

Ntuaflix

Software Requirements Specification

Version 2.0

Revision History

Date	Version	Description	Authors
16/12/2023	1.0	First Composition of SRS Document	Antonios Alexiadis Nikolaos Bothos Chrisostomos Kopitas Charidimos Papadakis
31/01/2024	2.0	Final Composition of SRS Document	Antonios Alexiadis Nikolaos Bothos Chrisostomos Kopitas Charidimos Papadakis

Table of Contents

Revision History	2
Table of Contents	3
Introduction.....	4
Purpose.....	4
Scope.....	5
Definition, Acronyms and Abbreviations	5
Acronyms.....	5
References.....	6
Overall Description.....	6
Problem Statement.....	6
Background.....	6
Objectives	7
Constraints	8
Functional Requirements.....	8
Use Cases.....	9
Use Case Diagram	9
Requirements	11
Requirement Diagram.....	11
User Requirements.....	13
Technical Requirements.....	13
Performance	13
Scalability	14
Security	14
Maintainability.....	14
Usability.....	14
Localization Support.....	14
Auditing and Logging.....	14
Availability.....	14
Hardware Requirements	15
Network	15
Client Computers	15
Deployment Requirements	16
Deployment Diagram.....	16
Nodes	17
Components	18

Introduction

This SRS document is structured to provide a clear and organized presentation of all system requirements. It is divided into the following main sections:

- **Overall Description:** This section provides a general overview of the software, including its functionality, user demographics, and the problem it aims to solve.
- **System Features:** A detailed description of each system feature, its utility, and how it interacts within the broader system architecture.
- **Use Cases:** Narrative and diagrammatic representations of user interactions with the system, detailing the main success scenarios and possible extensions.
- **Requirements:**
 1. **Functional Requirements:** Specific functionalities that the software must perform, derived from the use cases and additional stakeholder input.
 2. **Non-Functional Requirements:** The standards, performance metrics, and quality attributes that the system must adhere to and any additional constraints or compliance requirements, including regulatory and data privacy considerations.
 3. **User Requirements:** What end-users expect from the system. The needs, goals, and tasks of the user, including accessibility, user interface preferences, and specific functionalities users want to perform with the system.
 4. **Technical Requirements:** The requirements that collectively ensure the system is performant, secure, maintainable, usable, and able to operate in the designated environments while meeting all regulatory and compliance needs.
 5. **Hardware Requirements:** The physical components necessary to support the operation of the system. This includes the servers that host the system, network infrastructure for connectivity, and client devices like computers and smartphones that access the system.
 6. **Deployment Requirements:** The conditions necessary to successfully deploy the software in a live environment.

The main focuses of this document are to:

- Provide clear and precise descriptions of the functional behaviour of the system.
- Outline the performance and quality standards the system must meet.
- Serve as a reference to ensure mutual understanding between the stakeholders and the development team.
- Act as a foundational agreement that specific system functionalities are designed, developed, and tested appropriately.

Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed outline of the functional and non-functional requirements for the development of a film information repository

and recommendation platform, herein referred to as "Ntuaflix". This document is intended to be a comprehensive guide for the stakeholders involved in the design, development, testing, and maintenance of Ntuaflix.

Scope

Ntuaflix will serve as an interactive platform that aggregates and presents metadata about films and television series. It is designed to cater to cinephiles and industry professionals who seek detailed information, such as synopses, cast and crew lists, genre classifications, ratings, reviews, and personalized recommendations. This platform is designed not to host film content directly, but rather to offer extensive metadata and personalized recommendations that enhance users' exploration and discovery of cinema. Its focus is on enriching the cinematic experience through detailed information and tailored suggestions.

Definition, Acronyms and Abbreviations

This section provides definitions, acronyms, and abbreviations for terms used throughout this SRS to ensure clarity and prevent misunderstandings.

Term	Definition
Film Information Repository	A system that aggregates and presents detailed metadata about movies and television series without hosting or streaming the actual media content.
Metadata	Data that provides information about other data. In the context of this system, it refers to details such as film titles, release dates, cast and crew information, synopses, ratings, and reviews.
Recommendation Engine	A component of the system that analyzes user data and metadata to suggest films and TV series to users.
API(Application Programming Interface)	A set of protocols and tools for building software applications, which in this context, allows for interaction with the Ntuaflix platform by third-party developers or applications.

