TED UNIVERSITY

CMPE 491 / SENG 491 Senior Project

<<BooTunes>>

High Level Design Report

Fall 2024

Team Members:

Mehlika Eroğlu, 58717180232, Software Engineering

Melih Aydın, 28180576276, Computer Engineering

Elif Alptekin, 31376233198, Computer Engineering

Batuhan Dalkılıç, 11491297756, Computer Engineering

Supervisor: Venera Adanova

Jury Members:

Gökçe Nur Yılmaz
Eren Ulu
Firat AKBA

Contents

1. Introduction

1.1. Purpose of the System

The ‘BooTunes’ project is an application that redefines the digital reading experience by integrating artificial intelligence-powered sentiment analysis with dynamic music and visuals. Designed as a cross-platform application, BooTunes enhances engagement by analyzing the emotional tone of book pages and aligning them with mood-appropriate multimedia elements. This creates an immersive, multi-sensory environment that resonates deeply with readers.

BooTunes allows users to upload books or select from a library of previously uploaded books, where each page is analyzed using advanced Natural Language Processing (NLP) techniques. Music and visuals are dynamically curated to reflect emotional shifts, ensuring a personalized and engaging reading experience. For example, tense moments are accompanied by suspenseful music and darker visuals, while joyful passages feature uplifting tunes and brighter imagery.

Beyond enhancing engagement, BooTunes promotes accessibility and inclusivity. Customizable settings for visual effects and music genres ensure the platform caters to diverse preferences. Features like dark mode make it inclusive for varied audiences.

1.2. Design Goals

BooTunes' design focuses on creating a reading experience that will use AI technologies to improve, innovate and personalize the reading experience of its users. The following objectives summarize the basic principles that guide the development of the system:

- To ensure an immersive reading experience based on the seamless integration of natural language processing (NLP) and artificial intelligence-driven sentiment analysis. The system dynamically adapts to emotional shifts in the text, providing users with matching music and visuals that increase engagement and emotional resonance.
- Ensure cross-platform compatibility and consistent user experience across mobile and web platforms. The design emphasizes responsive and intuitive interfaces that cater to diverse device preferences, maintaining functionality and aesthetic appeal.
- Prioritize user privacy and data security by adhering to strict standards, including GDPR compliance. The system encrypts sensitive information and provides users with transparent control over their data.
- Optimize performance to achieve minimal latency in emotion detection, music playback, and visual rendering. This ensures smooth and uninterrupted transitions, enhancing the overall user experience.

- Offer extensive customization options to accommodate diverse user needs. Users can tailor image styles, and music genres to create a highly personalized interaction.
- Foster user engagement and satisfaction through innovative feedback mechanisms. The system learns from user interactions to refine recommendations and improve the alignment of multimedia elements with individual preferences.
- Support scalability and modularity in design, allowing for future enhancements and updates without disrupting existing functionalities. This includes integrating new AI models and expanding the library of multimedia resources.

By meeting these goals, BooTunes aims to redefine how users interact with literature, bridging the gap between traditional storytelling and modern technology while delivering a truly enriched reading experience.

1.3. Definitions, Acronyms, and Abbreviations

To ensure clarity and consistency throughout this report, the following definitions, acronyms, and abbreviations are provided:

- **AI (Artificial Intelligence)**: A branch of computer science focused on creating systems capable of performing tasks that typically require human intelligence, such as reasoning, learning, and decision-making. In BooTunes, AI drives sentiment analysis and multimedia integration.
- **NLP (Natural Language Processing)**: A subfield of AI that enables machines to understand, interpret, and respond to human language. BooTunes employs NLP to analyze the emotional tone of book pages in real time.
- **API (Application Programming Interface)**: A set of protocols and tools that allow different software applications to communicate with one another. BooTunes integrates APIs such as Spotify for music recommendations and services as cloud for data processing.
- **GDPR (General Data Protection Regulation)**: A regulatory framework established by the European Union to ensure the protection of personal data. BooTunes adheres to GDPR standards to safeguard user information and provide transparent data usage policies.
- **UI/UX (User Interface/User Experience)**: Refers to the design and functionality of the application interface and the overall user interaction experience. BooTunes prioritizes intuitive UI/UX design to ensure seamless navigation and accessibility for all users.
- **Sentiment Analysis**: The process of identifying and categorizing the emotional tone within a piece of text. This is a core feature of BooTunes, enabling the system to adapt multimedia elements to match the narrative's mood.

- **Cross-Platform**: Compatibility across various operating systems, including iOS, Android, Windows, and macOS, ensuring that BooTunes delivers a consistent experience regardless of the user's device.

1.4. Overview

This report outlines the comprehensive system design for BooTunes, detailing the decomposition of subsystems, mapping of hardware and software components, and the strategies employed for managing data and ensuring system security. The document explores creative and technical design principles that underpin BooTunes' innovative approach to integrating AI-driven sentiment analysis, dynamic multimedia, and cross-platform functionality.

The report provides an examination of how each subsystem—including emotion analysis, music management, and visual generation—contributes to the seamless user experience. It also explains how hardware resources, such as cloud infrastructure and local devices, interact with the software architecture to deliver low-latency, high-quality performance. Global control flow and boundary conditions considerations ensure that BooTunes maintains its performance and functionality under changing network or user conditions.

Additionally, this document highlights the application's compliance with data privacy regulations like GDPR, emphasizing user-centric design principles such as transparency and control over personal data. Accessibility and inclusivity are key aspects of the system design, ensuring that BooTunes caters to diverse audiences with some features and customizable settings.

Through this report, one can learn about the software architecture of BooTunes, which not only offers a new and functional solution but also sets a new standard for interactive digital reading experiences.

2. Current Software Architecture

Currently there is no software architecture for our project .The project is in its planning phase, with models, data sources, and the project structure defined. Implementation is set to begin soon, paving the way for this innovative application.

3. Proposed Software Architecture

3.1. Overview

The proposed software architecture is to use a modular and scalable design that divides functions into different subsystems to ensure seamless integration and in-application flexibility. Real-time sentiment analysis is powered by advanced NLP models that detect emotional tones in text and make dynamic adjustments to music and visuals.

The Sentiment Analysis Model interprets in-text sentiment sentiment, allowing the Music Manager Model to select matching tracks and the Visual Manager Model to create corresponding visuals, creating an engaging, mood-appropriate reading environment for users. In architecture, one of our goals is to provide a seamless and continuous platform using secure tools to ensure efficient processing, low latency and scalability.

