

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING THE UNIVERSITY OF TEXAS AT ARLINGTON

PROJECT CHARTER CSE 4316: SENIOR DESIGN I SPRING 2025



THE BSC FANTIER

AKITO KISHI
ANDREW GRAZIANO
ALI SHIRAZI-NEJAD
GREGORY JOLLY
MUJTABA ELSHAIKH ELRAYAH

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	02.28.2025	AG, AK, AS, GJ, ME	document creation
0.2	03.07.2025	AG, AK, AS, GJ, ME	complete draft
0.3	03.07.2025		release candidate 1
1.0	03.07.2025		official release
1.1	03.07.2025		added customer change requests

CONTENTS

1 Problem Statement	6
2 Methodology	6
3 Value Proposition	6
3.1 Value for Sponsors and Stakeholders	6
4 Development Milestones	7
5 Background	8
6 Related Work	8
7 System Overview	9
8 Roles & Responsibilities	10
9 Cost Proposal	11
9.1 Preliminary Budget	11
9.2 Current & Pending Support	11
10 Facilities & Equipment	11
11 Assumptions	12
12 Constraints	12
13 Risks	12
14 Documentation & Reporting	13
14.1 Major Documentation Deliverables	13
14.1.1 Project Charter	13
14.1.2 System Requirements Specification	13
14.1.3 Architectural Design Specification	14
14.1.4 Detailed Design Specification	14
14.2 Recurring Sprint Items	14
14.2.1 Product Backlog	14
14.2.2 Sprint Planning	14
14.2.3 Sprint Goal	14
14.2.4 Sprint Backlog	15
14.2.5 Task Breakdown	15
14.2.6 Sprint Burn Down Charts	15
14.2.7 Sprint Retrospective	15
14.2.8 Individual Status Reports	16
14.2.9 Engineering Notebooks	16
14.3 Closeout Materials	16
14.3.1 System Prototype	16
14.3.2 Project Poster	16

14.3.3 Web Page	16
14.3.4 Demo Video	16
14.3.5 Source Code	16
14.3.6 Source Code Documentation	17
14.3.7 Hardware Schematics	17
14.3.8 CAD files	17
14.3.9 Installation Scripts	17
14.3.10 User Manual	17

LIST OF FIGURES

1	Example sprint burn down chart	15
---	--	----

1 PROBLEM STATEMENT

Have you ever wanted to see your favorite artist but Ticket Master and other ticket vendors keep giving you the run around. You're not just a normal fan you're the die hard fan, the one who has every song memorized the day it drops, the one who collects all the merch drops, the top <1% of listener but you're always getting the short end of the stick. This years notable concert tours have sold out within minutes (and sometimes even less than a minute). Taylor Swift, Beyonce and The Weeknd's concerts all sold out within minutes and left a lot of die hard fans truly disappointed that they won't get to see their favorite artist perform.

There's no current perfect form of ticketing vendors and almost each platform has been hit with allegations of running pay to play schemes and other ethical violations when it comes to popular concert sales. This is an industry that needs new innovations both for the fans (buyers) and artists. The current ways of ticketing vendors have left some fans hopeless of ever getting to hear their favorite artist perform which is just saddening.

2 METHODOLOGY

For this issue we've decided to create a new concept ticket vendor for artists to make sure their top supporters get priority access to their shows called FanTier. How this would work is it would be a concert ticketing vendor which prioritizes top listeners of a specific artist to get preferential access to tickets based off their streaming data from Spotify API and Apple Music API. The WebApp & PhoneApp would consume Spotify and Apple Music streaming data and calculate where you fall within a ranking of the artists listeners (Ex. Top 0.0005% of Taylor Swift listeners) to a certain artist and then assign your rank a preferential status when it comes time to purchase tickets for the artists concerts. The WebApp and PhoneApp would also include social and community integration aspects through fan leaderboards and community forums where fans can see how they rank against other listeners and build a social space for fans to discuss upcoming shows, share concert experiences, and connect over streaming achievements. This would help an artist show they care about their die hard fans and reward loyal listeners which would strengthen an artists core fan base and excite fans to stream their music (thus generating them more money). Being a ticketing vendor can be a win-win situation for fans and artists but the recent reality of it has been far from that.