Acronyms

SRS: Software Requirements Specification

UI: User Interface

UX: User Experience

DB: Database

GDPR: General Data Protection Regulation

SEO: Search Engine Optimization

API: Application Programming Interface

References

The following reference materials were utilized to guide the development of this Software Requirements Specification:

- **IEEE Guide to Software Requirements Specifications (IEEE 29148:2011):** A guideline that establishes a set of best practices for writing software requirements specifications.
- **ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models:** An international standard defining the quality attributes required for software design and testing.
- **General Data Protection Regulation (GDPR):** The regulation in EU law on data protection and privacy in the European Union and the European Economic Area.

Overall Description

Problem Statement

The vast and ever-growing cinematic landscape presents a paradox of choice for film enthusiasts: an abundance of options that leads to overwhelm rather than empowerment. Current market solutions face significant challenges that hinder effective interaction with film content:

- 1. Limited Personalization:** Existing platforms lack sophisticated personalization options, leading to generic and often irrelevant recommendations. Users struggle to navigate through the noise to find content that aligns with their tastes and preferences.
- 2. Inadequate Search Capabilities:** The current systems offer basic search functionalities that do not cater to the nuanced needs of cinephiles who wish to search using a combination of attributes such as genre, release year, cast, and ratings.
- 3. Content Discovery:** Users seeking to explore new content based on specific contributors (actors, directors, etc.) find it challenging due to the lack of intuitive linkages and exploration tools.

Ntuaflx aims to address these issues by developing a robust, intuitive, and comprehensive Film Information Repository. The platform is designed to enhance the ability of users to discover and explore movie content, fostering a more informed and connected film community.

Background

The current challenges in the cinematic landscape, as outlined in the Problem Statement, are largely attributable to a combination of rapid technological advancements, evolving user behaviours, and a dynamic entertainment industry.

Technological Constraints: Existing platforms were often developed during a time when the digital landscape was less complex. As technology has advanced, these systems have struggled to adapt to the high volume and diversity of content, leading to limitations in personalization and search capabilities. Moreover, the infrastructures of these systems were not originally designed to handle the massive, concurrent user loads we see today, contributing to performance issues.

User Behavior Trends: Modern users seek high levels of personalization in their online experiences. Legacy platforms were not built with these expectations, leading to a gap between user expectations and platform capabilities, particularly in personalized content discovery.

Industry Evolution: The film industry itself has undergone significant changes. There has been a proliferation of independent films and a surge in the number of streaming platforms, each hosting their exclusive content. This expansion has made the discovery process more fragmented and complex for users who are fans of specific actors or filmmakers but are confronted with inadequate tools to track their careers across different platforms.

Business Environment Changes: With the rise of mobile computing and increased internet accessibility, users are consuming content in fundamentally different ways than they did even a decade ago. The business environment has had to adapt rapidly, often outpacing the ability of existing platforms to keep up. This has led to a disjointed experience where users might have to use multiple services to fulfill all their movie-related needs.

Ntuaflx recognizes these issues and aims to address them by leveraging the latest technological developments to build a system that can scale efficiently, offer advanced personalization and search features, foster a vibrant community, and provide intuitive content discovery tools. By doing so, Ntuaflx will enable a seamless, enriched film exploration experience for a diverse, global user base.

Objectives

Ntuaflx is committed to transforming the way users interact with film metadata by creating an advanced Film Information Repository. The system will achieve the following objectives:

- 1. Enhanced Recommendation:** Develop a sophisticated recommendation system that takes into account user behaviour and interactions to deliver accurate content suggestions.
- 2. Robust Search Functionality:** Implement a multifaceted search feature allowing users to discover movies and TV shows based on a wide range of criteria, such as genre, release year, cast, crew, and user ratings, with rapid response times and high accuracy.
- 3. Intuitive Content Exploration:** Provide users with tools that make it easy to follow their favourite actors, directors, and industry contributors, ensuring they can effortlessly track filmographies, upcoming projects, and related news.
- 4. Seamless User Experience:** Ensure that the platform is intuitive, responsive, and accessible across all devices, providing a seamless experience whether users are at home or on the go.
- 5. Data Privacy and Security:** Prioritize user privacy and data security by employing best practices in data management, adhering to regulations such as GDPR, and allowing users to have control over their data and how it is used.
- 6. Adaptability and Scalability:** Design a system architecture that is both flexible and scalable, capable of evolving with technological advancements and expanding to accommodate an increasing user base and data volume.
- 7. Comprehensive Metadata Management:** Allow for efficient updating and management of movie metadata by contributors and administrators, ensuring the information remains current and accurate.