3.2. Subsystem Decomposition

- **Emotion Analysis Subsystem**: This subsystem is the core of the BooTunes architecture, responsible for processing textual content to identify emotions using advanced NLP models. It outputs metadata that guides the Music and Visual Management subsystems. This subsystem supports real-time sentiment detection, ensuring low latency and high accuracy, and includes customization options for sensitivity adjustments to cater to user preferences.
- **Music Management Subsystem**: The Music Manager subsystem selects and streams music tracks that align with the emotional tone defined by the Sentiment Analysis Subsystem. It integrates with external APIs with extensive data to access a large library of categorized music.
- **Visual Management Subsystem**: This subsystem generates and displays visuals that reflect the emotional tone of the text, enhancing the reader's immersion. It utilizes AI-based image generation tools and preloaded thematic visuals to ensure high-quality, mood-appropriate imagery. The subsystem supports offline caching, adjustable intensity settings, and compatibility with varying screen sizes to maintain a consistent visual experience across platforms.

- **User Interface Subsystem**: The user interface subsystem provides navigation experience to users. It will be created with a design appropriate for a reader's anticipated expectations, ensuring consistency across devices and platforms, including mobile and desktop. Features include personalized dashboards, accessibility options (e.g. dark mode), and real-time feedback integration for user interaction. It prioritizes ease of use and visual clarity to enhance the overall experience.
- **Backend Subsystem**: The backend subsystem is the foundation of the BooTunes system, managing API integrations, data storage, and user authentication. It must support secure communication protocols to handle high user loads, the cloud infrastructure we intend to implement, and compliance with privacy standards such as GDPR. This subsystem is also responsible for processing and storing user preferences, reading history, and feedback; thus providing a personalized and secure experience.
- **Recommendation Subsystem**: (Additional subsystem that can be added if the main objectives of the project are met) Suggests books and music tracks based on user preferences, reading history, and past interactions. This subsystem leverages AI algorithms to refine its suggestions over time, enhancing personalization and user engagement.

By dividing the system into these well-defined subsystems, BooTunes ensures flexibility, scalability, and the ability to provide a cohesive and immersive reading experience that adapts dynamically to user preferences and emotional tones.

3.3. Hardware/Software Mapping

- **Frontend**: The frontend will be built using React.JS for web applications and React Native for mobile platforms. These technologies aim to provide a consistent and responsive user interface across devices. With React.JS and React Native, the platform provides a user-friendly application by supporting modern UI/UX features such as navigation, modular component-based design, and dynamic rendering.
- **Backend**: The backend leverages Python frameworks such as Django or Flask to handle server-side logic, API integrations, and user authentication. These frameworks provide robust tools for managing complex application workflows, ensuring scalability and reliability. The backend also facilitates secure communication between subsystems, managing requests for emotion analysis, music selection, and visual rendering.
- **Database**: MongoDB is used as the database solution for storing user data, preferences, reading history, and uploaded content. Its document-based structure allows for efficient querying and flexible schema design, which is essential for handling the diverse and dynamic data associated with BooTunes' personalized features. MongoDB also supports scalability, ensuring the database can accommodate growing user activity.

- **Deployment**: The application will initially be developed and utilized locally during the development phase to ensure rapid prototyping and testing. In the later stages, the system will be migrated to scalable cloud platforms like AWS, Azure, or Google Cloud to provide high reliability, scalability, and uptime. These cloud platforms enable features such as load balancing, auto-scaling, and secure storage of user data. Deployment will be optimized to ensure low latency for real-time emotion analysis and multimedia adjustments, providing users with a smooth and responsive experience.

The corresponding UML Class Diagram:

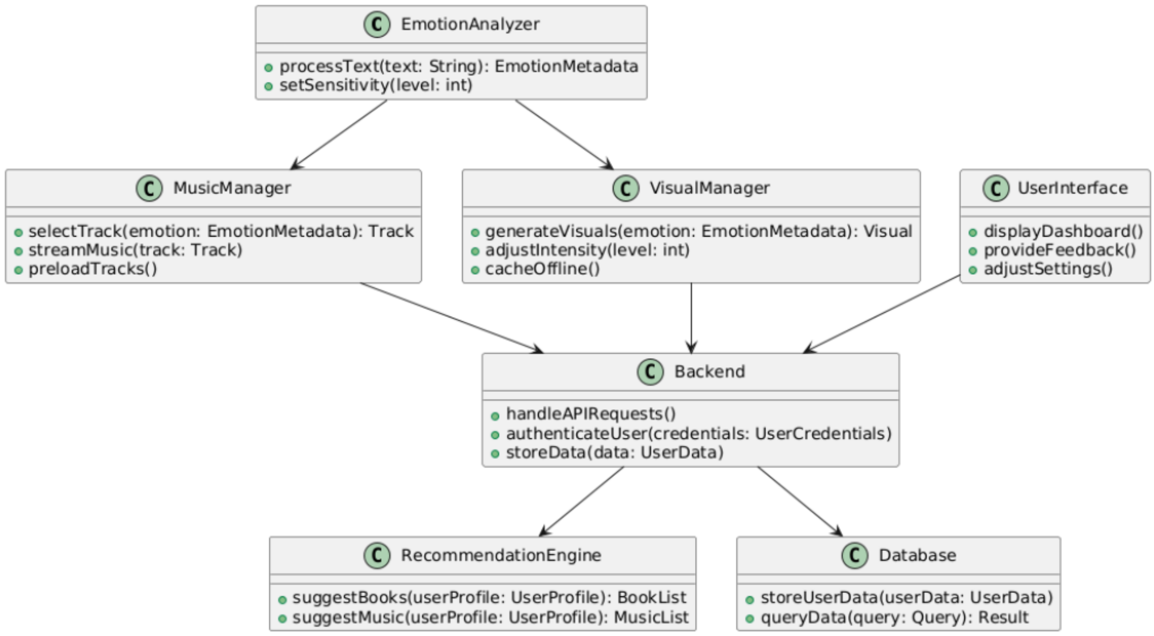


Diagram 1: UML Class Diagram [1]

Class Diagram

- _EmotionAnalyzer:_ Processes text to detect emotions and provides metadata for music and visuals.
- _MusicManager:_ Selects, streams, and preloads music tracks based on detected emotions.
- _VisualManager:_ Generates and caches visuals corresponding to emotional tones.
- _UserInterface:_ Offers an intuitive interface for settings, feedback, and navigation.
- _Backend:_ Handles API requests, authentication, and data storage securely.

- *RecommendationEngine:* (Additional) Suggests books and music tailored to user preferences.
- *Database:* Manages user data and preferences efficiently.