3 VALUE PROPOSITION

Our project provides a fair and engaging ticketing system that rewards dedicated music fans by leveraging their streaming data from Spotify and Apple Music. By integrating listening history into the ticket-buying process, our system ensures that the most dedicated fans receive better positions in the queue, reducing the impact of ticket scalping and reselling.

3.1 VALUE FOR SPONSORS AND STAKEHOLDERS

Music Streaming Services (Spotify, Apple Music)

- Increases user engagement by incentivizing continued platform usage.
- Strengthens artist-fan relationships, fostering loyalty to streaming services.
- Provides valuable insights into user listening behavior for targeted marketing and artist promotion.

Ticketing Platforms and Promoters

- Reduces scalping and bot activity by prioritizing real music fans.

- Improves reputation and trust in ticketing platforms.
- Encourages partnerships with artists seeking fairer ticketing solutions.

Artists and Record Labels

- Ensures dedicated fans have better access to concerts.
- Provides real-time fan engagement metrics for marketing and tour planning.
- Increases streaming numbers, generating additional revenue.

Universities and Academic Sponsors

- Demonstrates innovative applications of data science and software development.
- Showcases expertise in data privacy, API integration, and ethical fan engagement.
- Serves as a case study for real-world digital marketing and data-driven decision-making.

By aligning the interests of streaming platforms, ticketing companies, artists, and fans, our application creates a win-win ecosystem that enhances fan engagement while addressing key industry challenges.

4 DEVELOPMENT MILESTONES

This list of core project milestones should include all major documents, demonstration of major project features, and associated deadlines. Any date that has not yet been officially scheduled at the time of preparing this document may be listed by month.

- Project Charter first draft - March 2025
- System Requirements Specification - March 2025
- Architectural Design Specification - March 2025
- CoE Innovation Day poster presentation - April 16 2025
- Demonstration of Spotify and Apple Music data processing - end of April 2025
- Detailed Design Specification - May 2025
- Demonstration of ticket purchasing and queuing system - June 2025
- Demonstration of social media aspect - Month 2025
- Demonstration of full FanTier Website - beginning of July 2025
- Demonstration of full IOS FanTier app - end of July 2025
- Demonstration of Buying mock tickets and the social media aspects - August 2025
- Final Project Demonstration - August 2025

5 BACKGROUND

The Business Case for FanTier: Placing Committed Fans First in Concert Tickets.

The contemporary ticketing industry is riddled with flaws that hurt both musicians and their most loyal fans. Most ticket vendors mainly use a first-come-first-served model or use lottery systems that often don't put loyal fans first. Moreover, scalpers and bot programs take advantage of these systems by buying large numbers of tickets with no real intention of watching the event with the sole aim of selling them at a higher price. This practice makes it more difficult for real fans to acquire tickets at an affordable price. The gap between musicians and their loyal fans is a significant challenge that FanTier seeks to bridge by revolutionizing the ticketing process.

FanTier presents a unique solution that uses streaming activity from both Apple Music and Spotify APIs to determine a person's level of engagement with a particular artist. In this model, loyal fans—who have consistently supported an artist through their streaming behavior—are provided with preferential access to tickets for a performance. This model not only remunerates an artist's most loyal fans but also fosters greater streaming behavior and therefore creates a direct revenue stream. By rolling out this model on both a mobile app and a web app, FanTier provides a seamless and convenient experience that classifies people on their listening behavior while providing tiered ticketing privileges.

In addition to its ticketing functionality, FanTier has a social and community element that allows fans to gauge their ranking in relation to other listeners, discuss upcoming concerts with one another, leave personal recollections, and bond over their appreciation of an artist's music. This higher level of interaction creates loyalty among fans, develops a community atmosphere, and at the same time encourages more streaming to artists' advantage.