By meeting these objectives, Ntuaflx aspires to fill the existing gaps in the market and establish a new benchmark for online film information repositories. The platform will serve not just as a vast resource for

movie metadata but also as a catalyst for enhancing user engagement and satisfaction through a personalized and intuitive exploration experience.

Constraints

Ntuaflix's development and operation are subject to the following constraints:

1. Platform Compatibility: The initial release of Ntuaflix will be optimized for web browsers on desktop platforms. While the design will be responsive, native app versions for mobile platforms (iOS and Android) may not be available until later development phases.

2. Data Source Dependence: The accuracy and comprehensiveness of movie metadata are dependent on third-party data providers. Any inaccuracies or omissions in the provided data could affect the quality of information presented in Ntuaflix.

3. Regulatory Compliance: Ntuaflix must adhere to international data protection and privacy laws, which may restrict certain features or the extent of personalization possible, particularly across different regions.

4. Scalability: While designed to be scalable, the initial deployment of Ntuaflix may have limitations in handling an unanticipated surge in user traffic, which could necessitate additional development cycles to enhance system infrastructure.

5. Maintenance Downtime: Scheduled maintenance and updates will result in periodic downtime. Efforts will be made to minimize disruption by scheduling these during off-peak hours.

6. Limited Machine Learning Integration: At its current stage, Ntuaflix does not employ complex machine learning algorithms for movie recommendations. Instead, it relies primarily on user ratings and reviews to suggest movies. This approach may limit the personalization and predictive accuracy compared to systems that use advanced machine learning techniques. Future enhancements may include more sophisticated algorithms for tailored recommendations.

7. Metadata-Only Repository: Ntuaflix will not store or stream actual movies or TV shows but will instead focus on the aggregation and presentation of metadata, such as cast details, synopses, user ratings, and reviews. Users seeking full-length content will be redirected to external services where the content is legally hosted.

8. Internet Dependency: As a web-based platform, Ntuaflix requires a stable and fast internet connection for optimal performance. Users with limited connectivity may not be able to access all features effectively.

By acknowledging these constraints upfront, Ntuaflix sets realistic expectations for stakeholders and informs the development team of the boundaries within which they must design and build the system.

Functional Requirements

Ntuaflix is a comprehensive Film Information Repository that aims to enrich the movie-watching experience by offering a suite of interactive and user-focused features. Below is an overview of the core functionalities:

1. Personalized Recommendations: Ntuaflix currently employs a user-centric approach to curate personalized movie and TV show recommendations. Instead of using complex machine learning models, the platform relies on a robust system that analyzes user ratings and reviews. This method ensures that recommendations are tailored based on collective user preferences and individual engagement patterns.