Use Case Diagram

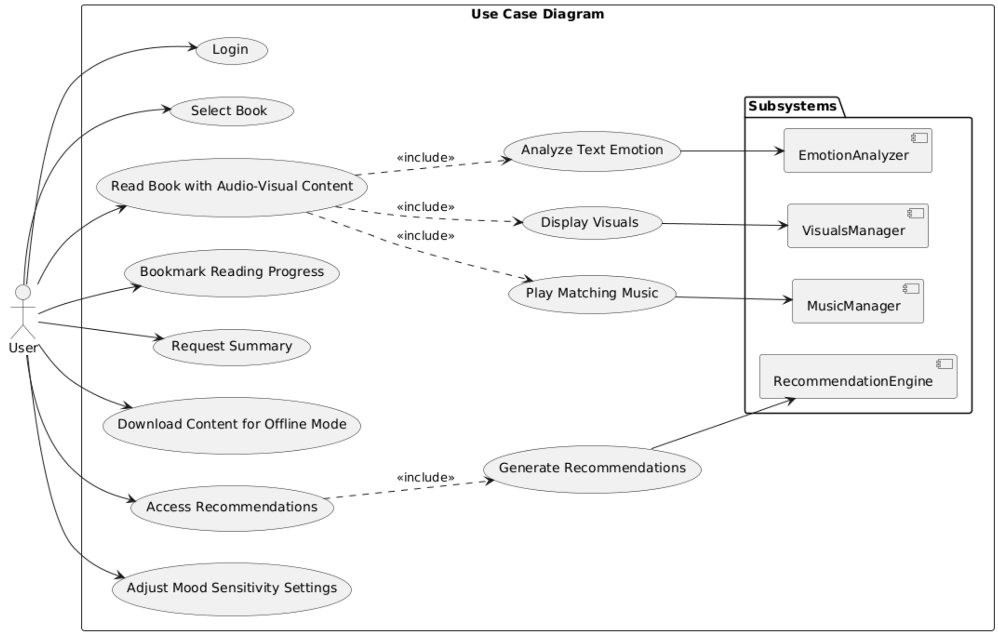


Diagram 2: Use Case Diagram [2]

This use case diagram represents the key functionalities of the BooTunes system and the interaction between the user and its subsystems:

1. *Login*: The user logs into the application.
2. *Select Book:* The user selects a book to read.
3. *Read Book with Audio-Visual Content*: The system analyzes the text emotions, plays matching music, and displays visuals using the EmotionAnalyzer, MusicManager, and VisualManager subsystems.
4. *Bookmark Reading Progress*: The user can save their reading position.
5. *Request Summary*: The user requests a summary of the book or recent progress.
6. *Download Content for Offline Mode:* Users can download books, music, and visuals for offline use.
7. *Access Recommendations*: The system generates personalized book and music recommendations using the RecommendationEngine.

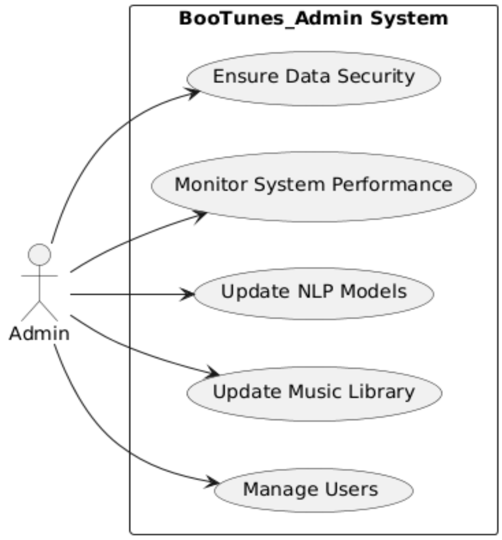


Diagram 3: Key Administrative Functions Use Case Diagram [3]

This diagram illustrates the key administrative functions within the BooTunes_Admin system:

1. ***Ensure Data Security*:** The admin ensures user data is protected and complies with privacy standards.
2. ***Monitor System Performance*:** The admin tracks and analyzes the system's performance to maintain reliability.
3. *Update NLP Models:* The admin updates and enhances the sentiment analysis models as needed.
4. ***Update Music Library*:** The admin manages and expands the music library for better recommendations.
5. *Manage Users:* The admin oversees user accounts, including resolving issues and maintaining system integrity.

Sequence Diagram

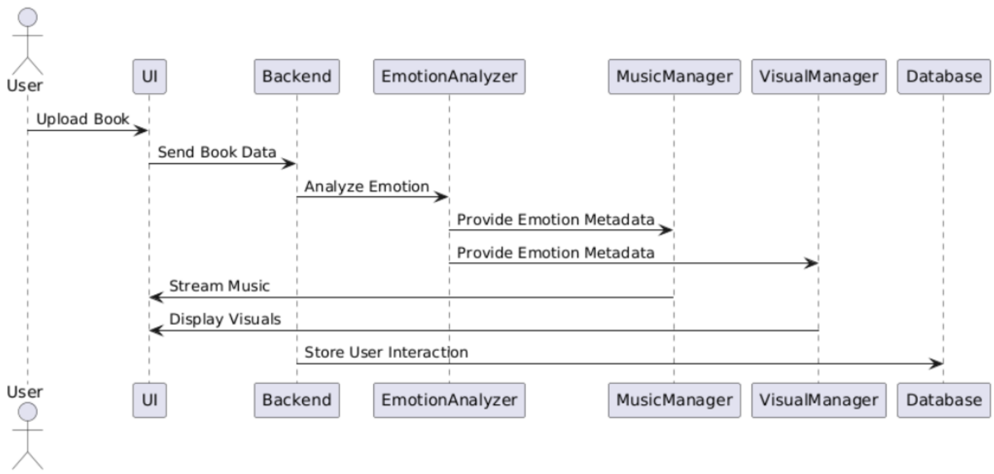


Diagram 4: Sequence Diagram [4]

This sequence diagram illustrates the BooTunes system's interaction flow:

1. **User Upload**: The user uploads a book via the **UI**.
2. **Data Processing**: The **UI** sends the book data to the **Backend**.
3. **Emotion Analysis**: The **Backend** forwards the content to the **EmotionAnalyzer**, which analyzes the text and returns emotion metadata.
4. **Music Streaming**: The **MusicManager** uses the emotion metadata to select and stream matching music.
5. **Visual Display**: The **VisualManager** generates visuals based on the emotion metadata and sends them to the **UI**.
6. **Data Storage**: User interactions and preferences are sent to the **Backend** and stored in the **Database** for personalization.

This process enables collaboration between subsystems, providing a personalized and user-friendly reading experience.

Deployment Diagram

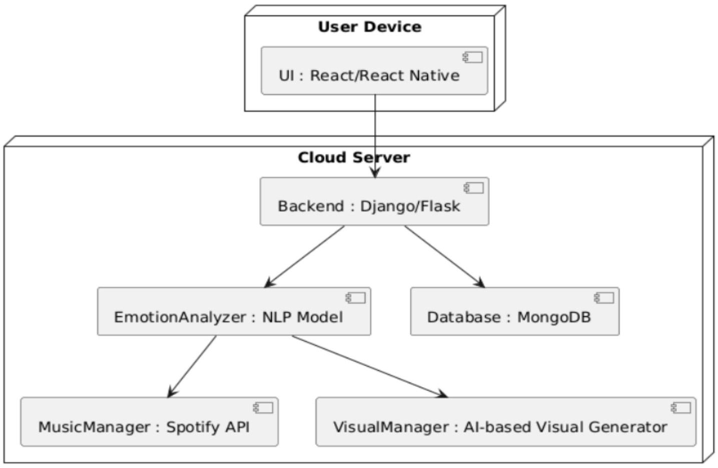


Diagram 5: Deployment Diagram [5]

The deployment diagram shows the BooTunes architecture, divided into the user device and cloud server components: (As we have discussed before, the cloud server will be applied to app in the later phases of development).

