3 golden hour JVKE 03:29
js ever - Engl
Immortal ak
Save on Spotify
1 Heat Waves Glass Animals 03:58
2 Hope E XXXTENTACION 01:50
3 Infinity Jaymes Young 03:57
Sad Songs
music._vibe4you
Save on Spotify
sh Romantic
Romantic
Save on Spotify
Songs [SAD]
Jung Yoora YN_JK
Save on Spotify
ongs English
StereoAux
Save on Spotify
Vite + React
localhost:5173/results
3 Let Me Down Slowly Alec Benjamin 02:49
3 Viramayak Bhashi Devanga 03:42
Duncan Laurence 03:05
English Sad
music._vibe4you
Save on Spotify
1 Love Is Gone - Acoustic SLANDER, Dylan Matthew 02:56
2 Dancing With Your Ghost Sasha Alex Sloan 03:17
3 Moral of the Story E Ashe 03:21
025 | Best
Romantic
Save on Spotify
1 Sapphire Ed Sheeran 02:59
2 Love Me Like You Do - Fro... Ellie Goulding 04:12
3 Shape of You Ed Sheeran 03:53
Sinhala Moo
Jung Yoora YN_JK
Save on Spotify
1 Dekopul Kandulin Thema La Signore 02:53
2 Numba Ha DILU Beats 02:43
3 Mandaaram Kathawe Anushka Udana, Nisal Gamage, A... 03:25
eartbreak Hi
StereoAux
Save on Spotify
1 The Night We Met Lord Huron 03:28
2 Sapphire Ed Sheeran 02:59
3 Love Is Gone - Acoustic SLANDER, Dylan Matthew 02:56
රකට සිංදු #
Mowin Mac
Save on Spotify
1 Handawaka Yasas Medagedara, Raveen Tharu... 03:40
2 Adarei Uthuranna Gangadara 01:06
3 Sanda Numba Awidin Uvindu Ayshcharya, DILU Beats 03:56
h sad songs
Atulya Sharma
Save on Spotify
1 Past Lives - slowed + reve... sapientdream, Slushii 02:50
2 Can We Kiss Forever? Kina, Adriana Proenza 03:07
3 Into Your Arms x Alone Kausak 04:02
n songs - E
GaaraTheGod.Exe
Save on Spotify
1 Can We Kiss Forever? Kina, Adriana Proenza 03:07
2 Let Me Down Slowly Alec Benjamin 02:49
3 lovely (with Khalid) Billie Eilish, Khalid 03:20
alm English
Feel Magical Vibes
Save on Spotify
1 Apocalypse Cigarettes After Sex 04:50
2 Romantic Homicide d4vd 02:12
3 Atlantis Seafret 03:49


Vite + React
localhost:5173/detect
EmoTune
HOME
DETECT EMOTION
Capture Your Vibe
Use your webcam or upload a photo to analyze your emotion.
WEBCAM
UPLOAD
CHOOSE FILE
ANALYZE EMOTION


Vite + React
localhost:5173/results
Your EmoTune Results
Detected Emotion
Angry
TRY AGAIN


Vite + React
localhost:5173/results
Angry
Confidence: 75.6%
TRY AGAIN
Recommended Playlists For You
PlushiePlucky
Save on Spotify
1 Runaway AURORA 04:08
2 death bed (coffee for your... Powfu, beabadoobee 02:53
3 Mary On A Cross Ghost 04:04
London Grams
3 Espresso Sabrina Carpenter 02:55
4 Birds Of A Feather Lone Tusker 03:06
5 Tough Quavo, Lana Del Rey 03:08
Lonely Coffee
1 Let Me Down Slowly Alec Benjamin 02:49
2 lovely (with Khalid) Billie Eilish, Khalid 03:20
3 comethru Jeremy Zucker 03:01
dhana_div
1 Pathiniya Miyuru Sangeeth 03:36
2 Viramayak Bhashi Devanga 03:42
3 Ansathu Oba Thisara Weerasinghe 03:54
Feel Magical Vibes
Top Playlists
Isumi.ayansa