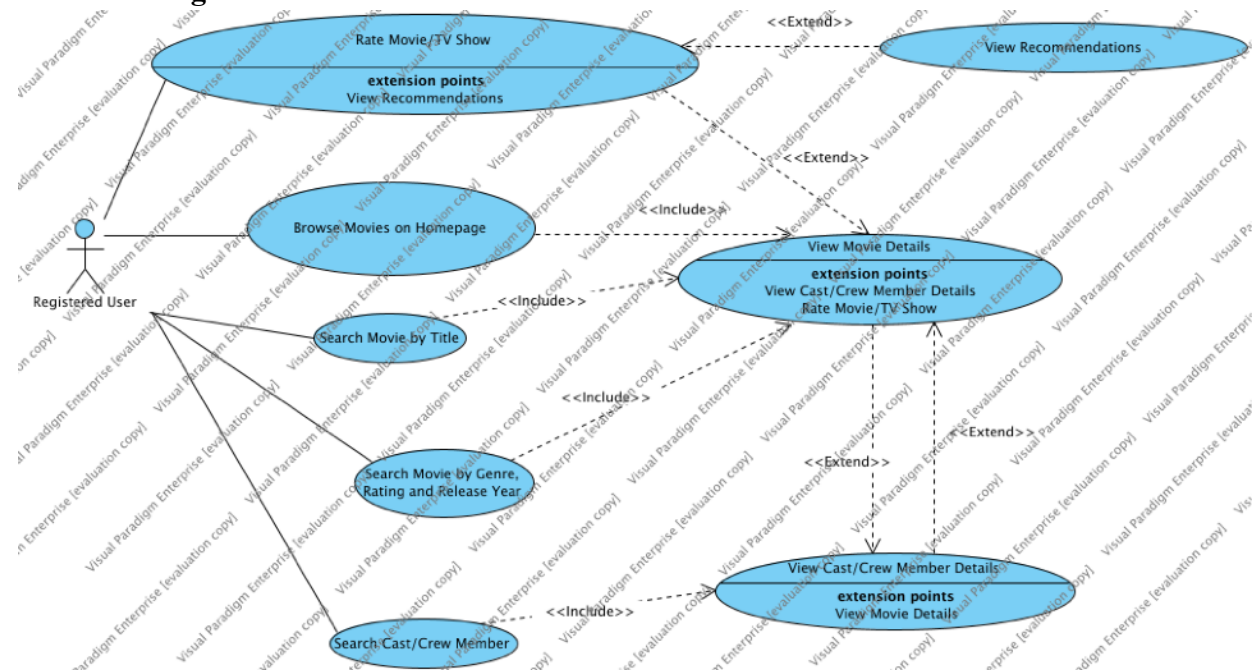
As Ntuaflx evolves, there are plans to integrate more sophisticated algorithms to enhance the personalization of content suggestions.

2. Advanced Search Capability: The platform's advanced search feature extends beyond typical title-based queries. Users can explore films and TV shows using a variety of criteria including genre, release date, and user ratings. Additionally, Ntuaflx offers a unique functionality where users can search for movies and TV shows by entering names of cast and crew members. This feature also uncovers the intricate network of connections between different movies, shows, and their associated personnel, providing a comprehensive view of the relationships and collaborations in the entertainment industry.

3. Comprehensive Movie and TV Metadata: Ntuaflx boasts an extensive database of metadata for each listed movie and TV show. This includes detailed synopses, full cast and crew information, production notes, award histories, and user-generated ratings. The platform serves as a rich repository of information, catering to both casual viewers and film enthusiasts who seek in-depth knowledge about their favorite content.

Use Cases

Use Case Diagram



This use case diagram represents the functionality of the Ntuaflx software system. It outlines the interactions that registered users can have with the system.

Use Case	Description
Browse Movies on Homepage	This use case allows users to browse through a selection of movies displayed on the homepage or a dedicated section. The movies are categorized and sorted based on different criteria like popularity, new releases, etc.
Rate Movie/TV Show	This use case enables registered users to rate movies or TV shows on a predefined scale (1 to 10 stars). After selecting a movie or TV show, users can provide their ratings, which are then aggregated and reflected in the overall rating of the title. This feature includes user authentication, rating submission, and updating the movie's average rating based on user input.
Search Cast/Crew Member	This use case allows users to search for cast and crew members by name. It enables users to find detailed information about the filmographies, biographies, and other relevant details of these individuals.
Search Movie by Genre, Rating and Release Year	This use case allows users to filter the movie database by genre categories such as Action, Drama, Comedy, etc., based on a minimum rating (on a 1-10 scale) and, optionally, a release year that defines a range of years. The system will offer an intuitive interface for users to select one genre. Once selected, the system will filter and display movies corresponding to the chosen genre, rating and starting year.
Search Movie by Title	This use case allows both registered and guest users to search for movies by typing in the full or partial movie title. The system then queries the movie database and returns a list of movies that match or closely resemble the search query. The functionality includes handling various forms of movie titles, managing partial inputs, and presenting the results in a user-friendly manner.
View Cast/Crew Member Details	In this use case, when a user selects a cast or crew member from a movie's details or after conducting a search, detailed profiles are displayed. These profiles include information such as their filmography, biography, notable works, and other relevant details. Additionally, each movie mentioned in their filmography is linked, allowing users to easily access and explore these movies directly from the individual's profile. This feature is designed to enhance user engagement by providing comprehensive insights into the careers of cast and crew members, along with convenient access to their associated movies.
View Movie Details	Upon selecting a movie (through search, browsing, or any other means), this use case provides detailed information about the movie, including the synopsis, cast and crew details, and production information. Additionally, it displays ratings and other relevant data. For each member of the cast and crew, there are links to their profiles, allowing users to explore more about these individuals. This comprehensive presentation of information aims to enrich the user's understanding and appreciation of the film.