From a business standpoint, FanTier creates a mutually advantageous model. Fans are given fair access to tickets, and artists are guaranteed that their most loyal fans will be in attendance at performances. In addition, the platform itself benefits from a loyal base of customers. By balancing both artist and fan interests, FanTier is a more attractive option than current ticketing providers that have been criticized for putting profits over people.

The prevailing model is one where large corporations have very little incentive to make changes that would be positive for fans or artists. FanTier seeks to disrupt this current model by creating a more balanced and interactive ticketing system. By combining streaming with live performance, FanTier overcomes the flaws in current systems while at the same time making artists and fans' relationship more positive and better access and value to concerts for those who have already proven their commitment.

6 RELATED WORK

Currently, the concert ticket industry faces many significant challenges, particularly in allowing fans to access tickets reasonably and at an affordable cost. The current situation with Ticketmaster and other websites has been widely criticized for enabling scalping, facilitating high fees, and creating a frustrating experience for consumers. The rapid sellout of major concert tours, such as Taylor Swift, and the increase of scalpers have highlighted the issue in the current ticketing landscape [1].

Existing Solutions

- **Face Biometrics for Ticket Authentication:** Currently, some Korean companies have been exploring the use of biometric authentication through facial scanning. This would result in an individual's facial recognition data being their ticket and would significantly reduce unauthorized resale. However, such solutions raise privacy concerns and require widespread adoption of biometric technology at venues [2].
- **Blockchain-Based Ticketing Systems:** Protocols such as Aventus utilize blockchain technology to create a transparent and verifiable ticketing process. Which provides a uniquely traceable ticket

that can prevent fraud and scalping. However, blockchain-based systems often face scalability issues and may introduce complexities in user adoption [3].

- **Dynamic Pricing:** One of the most controversial practices in ticketing today is dynamic Pricing, where ticket prices fluctuate based on demand. This strategy, often used by major vendors like Ticketmaster, has led to extreme price surges, making it difficult for many fans to afford tickets [1]. In addition, secondary markets and resale platforms continue to be significant issues. While some businesses have implemented policies to cancel resold tickets, such as the Oasis reunion tour [4], enforcement remains inconsistent, and resellers still manage to exploit loopholes. As a result, loyal fans often compete against professional resellers rather than being prioritized for ticket access.
- **Fraud Detection and Anti-Scalper:** Companies like TrustDecision created extensive solutions to fight ticket scalping by using AI and machine learning algorithms. Their platform provides real-time fraud detection, behavioral analysis, and a configurable rules engine that is meant to identify and hinder suspicious activity, thus ensuring a fair ticketing experience for genuine fans. Although this solution incorporates AI to ward off scalpers, our solution does not need many algorithms to ward off, but a more straightforward solution to help the loyal fanbase instead. [5].

7 SYSTEM OVERVIEW

The FanTier system will consist of user authentication, streaming data integration, a ranking algorithm, a ticketing system, and a social media platform for fans to interact.

User Authentication

- Users will sign in using Spotify or Apple Music in order to authenticate the member.
- A user profile will be created, storing relevant streaming data and preferences.

Streaming Data Integration

- APIs from Spotify and Apple Music will provide user listening history.
- Data processing will calculate the users ranking within an artists listener base.

Ranking Algorithm

- The system will determine fan tiers (e.g., Top 0.01%, Top 1%) based on listening time, song repeats, and engagement.
- A scoring mechanism will be implemented to ensure fairness and prevent manipulation.