1. **User Device**:
   - <u>*UI (React/React Native):*</u> Provides a responsive interface for web and mobile platforms, allowing users to interact with features like book navigation, music, and visuals.
2. **Cloud Server**:

- *Backend (Django/Flask):* Manages communication between the UI and subsystems, handling requests, user authentication, and API integrations.
- *EmotionAnalyzer (NLP Model):* Analyzes text to detect emotions and generates metadata for music and visuals.
- *Database (MongoDB):* Stores user data, preferences, reading history, and system metadata for efficient retrieval.
- *MusicManager (Spotify API):* Selects and streams music aligned with detected emotions using the Spotify API.
- *VisualManager (AI-based Visual Generator)*: Creates mood-matching visuals based on emotional analysis using AI tools.

3.4. Persistent Data Management

In the BooTunes project, managing data effectively is critical to ensure a smooth and secure user experience. The system will design to handle user information, book content, and multimedia elements while keeping data privacy and system performance as top priorities. The architecture leverages MongoDB as the primary database solution, complemented by secure APIs and cloud-based deployment for scalability and high availability. Below are the critical aspects of the persistent data management strategy:

Data Categories:

User Data: Includes user credentials, preferences, reading history, and feedback. Sensitive information is encrypted and managed in compliance with GDPR and other global data privacy standards.

Book Content: Stores uploaded PDFs and preloaded books, along with metadata such as titles, authors, and mood-based tags.

Multimedia Data: Metadata for emotion-aligned music tracks and visuals, including preloaded content for offline use.

Logs and Analytics: Maintains system logs for emotion analysis and multimedia recommendation algorithms, supporting continuous learning and model updates.

Database Structure

- The system uses MongoDB, which is flexible and suitable for storing different types of data in a way that is quick to access.
- Collections are organized to make searching for user preferences or book-related data efficient.

Customization and Personalization

- User-specific settings such as music preferences and visuals are stored and dynamically retrieved to tailor the experience.
- The recommendation engine uses stored analytics to suggest books and music aligned with user interests and emotional patterns.

Data Synchronization

- Real-time synchronization of user preferences and reading progress across devices ensures a multi-platform experience.
- Data preloading for offline mode supports uninterrupted user interactions.

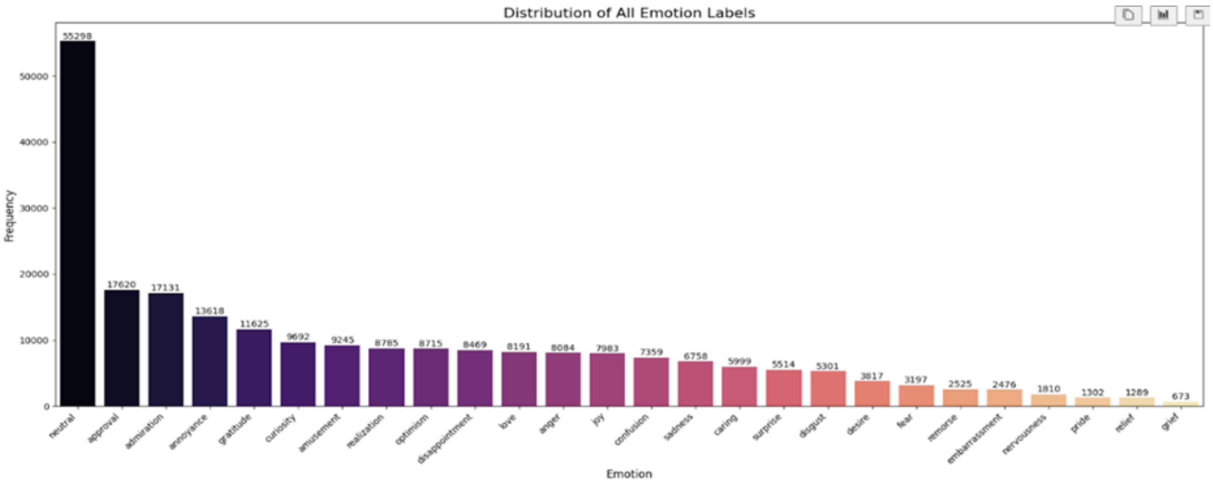
3.4.1. GoEmotions Dataset Analysis

One of the datasets we will use for our project is the GoEmotions dataset, which is a comprehensive resource on emotion detection. This dataset provides a rich set of Reddit comments annotated with 27 different emotion categories. To ensure effective integration into our project, we analyzed the structure, distribution and features of the dataset in detail and focused on its connection with our goals. this dataset contains approximately 58,000 annotated examples associated with multiple emotion labels. Each comment can be assigned one or more sentiment labels, making it a multi-label dataset.