Vite + React
localhost:5173/results
Feel Magical Vibes
Save on Spotify
1 Apocalypse Cigarettes After Sex 04:50
2 Romantic Homicide d4vd 02:12
3 Atlantis Seafret 03:49
1 Ordinary Alex Warren 03:06
2 Pretty Little Baby Connie Francis 02:22
3 Run It Up Hanumankind 02:53
Top Playlists
1 BIRDS OF A FEATHER Billie Eilish 03:30
2 Die With A Smile Lady Gaga, Bruno Mars 04:11
3 Ordinary Alex Warren 03:06
Isumi.ayansa
1 Boho De Yuki Navaratne 04:01
2 Oba Dakala Yuki Navaratne 04:20
3 Joey Yuki Navaratne 03:04
VipmusikRecords
1 Flowers - Acoustic Covers ... Eric Lumiere 02:57
2 Sweater Weather Denis Kalytovskyi 04:15
3 Fuck You Son&Dad 03:00
LiaZuF
1 Get You The Moon (feat. S... Kina, Snøw 02:59
2 Can We Kiss Forever? Kina, Adriana Proenza 03:07
3 I Need U (Remix) yaeow, Kina 02:29
madhanlokesh
1 lovely (with Khalid) Billie Eilish, Khalid 03:20
2 Love Your Voice JONY 02:30
3 Иса ANDRO 02:39
BoomBerry TV
1 You Are My Sunshine Nursery Rhymes 123 02:16
2 Rock a Bye Baby Nursery Rhymes 123 01:24
3 Girls and Boys Come out t... Nursery Rhymes 123 01:50