Ticketing System

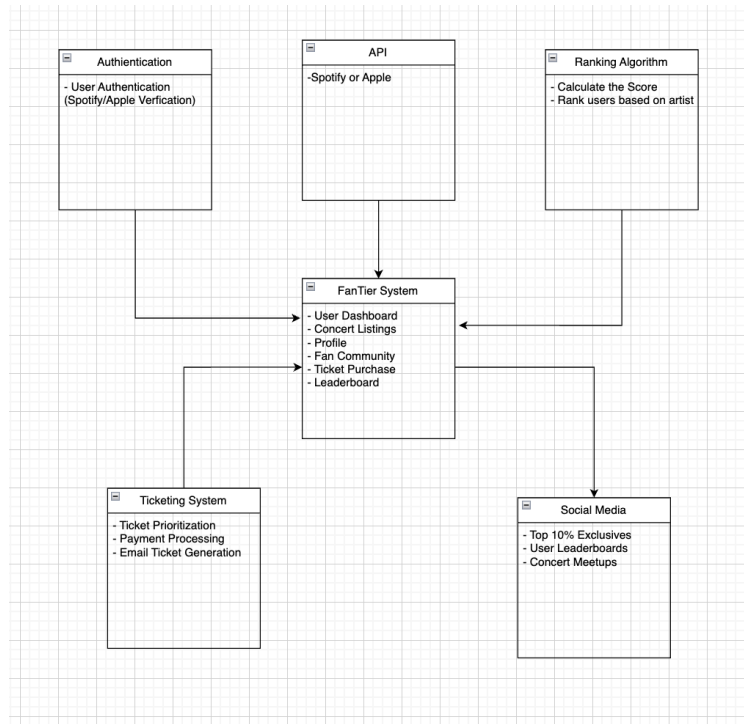
- When an artist announces a concert, the system will prioritize ticket availability based on fan tier rankings.
- Users in higher tiers will have earlier or exclusive access to tickets.

Web and Mobile Interface

- Users can view upcoming concerts, and fan community forums.
- Fans can interact, discuss, and communicate via location.

Payment and Ticket Distribution

- Secure payment processing using a Payment API.
- Digital ticket email generation.



8 ROLES & RESPONSIBILITIES

The FanTier project brings together a multitude of stakeholders, their interests and contributions are essential to the product's success. Critical stakeholders include artists and their management teams who get to please their die hard fans and make sure they are taken care of. This will ensure that die hard fans see that the artist cares and wants to reward them for their loyalty. Theoretically this would inspire fans to adore the artist more. The top-tier listeners (fans) are also important stakeholders, as they drive engagement in the application by using the social media aspect and purchasing ticket. The fans will ultimately define the product's success. Strategic partners include the streaming services which play a significant role by offering the data and APIs necessary for accurately ranking and prioritizing ticket access.

Team Roles and Responsibilities

- Ali Shirazi - Product Owner

Ali will oversee the team and help to guide the overarching product vision and ensure that key priorities are clearly communicated and adhered to throughout the development process. Ali will also be the point of contact from the sponsor and customer sides. This role may change depending on whether Ali's expertise could be used elsewhere as he is a very proficient programmer and might be able to utilize his skill set to aid the team.

- Andrew Graziano - Scrum Master

Andrew will host scrum calls, stand-ups, help communicate blockers, conduct sprint plannings, identify backlogged items and oversee the retro after each sprint. The scrum master is like the messenger of the group he makes sure everyone is getting things done and have an idea of what everyone is working on, he is responsible for making sure there is a strict adherence to Agile principles. The job of Scrum Master is a complicated one, but Andrew has proved that he is a good

communicator, and this role will most likely be permanent.

- **Mujtaba Elshaikh - Lead Developer**
Mujtaba has extensive experience on various projects and in multiple programming languages which makes him an ideal candidate for the role of Tech Lead. In this role he will coordinate both the front-end and back-end development efforts to ensure seamless integration and compatibility. His responsibilities will also include supporting our stretch goal of developing a mobile app.
- **Gregg Jolly with assistance from the entire team - Lead Front End Developer**
Gregg will lead the front end developer aspect of the project and focus specifically on creating an intuitive and responsive user experience through the WebApp and phone app (stretch goal). Although Gregg will lead this aspect of the project it will be with the assistance of the entire team as it is a big task that will need to be broken down.
- **Akito Kishi with assistance from the entire team - Lead Backend Developer**
Akito will lead the back end developer aspect of the project and focus specifically on creating the core functionalities of FanTier such as calculating user ranks and creating a fan tier system for getting the tickets. Although Akito will lead this aspect of the project it will be with the assistance of the entire team as it is a big task that will need to be broken down.

