**Product Document: Streamflix Analytics Dashboard**

**Version:** 2.0  
**Author:** Group 2  
**Date:** 28 May 2025

**1. Introduction**

Streamflix, an Italian company with global expansion ambitions, is transitioning from a free, ad-supported platform to a subscription-based service.

The **Streamflix Analytics Dashboard** is designed to provide actionable insights into user engagement, content performance, and subscription trends. By leveraging Power BI, this tool enables **data-driven decision-making** for content optimization and user retention.

**2. Purpose**

This dashboard empowers stakeholders to:

* Analyze user behavior and preferences
* Evaluate content performance based on key metrics
* Forecast subscription trends
* Optimize **genre-based filtering and dynamic slicers** for storytelling insights

**3. Features**

* **Interactive Dashboards:** Customizable visuals for deep analysis
* **Genre-Based Filters:** Dynamic slicers for better storytelling
* **Regional Analysis:** Understanding viewer preferences by location
* **Subscription Performance Tracking:** Metrics on user retention and churn rates

**4. Target Audience**

* Data analysts and business intelligence teams
* Marketing teams focusing on user engagement
* Content creators analyzing genre popularity
* Executives tracking revenue growth

**5. Technology**

* **Power BI** (for visualization and analysis)
* **SQL** (for data preparation and optimization)
* **Python (Seaborn, Matplotlib)** (for advanced analytics and insights)

**6. Data Model & Schema**

Below is the relational **schema** used in the Streamflix Analytics Dashboard:

**Database Schema Overview**

The Streamflix database is designed to manage a movie streaming service, capturing information about movies, users, and their interactions via ratings. The schema supports key functionalities such as tracking viewership, user behavior, and movie performance metrics.

**Movies Table**

Stores metadata for each movie in the catalog.

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| MovieID | INT | Unique identifier for each movie |
| Title | VARCHAR(500) | Movie title |
| Genres | VARCHAR(200) | Comma-separated genres (e.g., Action, Drama) |
| Language | VARCHAR(20) | Primary language of the movie |
| Country | VARCHAR(10) | Country of origin |
| TotalViews | INT | Number of views across all users |

Rationale: Designed to be scalable and support analysis of movie trends, genre preferences, and regional performance.

**Users Table**

Captures demographic and behavioral data for each user.

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| UserID | INT | Unique identifier for each user |
| Age | INT | User's age |
| Gender | CHAR(1) | Gender (M, F, O) |
| Country | VARCHAR(10) | Country of residence |
| SubscriptionStatus | VARCHAR(20) | e.g., Free, Premium, Cancelled |
| TotalWatchTime | INT | Total time spent watching (in minutes) |
| Device | VARCHAR(50) | Device used to access content |

Rationale: Provides a profile for user segmentation and targeted recommendations.

**Ratings Table**

Records individual movie ratings provided by users.

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| RatingID | INT | Unique identifier for each rating entry |
| UserID | INT | Foreign key referencing Users(UserID) |
| MovieID | INT | Foreign key referencing Movies(MovieID) |
| Rating | DECIMAL(2,1) | Rating value (e.g., 4.1, 3.5) |
| Timestamp | DATETIME | When the rating was submitted |

Rationale: Supports user feedback analysis, average rating calculation, and movie ranking features.

**Relationships & Constraints**

- Ratings.UserID → Users.UserID  
- Ratings.MovieID → Movies.MovieID  
  
All primary and foreign keys are strongly typed and indexed for optimal join performance.

**7. Design Decisions & Rationale**

1. Normalization: The schema is normalized to 3rd Normal Form (3NF) to reduce redundancy.  
2. Data Types: VARCHAR used for text flexibility; DECIMAL for precise ratings; DATETIME for timestamping.  
3. Scalability: Genres stored as strings initially for simplicity, but normalization is possible later.  
4. Performance: Proper indexing ensures fast JOIN operations.

**8. Relationships**

- Ratings.UserID → Users.UserID  
- Ratings.MovieID → Movies.MovieID  
  
All primary and foreign keys are strongly typed and indexed for optimal join performance.

**9. Technical Design Decisions**

* **Database Management & Query Optimization**
* **Power BI Implementation**
* **DAX Measures** for aggregations (e.g., total revenue, watch time trends).
* **Conditional Formatting & Sorting** for intuitive data visualization.
* **Genre-based filtering** for storytelling insights.
* **Add WatchHistory table to track viewing per movie per user.**
* **Normalize Genres into a separate table for better querying.**
* **Track session data for behavioral analytics.**

**10. Conclusion**

The **Streamflix Analytics Dashboard** streamlines data exploration and decision-making. Its **intuitive design and insights** ensure that users can **unlock actionable intelligence** for business growth.