Appendix F ConvNext V2 Accuracy Graph

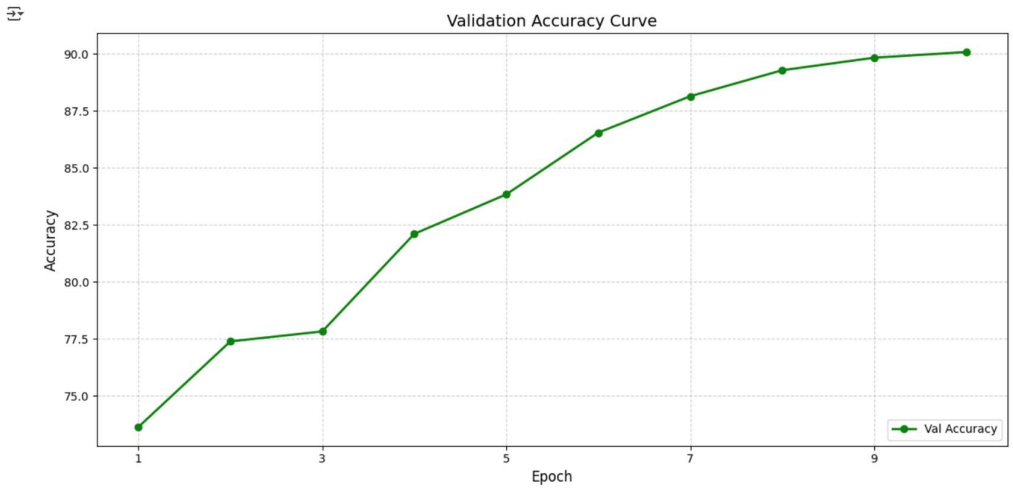


Figure 26 - Model Testing Accuracy

Appendix G Model Design

```
[ ] import timm
    import torch
    from torch import nn, optim
    from torch.optim.lr_scheduler import CosineAnnealingLR
    from timm.loss import LabelSmoothingCrossEntropy
    from torch.cuda.amp import autocast, GradScaler
    from tqdm import tqdm
    import matplotlib.pyplot as plt
    import numpy as np

    num_epochs = 10
    learning_rate = 1e-4
    weight_decay = 0.1
    batch_size = 32

    device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
    print(f"Using device: {device}")

    num_classes = len(train_dataset.classes)
    model = timm.create_model(
        "convnextv2_tiny.fcmae_ft_in1k",
        pretrained=True,
        num_classes=num_classes
    )
    model.to(device)

    criterion = LabelSmoothingCrossEntropy(smoothing=0.15)
    optimizer = optim.AdamW(model.parameters(), lr=learning_rate, weight_decay=weight_decay)
    scheduler = CosineAnnealingLR(optimizer, T_max=num_epochs, eta_min=1e-6)
    scaler = GradScaler()

    train_loss_values = []
    val_loss_values = []
    val_acc_values = []

    for epoch in range(num_epochs):
        model.train()
        running_train_loss = 0.0
        correct = 0
        total = 0

        pbar = tqdm(train_loader, desc=f"[Epoch {epoch+1}/{num_epochs}] Training")
        for images, labels in pbar:
            images, labels = images.to(device), labels.to(device)

            optimizer.zero_grad()
            with autocast():
                outputs = model(images)
                loss = criterion(outputs, labels)

            scaler.scale(loss).backward()
            scaler.step(optimizer)
            scaler.update()

            running_train_loss += loss.item() * images.size(0)
            _, predicted = outputs.max(1)
            total += labels.size(0)
            correct += predicted.eq(labels).sum().item()

            pbar.set_postfix(loss=loss.item(), acc=100. * correct / total)
```

```
    scheduler.step()
    train_acc = 100. * correct / total
    epoch_train_loss = running_train_loss / total
    train_loss_values.append(epoch_train_loss)

    print(f" Epoch {epoch+1}: Train Loss = {epoch_train_loss:.4f}, Accuracy = {train_acc:.2f}%")

    model.eval()
    running_val_loss = 0.0
    correct = 0
    total = 0

    with torch.no_grad():
        for images, labels in val_loader:
            images, labels = images.to(device), labels.to(device)
            with autocast():
                outputs = model(images)
                loss = criterion(outputs, labels)

            running_val_loss += loss.item() * images.size(0)
            _, predicted = outputs.max(1)
            total += labels.size(0)
            correct += predicted.eq(labels).sum().item()

    val_acc = 100. * correct / total
    epoch_val_loss = running_val_loss / total
    val_loss_values.append(epoch_val_loss)
    val_acc_values.append(val_acc)

    print(f" Validation: Loss = {epoch_val_loss:.4f}, Accuracy = {val_acc:.2f}%\n")
```

Figure 27 - ConvNext V2 Summary

```
Using device: cuda
model.safetensors:   0%|          | 0.00/115M [00:00<?, ?B/s]
[Epoch 1/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=62.4, loss=1.15] Epoch 1: Train Loss = 1.1499, Accuracy = 62.36%
 Validation: Loss = 0.9840, Accuracy = 73.64%
[Epoch 2/10] Training: 100%|██████████| 365/365 [13:58<00:00,  2.30s/it, acc=71.9, loss=0.947] Epoch 2: Train Loss = 1.0061, Accuracy = 71.89%
 Validation: Loss = 0.9185, Accuracy = 77.39%
[Epoch 3/10] Training: 100%|██████████| 365/365 [13:58<00:00,  2.30s/it, acc=75.8, loss=0.809] Epoch 3: Train Loss = 0.9456, Accuracy = 75.79%
 Validation: Loss = 0.9090, Accuracy = 77.83%
[Epoch 4/10] Training: 100%|██████████| 365/365 [13:58<00:00,  2.30s/it, acc=79.3, loss=0.915] Epoch 4: Train Loss = 0.8927, Accuracy = 79.32%
 Validation: Loss = 0.8536, Accuracy = 82.11%
[Epoch 5/10] Training: 100%|██████████| 365/365 [13:58<00:00,  2.30s/it, acc=82.4, loss=0.911] Epoch 5: Train Loss = 0.8431, Accuracy = 82.41%
 Validation: Loss = 0.8177, Accuracy = 83.84%
[Epoch 6/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=85.4, loss=0.849] Epoch 6: Train Loss = 0.7954, Accuracy = 85.39%
 Validation: Loss = 0.7860, Accuracy = 86.55%
[Epoch 7/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=88, loss=0.808] Epoch 7: Train Loss = 0.7532, Accuracy = 88.05%
 Validation: Loss = 0.7609, Accuracy = 88.15%
[Epoch 8/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=90, loss=0.655] Epoch 8: Train Loss = 0.7210, Accuracy = 89.98%
 Validation: Loss = 0.7436, Accuracy = 89.29%
[Epoch 9/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=91.3, loss=0.675] Epoch 9: Train Loss = 0.7011, Accuracy = 91.30%
 Validation: Loss = 0.7356, Accuracy = 89.84%
[Epoch 10/10] Training: 100%|██████████| 365/365 [14:00<00:00,  2.30s/it, acc=91.9, loss=0.646] Epoch 10: Train Loss = 0.6908, Accuracy = 91.92%
 Validation: Loss = 0.7305, Accuracy = 90.09%
```

Appendix H Use Case Description

ID	UC:02
Description	This use case describes generating a music playlist tailored to the user's current emotional state.
Participating actors	User
Preconditions	User's emotion has been detected.