Some of these roles may not be permanent and overlap may exist but with the guidance of the Scrum Master and Product Owner these will be managed efficiently and swiftly.

9 COST PROPOSAL

We are utilizing the Spotify Web API alongside AWS services, predominantly on the free tier, to build a robust and scalable solution. The only nominal expense is the domain registration fee, which supports our web hosting needs and represents a strategic investment in establishing a professional online presence.

9.1 PRELIMINARY BUDGET

Item	Cost (USD)
Spotify Web API	\$0
AWS Services (Free Tier)	\$0
Web Hosting (Domain Name)	\$12
Total	\$12

Table 1: Cost Proposal for Project

9.2 CURRENT & PENDING SUPPORT

Funding for this project is provided through institutional support from the university or personal expense.

10 FACILITIES & EQUIPMENT

The FanTier project is entirely software-based, so there will be no need for physical lab space, testing grounds, or makerspaces will be required. All development, testing, and deployment will be conducted remotely using cloud, personal computers, and online tools.

Each team member will use personal computers for development and testing. No additional hardware such as a physical testing environments, or specialized equipment will be required. Since we are not using university resources, there is no need for school lab space or borrowed equipment.

All of the code we will be using will be contained in github and planned through Jira. For our software development, all of the code and frameworks used will be self-contained and won't require us to take up physical space.

11 ASSUMPTIONS

Here are the assumptions for our project:

- We assume that the Spotify and Apple Music APIs will reliably provide accurate and real-time streaming data, which is essential for calculating listener rankings and determining preferential ticketing status.
- We assume that users will consent to sharing their streaming data, enabling the platform to generate personalized fan rankings and ensure preferential ticket access.
- We assume that any external dependencies, such as collaborations with industry partners or access to specialized facilities, will be confirmed and secured early in the project timeline.

12 CONSTRAINTS

The following list contains key constraints related to the implementation and testing of the project.

- Any third-party APIs or tools (such as the Spotify and Apple Music APIs) are subject to their own usage limits and availability, which may impact the project if these services experience downtime or changes in their access policies.
- Total project costs must not exceed \$800, including any hardware purchases, software licenses, and incidental expenses.
- The final prototype demonstration must be completed and presented by end of summer 2025 semester
- Complete documentation, including user manuals, technical reports, and testing results, must be prepared and submitted by the project deadline. This documentation is required for assessment and may affect the project timeline.
- : All project components, including data collection and testing methods, must adhere to university policies and applicable government regulations. Data usage and storage must be reviewed and approved by the designated Information Security Office before any external transmission.

13 RISKS

This section contains the most critical risks related to the FanTier ticketing project. Each risk lists its probability of occurrence, the potential loss in days, and an exposure value (risk exposure in days = probability x loss). The following table summarizes these high-exposure risks. Mitigation strategies will be discussed in future planning sessions.

- **Streaming API Unavailability:** Implement caching strategies, maintain direct communication with API providers.

Risk description	Probability	Loss (days)	Exposure (days)
Streaming API Unavailability (e.g., Spotify/Apple Music temporarily revoke or limit API access)	0.30	14	4.2
High-Traffic Bottlenecks (performance issues during major ticket sale windows)	0.25	12	3.0
Regulatory or Privacy Noncompliance (delays in approvals due to data governance concerns)	0.20	15	3.0
Payment Gateway Integration Delays (payment processing or vendor issues)	0.20	15	3.0
Data Security Breach (compromised user credentials or ticketing transactions)	0.10	30	3.0

Table 2: Overview of highest exposure project risks

- **High-Traffic Bottlenecks:** Prepare load tests and auto-scaling solutions for peak ticket sale windows.
- **Regulatory or Privacy Noncompliance:** Partner with legal experts early for data handling and user consent flows.
- **Payment Gateway Integration Delays:** Integrate multiple payment providers to reduce single-point-of-failure risks.
- **Data Security Breach:** Conduct regular security assessments and enforce best practices for authentication.

14 DOCUMENTATION & REPORTING

14.1 MAJOR DOCUMENTATION DELIVERABLES

Each of the following deliverables constitutes a major grade component and ties directly to the FanTier development goals and milestones. They will be created, reviewed, and adjusted over the course of the project, with final versions delivered at closeout.