View Recommendations

When a user rates a movie at 7 or higher, the app will present an option to view tailored recommendations. This feature generates lists of the top movies that are either in the same genre or feature the same leading actor or director. This optional feature is designed to align with the user's shown preferences, offering them a choice to explore similar movies within the app. This way, users have the flexibility to decide whether they want to see these recommendations.

Requirements

Requirement Diagram



This requirements diagram provides a structured visual representation of the system's requirements. It outlines the specific functionalities, constraints, and goals that the system is designed to fulfill.

Requirement	Description
Browse Movies on Homepage	When users access the web page, they should be presented with a browsable selection of movies. This could include new releases, popular titles etc.
Cast/Crew Career Overview	Provide an overview of a cast/crew member's career, including filmography, awards, and biography.

Cast/Crew Search Function	This function enables users to search for cast and crew members associated with movies and TV shows. The search should be robust enough to handle partial names and common variations or misspellings. Additionally, the system should present the results with essential information like name, known roles, and a link to a detailed profile containing their filmography, biography, and other relevant details.
Cross-Linking Movie and Cast/Crew Profiles	Establish links between movies and their cast/crew members. Clicking on a cast/crew member's name in a movie's details should redirect to their profile, and vice versa.
Data Accuracy and Integrity	Ensure the accuracy and integrity of all data presented, including movie details, ratings, and cast/crew information.
Detailed Movie Information Display	Once a movie is selected, the system displays detailed information including synopsis, cast/crew, ratings, genre, and production details.
Movie Genre Filter	This feature allows users to filter movies based on genre. The system must be updated with a comprehensive list of genres and ensure that each movie in the database is tagged appropriately to facilitate accurate filtering.
Movie Title Search Function	This function enables users to search the database for movies by entering titles. It should accommodate partial matches, case insensitivity, and common typos. The search algorithm must be optimized for quick, accurate results, minimizing the response time.
Personalized Recommendations	When a user rates a movie highly in a specific genre or featuring a particular actor, the system automatically suggests the top movies of that genre or with that actor or director. This feature requires analyzing user preferences and leveraging data on movies and ratings to generate relevant recommendations.
Scalability	The system architecture should be scalable to handle increasing amounts of data and user traffic efficiently.
Search Movie by Rating	Allows users to filter movies based on their ratings
Search Movie by Year	Users can search for movies by specifying the year of release. The function should support searches for a specific year and ranges (e.g., 2000-2005).
Security and Privacy	Implement robust security measures to protect user data and privacy, including secure data transmission and storage.
System Performance	The system should maintain optimal performance, with rapid loading times and efficient handling of multiple simultaneous user queries.
User Interface Design	The user interface should be intuitive and easy to navigate, with a clean design that enhances user experience.
User Rating Submission	Users can rate movies/TV shows. The system should update the average rating in real-time post-submission.
User Registration	This feature allows new users to create an account by providing necessary information like email, username, and password. The system should validate the input to ensure data integrity and uniqueness of the username/email.

User Requirements

Target User Demographics

Ntuaflix is designed to serve a broad audience with varying interests and needs related to film and television content. The system is tailored to the following user groups:

- 1. Cinema Enthusiasts:** Individuals with a deep interest in films, seeking comprehensive information and personalized recommendation engine to discover hidden gems and classics aligned with their preferences.
- 2. Casual Viewers:** Regular consumers of film and television content looking for a convenient way to discover new releases and receive recommendations tailored to their tastes.
- 3. Film Industry Professionals:** Creators, critics, and industry experts who require detailed and accurate metadata for professional purposes, including research, networking, and portfolio management.
- 4. Academic Users:** Scholars and students engaged in film studies who need access to a rich database for research, educational content, and academic projects.