3.4.2. Key Findings

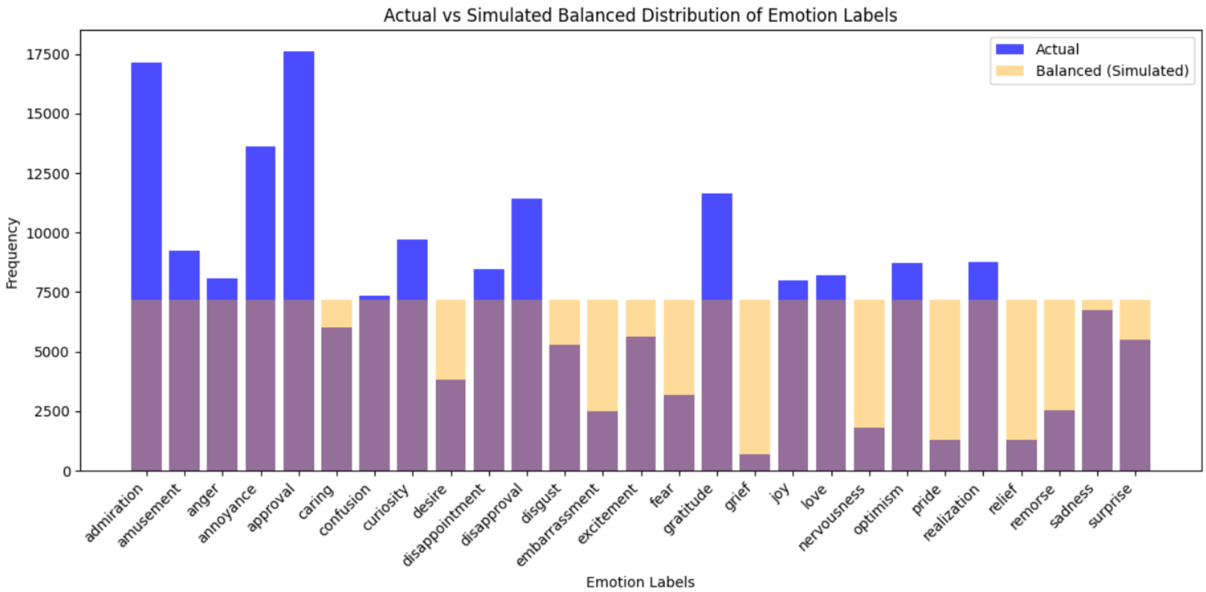
1. Data Composition

- *Textual Data:* Comments vary widely in length; Most comments are between 50 and 150 characters. Frequently used words include colloquial terms like "name," "love," and "people," reflecting hot topics in Reddit discussions.
- *Emotional Labels:* From what we observed, the labels contain a mix of positive, negative, and neutral emotions, such as admiration, joy, anger, and sadness. In particular, we observed that neutral comments made up a large portion of the dataset. There is also a serious class imbalance, with emotions such as admiration and gratitude being more common than labels such as pride and awareness.



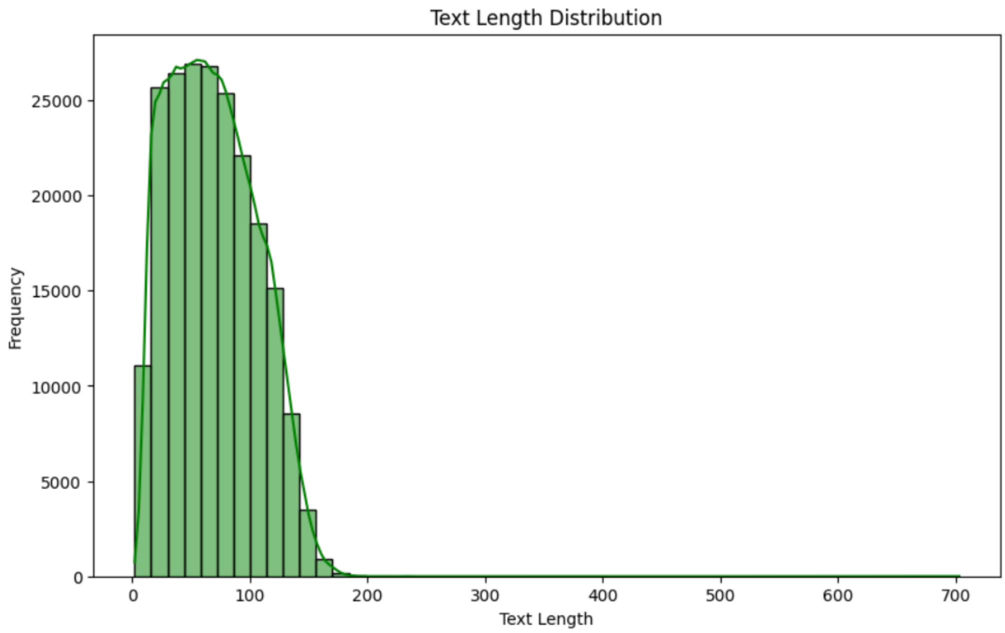
Graph 1: Distribution of All Emotion Labels Graph [6]

- To visualize the dataset's imbalance, we utilized two distinct graphs. Graph 1, which is the first graph, presented the raw distribution of all emotion labels and revealed their real frequencies. This visualization revealed clear disparity between frequent and rare tags, providing an unaltered view of the characteristics of the dataset. Graph 2, which is the second graph, resulted in a simulated balanced distribution of actual tag frequencies, where all tags were assigned the same frequency and calculated as the average frequency across all tags. This hypothesized scenario highlighted the gap between reality and a balanced dataset, highlighting how impactful the level of imbalance is within the project. However, it is very important to note that the simulated balanced bars are purely hypothetical and serve as a diagnostic tool rather than a reflection of actual data.

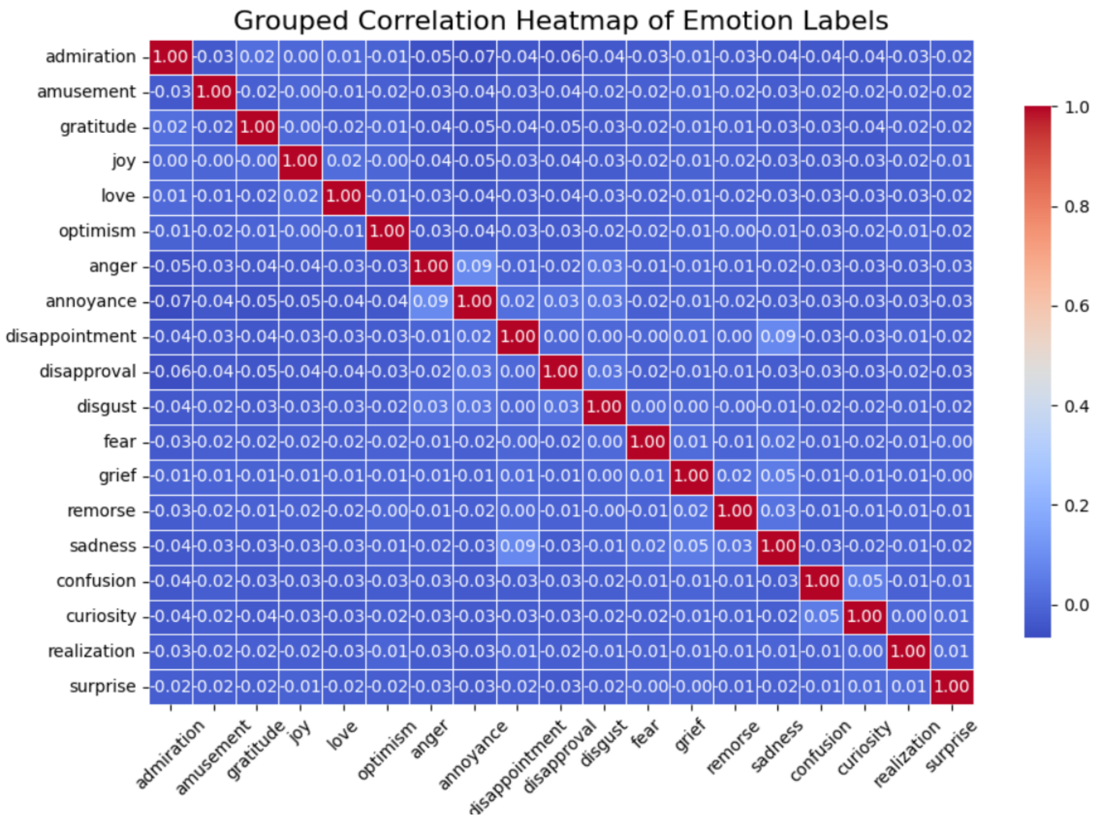


Graph 2: Actual vs Simulated Balanced Distribution of Emotion Labels Graph [7]

- Another aspect of the dataset analysis involved examining the distribution of text lengths. A histogram revealed that most comments fall between 50 and 150 characters, with a right-skewed distribution indicating fewer longer comments. This variability in text length has implications for preprocessing, as comments may require normalization through truncation or padding to ensure consistency when fed into models. Additionally, longer comments often convey richer emotional context, which could be beneficial for certain model architectures, such as transformer-based models.