14.1.1 PROJECT CHARTER

The Project Charter outlines the project's scope, objectives, stakeholders, and success criteria for FanTier.

- **Maintenance/Updates:** The charter will be updated whenever significant changes arise (e.g., shifting requirements, new sponsors).
- **Initial Delivery:** During the first sprint, once project roles and responsibilities are defined.
- **Final Delivery:** Prior to project closeout, reflecting any final alterations to scope or objectives.

14.1.2 SYSTEM REQUIREMENTS SPECIFICATION

The SRS fully documents functional and non-functional requirements for FanTier, including integration with streaming APIs and user authentication flows.

- **Maintenance/Updates:** Updated at least once per sprint to capture new or modified requirements.
- **Initial Delivery:** Early in development (around Sprint 1 or 2), once initial requirements are gathered.
- **Final Delivery:** Immediately before design specifications are frozen, reflecting any refinements from stakeholder feedback.

14.1.3 ARCHITECTURAL DESIGN SPECIFICATION

This document details the high-level architecture of FanTier, including system components, interfaces (Spotify/Apple Music), and data flows for ticket prioritization.

- **Maintenance/Updates:** Adjusted as new constraints, technologies, or performance considerations emerge.
- **Initial Delivery:** Once the foundational system approach is agreed upon (roughly mid-semester).
- **Final Delivery:** At the end of the primary design phase, before detailed implementation begins.

14.1.4 DETAILED DESIGN SPECIFICATION

The Detailed Design Specification breaks down system modules, algorithms, class designs, and database schemas for FanTier's WebApp and PhoneApp.

- **Maintenance/Updates:** Updated iteratively during implementation to capture design changes.
- **Initial Delivery:** As soon as major architectural decisions have been made (e.g., sensors or external service dependencies).
- **Final Delivery:** Right before moving into full testing and integration phases.

14.2 RECURRING SPRINT ITEMS

The team will produce or maintain the following items each sprint to track progress and optimize collaboration for FanTier's feature delivery.

14.2.1 PRODUCT BACKLOG

- **Source:** Primarily derived from the SRS and stakeholder feedback.
- **Prioritization:** Managed by the Product Owner with input from the development team, using a consensus-based approach or priority-based ranking.
- **Tool:** Maintained via a project management platform (e.g., Jira, Trello) so team members and stakeholders can access and update it.

14.2.2 SPRINT PLANNING

- **Process:** Conducted at the beginning of each sprint, reviewing backlog items, estimating effort, and defining acceptance criteria.
- **Number of Sprints:** Aligned with the course schedule (e.g., 4–6 sprints), each spanning approximately 2–4 weeks.

14.2.3 SPRINT GOAL

- **Ownership:** Formulated collaboratively by the Product Owner, stakeholders, and the team to ensure each sprint delivers meaningful progress.
- **Customer Involvement:** The sprint goal will be validated by the customer's representative, ensuring alignment with user needs.

14.2.4 SPRINT BACKLOG

- **Content:** Subset of the highest-priority backlog items plus any immediate bug fixes or feedback from previous sprints.
- **Tracking:** Managed dynamically in the team's Scrum board; tasks are added, updated, or removed throughout the sprint as needed.

14.2.5 TASK BREAKDOWN

- **Assignment:** Team members choose or are assigned tasks based on expertise or load balancing.
- **Time and Effort Tracking:** Documented within project management software (hours logged, story points completed).

14.2.6 SPRINT BURN DOWN CHARTS

- **Responsibility:** One designated team member updates the burn down chart daily.
- **Format:** A visual chart (in Jira or a shared spreadsheet) illustrating remaining work versus time.
- **Usage:** Monitored in daily stand-ups to identify risks or bottlenecks early.



Figure 1: Example sprint burn down chart

14.2.7 SPRINT RETROSPECTIVE

- **Timing:** Occurs directly after each sprint concludes.
- **Goals:** Identify successes, note areas needing improvement, and suggest actionable steps for the next sprint.
- **Documentation:** Captured in shared meeting notes or a retrospective tool so that lessons learned are tracked over time.