User Environment

The system will be used in a variety of environments, each with its own set of requirements:

- 1. Home Environment:** Users accessing Ntuaflix from the comfort of their homes, likely through personal computers, tablets, or connected smart TVs, primarily for entertainment and information.
- 2. Educational Institutions:** Academics utilizing the system within educational settings, often on desktops or library terminals, requiring in-depth access to metadata for teaching and research.
- 3. Professional Settings:** Industry professionals using the platform in a work capacity, possibly in production offices or on-set, who may require faster, streamlined access to specific data points.

Actor	Description
Registered User	A Registered User is defined as an individual who has signed up for the system by providing necessary details such as name, email, and password. Once registered, the user creates a profile within the system and gains access to its features. To use certain functionalities, such as viewing detailed information about movies, selecting movies to see their details, and accessing profiles of cast and crew members, the user must be signed in to their account.

Technical Requirements

Performance

- 1. TR-001 Response Time:** The system shall process and display search results within 2 seconds for at least 95% of requests under normal load conditions.
- 2. TR-002 Data Processing:** The system shall handle batch processing of metadata updates for up to 30,000 records within 1 minute.

Scalability

- 1. TR-003 User Load:** The system shall support concurrent usage by up to 10,000 users without degradation of performance.
- 2. TR-004 Data Volume:** The system shall be capable of storing and managing metadata for over 1 million individual film and television titles.

Security

- 1. TR-005 Data Encryption:** All user data shall be encrypted.
- 2. TR-006 Authentication:** The system shall implement secure protocols for user authentication.
- 3. TR-007 Vulnerability Testing:** The system shall undergo quarterly security vulnerability testing and necessary patches applied within 1 month of identification.

Maintainability

- 1. TR-008 Code Documentation:** The system's source code shall be fully documented following the documentation standards of the IEEE.
- 2. TR-009 Update Deployments:** The system shall support updates and maintenance without requiring more than 1 hour of downtime per month.

Usability

- 1. TR-010 User Interface:** The system shall provide a user-friendly interface.

Localization Support

- 1. TR-011 Multilingual Interface:** The system shall provide an interface in at least two languages: English, Greek.

Auditing and Logging

- 1. TR-012 Audit Trails:** The system shall maintain audit logs of all user activities that affect data changes for a minimum of 2 years.
- 2. TR-013 Log Analysis:** The system shall provide tools for the analysis of log data to detect unusual patterns that could indicate security breaches.

Availability

- 1. TR-014 System Uptime:** The system shall achieve 99.9% uptime, excluding scheduled maintenance windows.
- 2. TR-015 Disaster Recovery:** The system shall have a disaster recovery plan that allows restoration of service within 4 hours in the event of a major outage.

Hardware Requirements

Network

HR-001 Bandwidth and Throughput: The server hosting Ntuaflux shall be connected to a high-speed internet connection to ensure smooth data transfers.

HR-002 Latency: The network infrastructure shall support a maximum latency of 100 ms for users connecting to the service within the same region as the data center.

Client Computers

HR-003 Operating System Compatibility: Ntuaflux shall be compatible with the following operating systems: Windows 10 and above, macOS X 10.15 and above, and popular Linux distributions such as Ubuntu 20.04 and above.