Graph 3: Text Length Distribution Graph [8]



Graph 4: Grouped Correlation Heatmap of Emotion Labels Graph [9]

- In the chart above, graph 4, you can see the correlation heat map grouped to explore the relationships between Emotion labels. This visualization showed us that there were weak correlations between most labels; this reflected the diversity and independence of emotions. However, some positive correlations, such as between joy and fun, suggested opportunities for hierarchical classification or grouped label predictions. These insights strengthened the dataset's potential for subtle emotion detection, while also highlighting the challenges associated with overlapping and independent emotional categories.

From these insights, we understand better how addressing the imbalance in the dataset is essential for creating effective models. To tackle this issue, techniques like oversampling minority classes such as SMOTE and using class weights during training can help ensure fair representation of all labels.

- The GoEmotions dataset provides a valuable foundation for developing an emotion detection model tailored to our project needs. By integrating insights from this analysis into further preprocessing strategies, model selection, and evaluation metrics, we aim to create a system capable of accurately identifying emotions. These findings will guide our efforts to address class imbalance, handle text variability, and optimize the model for real-world application, ensuring that the dataset's potential is fully realized in our project.
- The BooTunes project also integrates advanced emotion analysis to redefine the digital reading experience. The BooTunes project also integrates advanced sentiment analysis to redefine the digital reading experience. To verify the performance and versatility of the Sentiment Analyzer, we selected two literary classics, Leo Tolstoy's Anna Karenina and Hermann Hesse's Siddhartha, for comprehensive sentiment analysis. We chose these two books because of their completely opposite themes and different emotional trajectories. Anna Karenina is a deeply emotional novel, characterized by intense personal struggles, romance, and tragedy. In contrast, Siddhartha offers a more philosophical and introspective narrative, emphasizing spiritual growth and inner peace. By analyzing these works, we aimed to evaluate the Emotion Analyzer's capability to capture and differentiate complex emotional dynamics across diverse literary styles.
  This analysis not only tests the accuracy and adaptability of the Emotion Analyzer but also highlights its potential to enrich the user experience by aligning music and visuals with the narrative's emotional shifts. The results of this study provide insights into how the system processes varying emotional tones, laying the groundwork for further refinements and enhancements.

3.4.2. Methodology of Analysis

The analysis began by preprocessing the text of *Anna Karenina* and *Siddhartha*. Each book was converted from PDF format into a structured text file, with the content segmented into individual pages. This segmentation mirrors the system's real-time processing of book pages during a typical reading session. The extracted text was then cleaned to remove extraneous characters or formatting artifacts, ensuring the integrity of the input data.

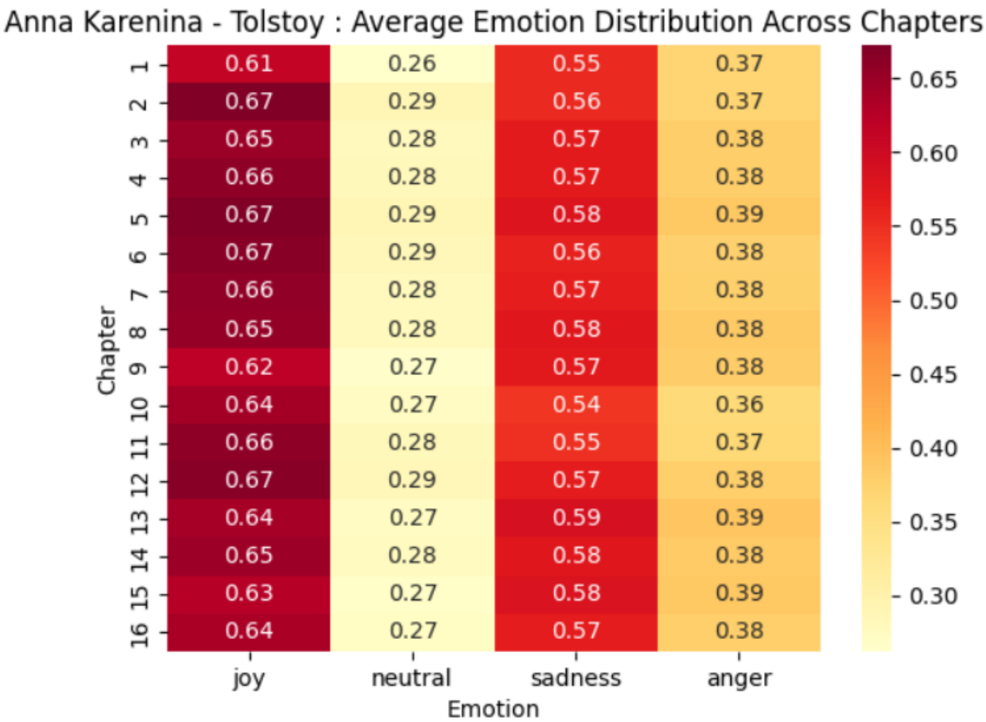
To analyze the emotional content, we utilized a pre-trained natural language processing model from the Hugging Face Transformers library. A specific model was chosen for its proven accuracy in multi-class emotion detection and its suitability for processing large volumes of text efficiently.