14.2.8 INDIVIDUAL STATUS REPORTS

- **Frequency:** Submitted by each team member at the end of every sprint.
- **Content:** Summaries of tasks completed, issues encountered, and plans for the next sprint.
- **Visibility:** Shared with the Product Owner and archived for reference.

14.2.9 ENGINEERING NOTEBOOKS

- **Updates:** Each team member documents research, test findings, or technical decisions at least weekly.
- **Accountability:** Team members sign off on each other's notebooks periodically; a rotating "note-book reviewer" ensures consistency and completeness.

14.3 CLOSEOUT MATERIALS

Upon project completion, the following materials will be delivered to the customer or sponsoring stakeholders to ensure a smooth handover and clear continuation path.

14.3.1 SYSTEM PROTOTYPE

- **Content:** Functional or partially functional demo of FanTier (supporting artist prioritization, streaming data fetching, etc.).
- **Acceptance Tests:** A formal Prototype Acceptance Test (PAT) with stakeholders to confirm functionality and user satisfaction.

14.3.2 PROJECT POSTER

- **Dimensions:** Typically 36" x 48" or as specified by the course guidelines.
- **Contents:** Problem statement, methodology, key results, and team roles.
- **Delivery:** Presented during the final showcase or demonstration event.

14.3.3 WEB PAGE

- **Goal:** Publicly accessible summary of FanTier's purpose, features, and team.
- **Delivery and Updates:** Deployed towards the end of the project, with minimal updates post-closeout unless otherwise agreed.

14.3.4 DEMO VIDEO

- **Scope:** Showcases core features (e.g., retrieving streaming data, prioritized queue, ticket purchase flow).
- **Format:** Approximately 5–10 minutes of narrated screen recordings and/or live footage.
- **Additional Assets:** Optional B-roll for future marketing or reference.

14.3.5 SOURCE CODE

- **Version Control:** Maintained in a Git repository (e.g., GitHub or GitLab).
- **Customer Delivery:** Provided as a private repo link or a final downloadable archive. May be open-sourced under an MIT or GPL license upon sponsor approval.

14.3.6 SOURCE CODE DOCUMENTATION

- **Creation:** Automatic documentation generated via tools like Doxygen or Javadoc where possible.
- **Final Format:** Hosted as HTML pages or PDF for offline review, included in the repository or a docs folder.

14.3.7 HARDWARE SCHEMATICS

- **Applicability:** N/A for this project - we have no hardware components such as scanners or kiosks.
- **Content:** Would typically include PCB diagrams, wiring layouts, sensor details, etc., if hardware were used.

14.3.8 CAD FILES

- **Applicability:** N/A for this project - no enclosures or other physical parts will be 3D-printed or manufactured.
- **Content:** Would typically include STL, STEP, or other formats for enclosures, ticket scanners, kiosks, etc.

14.3.9 INSTALLATION SCRIPTS

- **Scope:** Automated deployment scripts for setting up server environments, databases, or front-end components.
- **Maintenance:** Documented in the repository's README for ease of future updates or reinstallation.

14.3.10 USER MANUAL

- **Format:** A PDF or online help resource detailing step-by-step usage, FAQs, and troubleshooting instructions.
- **Supplement:** If necessary, a short setup or training video can accompany the manual to illustrate the ticket-purchasing workflow.

REFERENCES

- [1] V. Media, "Why concert tickets are so expensive," *Vox*, 2025.
- [2] IDTechWire, "Korean firms look to face biometrics to fight ticket scalping," *IDTechWire*, 2024.
- [3] Wikipedia, "Aventus (protocol)," 2024.
- [4] B. Allcock, "Oasis fans fuming after ticketmaster suddenly cancels tickets accuses them of being botsâ leaving dreams crushedâ," *TheIrishSun*, 2025.
- [5] Tanya, "Securing your event: Anti-scalping measures and best practices," *trustdecision*, 2024.