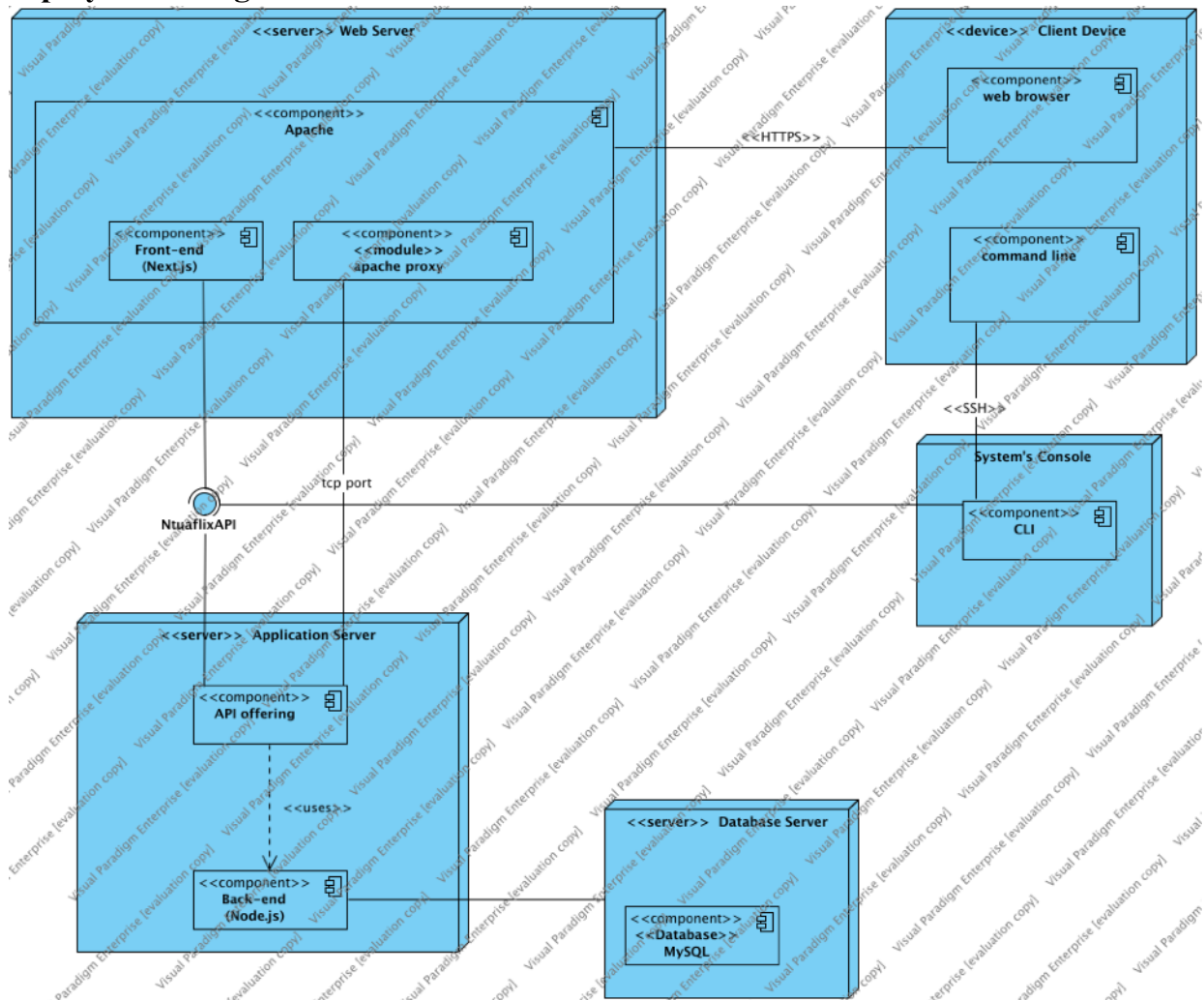
HR-004 Browser Support: The client-side shall be accessible through the latest versions of major web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

HR-005 Processor Requirements: Users' computers shall have at least an Intel i5 processor or equivalent to ensure responsive interactions with the Ntuaflux platform.

HR-006 Memory Requirements: A minimum of 8GB RAM is recommended for optimal performance when accessing Ntuaflux.

Deployment Requirements

Deployment Diagram



This diagram illustrates a multi-tier architecture that separates the presentation layer (web server), application logic layer (application server), and data storage layer (database server), which is a common structure for scalable web applications. The system is designed to facilitate secure and efficient communication between the user's interface and the servers providing the application's functionality.