3.4.3. Emotion Classification

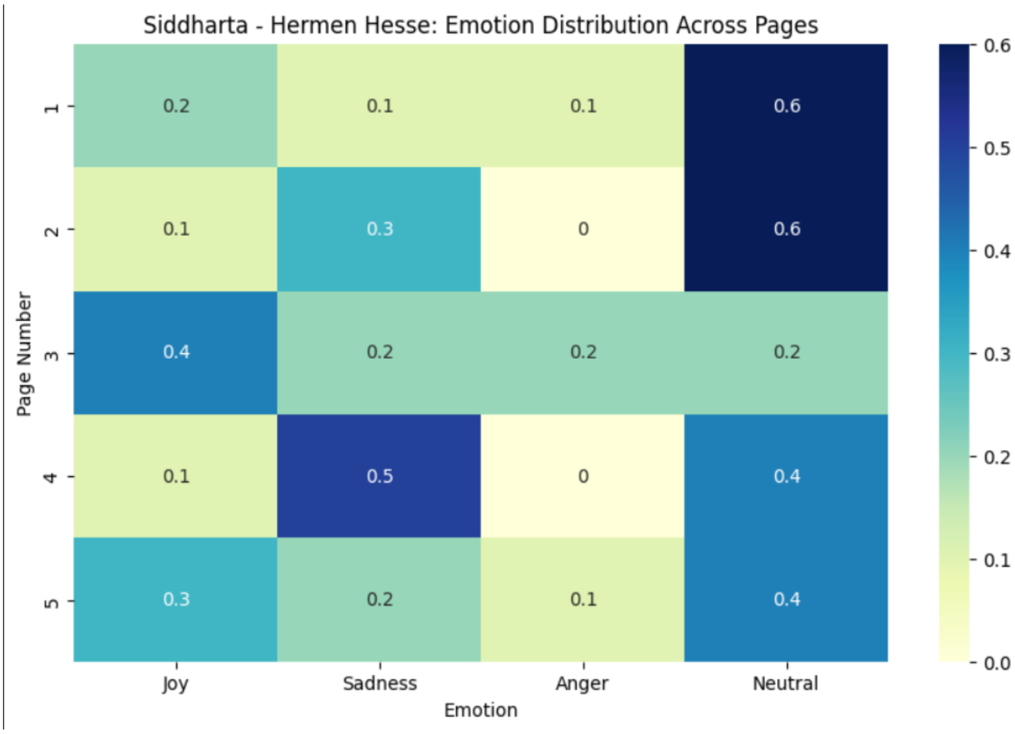
Each page of the two books was processed through the Emotion Analyzer. The pipeline operated as follows:

1. **Sentiment and Emotion Detection**:

The Emotion Analyzer classified each page into multiple emotion categories, assigning a score between 0 and 1 for each emotion. These scores represented the intensity of the respective emotions within the text. For example, a page from *Anna Karenina* might score highly for sadness and anger during moments of tragedy, while a page from *Siddhartha* might predominantly exhibit neutrality and joy.



Graph 5: Average Emotion Distribution Across Chapters for Anna Karenina [10]



Graph 6: Page-Level Emotion Distribution for Siddhartha [11]

2. Mapping and Aggregation:

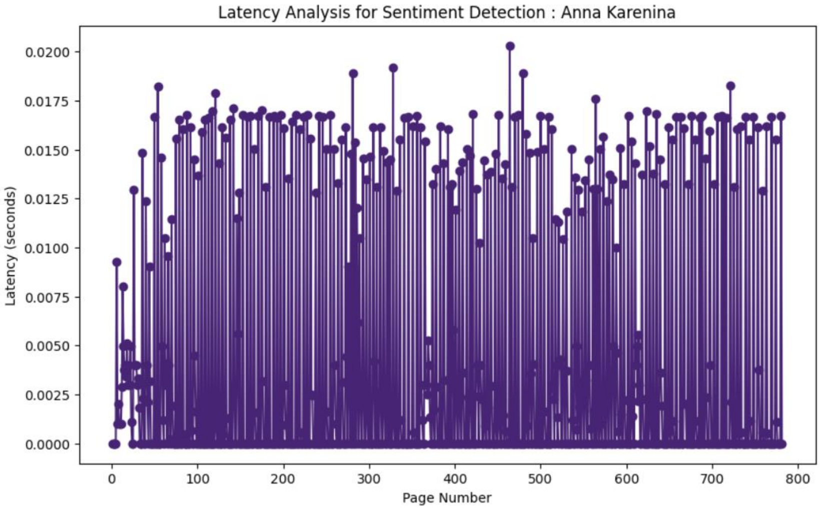
The raw emotion scores were aggregated to analyze trends across chapters and key narrative moments. This step allowed us to identify recurring emotional patterns and significant shifts in tone. Pages were grouped by chapters for comparative analysis, enabling the identification of emotional arcs within each book.

3. Visualization:

Heatmaps were generated to visually represent the distribution of emotions across pages and chapters. These heatmaps provided an intuitive overview of emotional intensity and variation throughout the books. Additionally, summary statistics, such as mean and standard deviation of emotion scores per chapter, were calculated to quantify the emotional variability within each narrative.

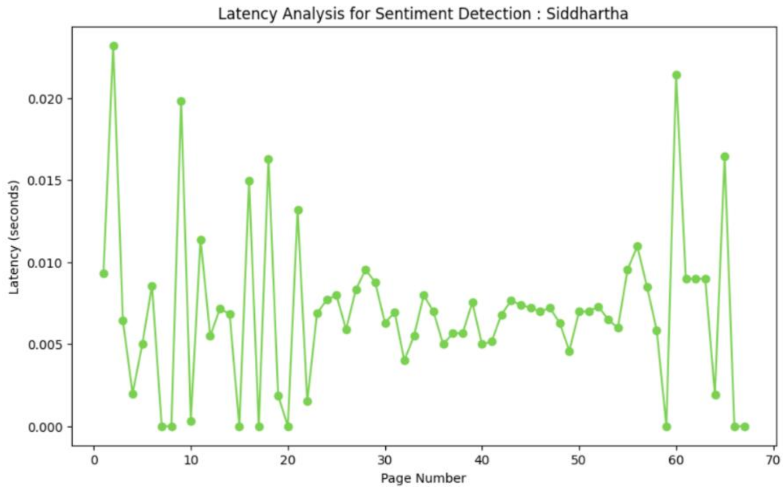
3.4.4. Results and Analysis:

The latency analysis for sentiment detection in *Anna Karenina* and *Siddhartha* reveals critical insights into the computational demands of processing text for emotional analysis.



Graph 7: Line Chart Latency Analysis for Sentiment Detection Anna Karenina [12]

The line graph above the 'Graph 7' definition shows that latency fluctuates significantly across 800 pages, with frequent peaks reaching as high as 0.02 seconds. This variability suggests that the book's length, complexity, and intense emotional content place greater demands on natural language processing (NLP) models.



Graph 8: Latency Analysis for Sentiment Detection: Siddhartha [13]

In contrast, *Siddhartha,* as shown in Graph 8 , a shorter book of approximately 70 pages, demonstrates much steadier latency, rarely exceeding 0.02 seconds. The reduced textual complexity and consistent processing times in *Siddhartha* make it less resource-intensive and better suited for maintaining real-time system responsiveness.

The latency analysis highlights how *Anna Karenina* experiences significant variability, with frequent peaks disrupting real-time updates like music and visuals, potentially harming user immersion. In contrast, *Siddhartha* shows stable and low latency, ensuring seamless and engaging interactions. Longer, more complex texts strain scalability and system resources, requiring optimization.

