

ScalaStream

Web Systems & Machine Learning Competition

Ingenium — IIT Indore

Competition Overview

Modern video streaming platforms operate at massive scale and require carefully engineered system architectures spanning data ingestion, processing, storage, and content delivery.

This competition challenges participants to design and implement a highly scalable and highly available video streaming platform. The focus is on distributed systems design, scalability trade-offs, cost-efficient architecture decisions, and the integration of machine learning for personalized recommendations.

Problem Statement

Participants must design and implement a video streaming platform that allows users to:

- Upload video content
- Process and transcode videos into streamable formats
- Store and stream videos with low latency
- Interact with content through metadata such as likes and comments
- Receive personalized video recommendations using machine learning

The system must handle concurrent uploads, high user traffic, and remain operational under failure conditions while optimizing storage and network costs.

Objectives

- Support concurrent video uploads from multiple users
- Convert uploaded videos into streamable formats
- Deliver video content with minimal latency
- Handle high traffic without system failure
- Maintain a clean and intuitive user experience
- Recommend videos based on user behavior and interests

Functional Requirements

- User authentication and authorization
- Video upload functionality
- Video transcoding and storage
- Video streaming playback
- Metadata management (likes, comments, views)
- Machine learning-based recommendation system for user feeds

Non-Functional Requirements

- Low-latency video delivery
- High availability and fault tolerance
- Cost-efficient storage and network utilization
- Scalable system architecture
- Graceful failure handling
- Clear justification of architectural trade-offs

AI / Machine Learning Requirements

- The system must include a machine learning-based recommendation mechanism
- Recommendations must be derived from user interaction data such as:
 - Watch history
 - Likes and engagement patterns
- Rule-based or static recommendation logic is not sufficient

Participants must clearly explain:

- Why machine learning is used
- What data is utilized
- How recommendations enhance user experience

Scope & Assumptions

- Live streaming is not required
- Deployment at real-world production scale is not mandatory
- Mock or synthetic data may be used
- The architecture must conceptually support horizontal scalability

Deliverables

Participants must submit:

- A working web application demonstrating:
 - Video upload and playback
 - Metadata handling
- System architecture diagram
- Explanation of scalability and cost optimization strategies
- Explanation of the machine learning recommendation approach
- Source code repository with a README

Optional Features

- Adaptive video quality selection
- Playback speed control
- Enhanced recommendation filters

Rules are subject to change at the discretion of the organisers.
Further instructions will be communicated to registered participants.