Nodes

Name	Description
<<device>> Client Device	The client layer is represented by the user's device, which primarily uses a web browser to interact with the system. The web browser is responsible for sending requests to the web server and rendering the content received in response. Users may also interact with the system through a command-line interface, which, communicates directly with the application server's API. This direct communication is critical for administrative tasks that require more control and precision than what is typically provided through the web interface. CLI also provides an alternative way of accessing the system's functions.
<<server>> Application Server	The application server layer contains the back-end logic of the system and is where the API offering resides. It is the nucleus of the application's functionality, processing API requests, executing the necessary business logic, and performing data operations as required. The back-end component is designed to handle complex processes, including data validation, user authentication, and interactions with the database server.
<<server>> Database Server	Data storage and management are handled by a separate database server running MySQL. This server is dedicated to the secure and efficient storage of application data, and it interacts with the application server's back-end component. SQL queries are generated by the back-end and sent to the MySQL server, which processes these queries and returns the required data.
<<server>> Web Server	On the server side, the Apache web server acts as the gateway for all HTTP requests from the client's web browser. It is configured to serve static content directly to the client, which includes files such as HTML, CSS, and client-side JavaScript. This content constitutes the front-end of the application, which users interact with via their web browser. In addition to handling static content, the Apache server is equipped with a proxy module that forwards dynamic content requests to the application server. This module ensures that the web server can delegate tasks that require server-side processing, thereby acting as a reverse proxy.
System's Console	The node system's console is the command-line environment where system administrators and authorized users access the backend of the Ntuaflix software system. It provides a powerful CLI component that offers direct interaction with the system's databases and services without the need for a graphical user interface. The CLI is also integral for performing various tasks that include database maintenance, data entry, batch updates and system diagnostics.

Components

Name	Description
<<Database>> MySQL	<p>This database, central to the system, stores comprehensive data about media titles, industry personnel, and user interactions. It consists of multiple interconnected tables, each serving a specific purpose:</p> <ul style="list-style-type: none">• Titles Table: Core table containing details about each media title (movies, TV shows, episodes), including their genres, release years, and ratings.• Personnel Tables (Names_, Directors, Writers): Store information about individuals in the media industry, like actors, directors, and writers, along with their roles in various titles.• Relationship Tables (Principals, Known_for, Had_role): Define the relationships between media titles and industry personnel, outlining who played what role in each title.• User Interaction Tables (User_Title_Ratings, Users): Manage user data, including profiles and their ratings for different titles.• Support Tables (Title_genres, Aliases, Episode_belongs_to): Provide supplementary information like genres for each title, alternate titles, and episode-to-show mappings. <p>The database supports complex queries, enabling detailed data retrieval for various application features, including user recommendations, title searches, and data analytics.</p>
<<module>> apache proxy	<p>Apache proxy serves as a gateway between users and the web service, handling incoming requests and directing them to the appropriate backend processes. It can provide additional features such as SSL encryption, load balancing, and access control, enhancing security and efficiency.</p>
Apache	<p>Apache is a pivotal module that incorporates the apache proxy and the front-end interface of the application. The front-end component, typically accessed through a web browser, is where users interact with the application. It is built using Next.js, a React framework that enables server-side rendering and static site generation to optimize performance and user experience. The front end is responsible for presenting information to the user in a visually coherent manner, handling user inputs, and managing the state of the user interface. Together, the Apache proxy and the front-end form the user-facing side of the application, bridging the gap between the client's requests and the server's responses.</p>

API offering	The API Offering of our application is a cohesive framework that facilitates client-server interaction via HTTP requests and responses. It includes a set of defined routes (endpoints), each designed to perform specific operations. The server configuration ensures these routes are accessible over the network, while the application setup ties together routing logic, middleware, and backend services to form a functional and scalable web service.
Back-end (Node.js)	The backbone of server-side logic, this component is responsible for initializing the server, configuring middleware, establishing database connections, and defining the routes that map to various controller functions. It uses frameworks like Express.js to streamline the creation of server routes and handle HTTP requests and responses.
CLI	The CLI serves as a command-line interface that allows administrators and advanced users to interact with the backend system. It provides a suite of commands enabling various operations like user management, data retrieval, and content manipulation. The CLI is designed for efficient, scriptable, and direct interaction with the system's core functionalities without the need for a graphical interface.
command line	The command line offers a text-based user interface that enables administrators and users to execute commands and interact with the application server remotely via Secure Shell (SSH). This access method is crucial for performing administrative tasks, managing application processes, and executing maintenance operations from any location with an internet connection. It allows for a secure, encrypted connection to the server's CLI, providing a powerful and direct method for controlling the core functionalities of the application without a graphical user interface.
Front-end (Next.js)	This is the main container representing the overall Next.js application. It encapsulates all the other components related to the frontend of the application.
web browser	The web browser is the client application used to access and interact with the front end of the web service. It renders the user interface, processes user inputs, and communicates with the server via the Apache proxy to request and display web pages.