As a result of these benchmarks, to handle latency issues with longer books we can focus on optimizing the sentiment analysis process. By using lighter NLP models or breaking text into smaller segments for analysis, we can reduce processing demands. Preprocessing and caching frequently accessed sections allow faster retrieval, while cloud-based scaling and load balancing help manage high resource usage during peak times. A hybrid approach can combine quick, simpler analysis for real-time updates with detailed processing in the background. Adding user-friendly features, like adjustable emotion sensitivity and smoother transitions, ensures a seamless and immersive experience even for longer, more complex texts which can increase user experience.

3.4.5. Used Tools

The following tools and libraries were employed during the analysis:

- Hugging Face Transformers: For pre-trained emotion classification models.
- Python Libraries:
    - pandas: To structure and manipulate the data.
    - matplotlib and seaborn: For generating heatmaps and visualizations.
    - *PyPDF2*: For extracting text from PDF files.

3.4.6. Recommendations and Future Work

Within the scope of our BooTunes project, we identified many areas where the system can be improved according to the analysis results. These recommendations aim to improve the functionality of the system and better align it with the goals of providing an optimal and personalized reading experience.

Address Ambiguous Emotions: During the analysis, some text passages were difficult to classify due to overlapping or subtle emotional cues. We propose incorporating additional models or hybrid methods that can better handle these ambiguities, ensuring more precise emotion detection.

Broaden Testing Across Genres: To make the Emotion Analyzer more versatile, we plan to test it on a wider range of book genres, such as thrillers, science fiction, or autobiographies. This will help us assess how well the system adapts to different narrative styles and emotional dynamics.

Refine Neutral Scores: We noticed that many passages in *Siddhartha* were classified as neutral, reflecting its philosophical tone. To improve this, we suggest linking neutral scores more closely with context, such as meditative or descriptive content, to provide a richer multimedia experience.

Optimize Real-Time Processing: While the latency analysis demonstrated consistent processing times, optimizing the system further could ensure smoother real-time emotion detection and multimedia synchronization, especially for lengthy books.

Incorporate User Feedback: Adding a feedback mechanism would allow readers to share their experiences with the multimedia elements and their alignment with the narrative's emotional tone. This feedback could then be used to fine-tune the recommendation engine and improve the overall system.

Enhance Personalization Features: Building on the current recommendation engine, we aim to tailor further suggestions for music and visuals to individual user preferences, such as specific genre interests.

These recommendations outline the next steps we will take to refine and expand BooTunes, ensuring it becomes a robust tool for enhancing digital reading experiences. Each of these steps aligns with our goal of creating a system that is not only accurate and efficient but also adaptable to a diverse range of user needs.

3.5. Access Control and Security

Authentication: The system uses authentication to ensure secure user access. Users must provide credentials, such as passwords, to access their accounts. This prevents unauthorized access and protects user data.

Role-Based Access Control (RBAC): Access to subsystems and administrative functions is restricted based on user roles (e.g., regular users, administrators). For instance, only administrators can update NLP models or manage the music library, ensuring secure and controlled system operations.

Data Compliance: BooTunes adheres to GDPR and other global data protection standards to ensure that user information, such as preferences and reading history, is securely stored and processed.

3.6. Global Software Control

The BooTunes system maintains a centralized control structure to synchronize interactions between its subsystems. When the **Emotion Analysis Subsystem** detects a change in emotional tone, it triggers simultaneous updates in the **Music Management** and **Visual Rendering** subsystems. This ensures that music and visuals remain aligned with the text's emotional context. The centralized control also manages dependencies, such as updating recommendations when new preferences are detected, ensuring a cohesive and dynamic user experience.

3.7. Boundary Conditions

- **Offline Mode:** Users can download books, music and images for uninterrupted access during periods when they are offline. Preloading enables emotion-based multimedia transitions to be used without requiring an internet connection.

- **Error Handling:** The system includes elegant degradation techniques to minimize user disruptions during network failures or API outages. For example, if the API from which Music tracks are retrieved is unavailable, the preloaded tracks are used as a backup.
- **User Personalization:** Users can manipulate settings such as visuals and music preferences to tailor the experience to their liking. These adjustments allow for greater personalization and inclusivity.

4. Subsystem Services

Emotion Analysis: Extracts emotional metadata from text content using advanced NLP techniques. This metadata is shared with the Music and Visual subsystems to create a synchronized experience.

Music Management: Matches perceived emotions with appropriate musical pieces and aims to convey them seamlessly. Integration with music APIs to use provides a diverse and high-quality music library.

Visual Rendering: Produces dynamic visuals that match the emotional tone of the text. These images are created by AI or taken from pre-installed libraries for offline mode as well.

Recommendation Engine: Analyzes user preferences, history, and interactions to suggest books and playlists that align with the user's emotional and thematic interests.(Burasi olcak kalsin)

User Feedback Mechanism: Collects real-time feedback on music, visuals, and overall experience. This data is used to refine emotion analysis models and enhance the personalization of recommendations.

5. Glossary

- **AI (Artificial Intelligence)**: A technology that enables systems to perform tasks typically requiring human intelligence, such as emotion detection and recommendation generation.
- **NLP (Natural Language Processing)**: A subfield of AI focused on processing and understanding human language. It powers BooTunes' emotion detection by analyzing textual content.
- **API (Application Programming Interface)**: A set of protocols that allow software components to interact. Included for integration and broadcasting of music tracks.
- **Emotion Analysis**: The process of identifying emotional tones within text that is central to providing mood-appropriate music and visuals in BooTunes.
- **GDPR (General Data Protection Regulation)**: A legal framework regulating data privacy and security. BooTunes complies with GDPR by protecting user data and offering transparent data management policies.

6. References

- [1], [2], [3], [4], [5]: LucidChart (diagrams) (https://www.lucidchart.com)
- [6], [7], [8], [9], [10], [11], [12], [13]: Python Libraries - Hugging Face Transformers
- Spotify API Documentation (https://developer.spotify.com/documentation/web-api)
- GoEmotions Dataset for Emotion Detection (https://www.kaggle.com/datasets/debarshichanda/goemotions)
- MongoDB Documentation (https://www.mongodb.com/docs/)
- React and React Native Development Guides (https://reactnative.dev/)
- 10 Best Customer Feedback Tools to Improve Customer Satisfaction in 2024(https://clickup.com/blog/customer-feedback-tools/?utm_source=google-pmax&utm_medium=cpc&utm_campaign=gpm_cpc_ar_nnc_pro_trial_all-devices_tcpa_lp_x_all-departments_x_pmax&utm_content=&utm_creative=_____&gad_source=1&gclid=CjwKCAiAmrS7BhBJEiwAei59i2ab9yMAqyhsKE1d6wtvchzWJ2VAyUUqlmV4FvqyfqQyOxnCRL28gRoCOjIQAvD_BwE)
- PlantUML (https://plantuml.com/)