Extended use cases	None	
Included use cases	Analyse User Emotions	
Main flow	**Actor**	**System**
	1. Request a playlist.	1. Match detected emotion to suitable genres. 2. Query tracks and build playlist. 3. Display playlist to user.
Alternative flows	None	
Exceptional flows	No tracks available: Display generic playlist.	
Post conditions	User receives an emotion-matched playlist.	

Table 36 - Use Case Description 2

ID	UC:03	
Description	This use case allows users to view their history of previously recommended playlists.	
Participating actors	User	
Preconditions	User has interacted with the system previously.	
Extended use cases	None	
Included use cases	None	
Main flow	**Actor**	**System**
	1. Request history.	1. Retrieve and display past recommendations.
Alternative flows	None	

Exceptional flows	No history available: Show empty state message.
Post conditions	User views past recommended playlists.

Table 37 - Use Case Description 3

ID	UC:04	
Description	This use case describes the process of analyzing the user's captured facial expression to detect their emotional state using a deep learning model.	
Participating actors	User	
Preconditions	User must have captured a valid facial image.	
Extended use cases	None	
Included use cases	Categorizing tracks by genre	
Main flow	**Actor**	**System**
	1. Provide captured image.	1. Preprocess the image for analysis. 2. Run the emotion recognition model. 3. Output detected emotion.
Alternative flows	None	
Exceptional flows	Image corrupted or invalid: Display error message. Model fails to predict: Display fallback message.	
Post conditions	Emotional state detected and passed to downstream personalization modules.	

Table 38 - Use Case Description 4

ID	UC:05	
Description	This use case describes categorizing available tracks into genres based on metadata from the Music API.	
Participating actors	Music API Provider	
Preconditions	Music metadata is accessible via API.	
Extended use cases	None	
Included use cases	None	
Main flow	**Actor**	**System**
	1. Provide track data.	1. Retrieve tracks. 2. Categorize into genres.
Alternative flows	None	
Exceptional flows	Track metadata unavailable: Show error.	
Post conditions	Tracks categorized and available for recommendation.	

Table 39 - Use Case Description 5

ID	UC:06
Description	This use case allows the system administrator to manage user profiles.
Participating actors	System Administrator
Preconditions	Administrator logged in.
Extended use cases	None
Included use cases	None

Main flow	**Actor**	**System**
	1. Select profile to edit/delete.	1. Perform requested action.
Alternative flows	None	
Exceptional flows	Invalid action: Show error message.	
Post conditions	Profiles updated as per request.	

Table 40 - Use Case Description 6

ID	UC:07	
Description	This use case allows the system administrator to monitor logs for performance and security checks.	
Participating actors	System Administrator	
Preconditions	Logs generated by system.	
Extended use cases	None	
Included use cases	None	
Main flow	**Actor**	**System**
	1. Request logs.	1. Display system activity logs.
Alternative flows	None	
Exceptional flows	Logs unavailable: Show error message.	
Post conditions	Logs reviewed and administrator informed.	

Table 41 - Use Case Description 7

ID	UC:08	
Description	This use case allows the deep learning specialist to retrain the emotion recognition model with updated datasets.	
Participating actors	Deep Learning Specialist	
Preconditions	New dataset is available and system resources are ready.	
Extended use cases	None	
Included use cases	None	
Main flow	**Actor**	**System**
	1. Start retraining process.	1. Load new data and train model. 2. Save updated model.
Alternative flows	None	
Exceptional flows	Training fails: Log and show error.	
Post conditions	Updated model deployed successfully.	

Table 42 - Use Case Description 8

ID	UC:09
Description	This optional use case allows users to save the captured facial image for their records.
Participating actors	User
Preconditions	Facial image has already been captured.
Extended use cases	None

Included use cases	None	
Main flow	**Actor**	**System**
	1. Opt to save image.	1. Save image to user profile or file system.
Alternative flows	None	
Exceptional flows	Storage error: Show failure message.	
Post conditions	Image successfully stored.	

Table 43 - Use Case Description 9