## **SPAM**

Next-Generation Smart Photo Album

Shengding Liu, Ziran Gao, Zhanwei Zhang, Yixian Zhou, Shuwei Zhang

Southern University of Science and Technology

March 16, 2025



### Table of contents

- Requirement Analysis
- Basic Features
- Non-Functional Features
- Data Requirements
- Technical Requirements
- Task & Plan
- Task Decomposition
- Plan

1.1. Basic Features

#### **Basic Features**

#### • Effortless Photo Management

Organize your photos seamlessly by time, location, and custom tags. Easily browse and retrieve memories with structured categorization.

- Advanced Search Capabilities
   Instantly find your photos based on content, date, location, or tags. Use AI-powered search to recognize faces, objects, and scenes within your photos.
- Built-in Photo Editing Tools
   Enhance your photos with powerful editing tools, including cropping, filters, adjustments, and retouching options—all within the app.



### Basic Features (Cont.)

- Memory Timeline & Video Generation
   Relive special moments with automatically generated memory timelines.
   The system can create video highlights using your best photos and clips.
- Seamless Social Sharing & Collaboration
  Share your favorite moments effortlessly with family and friends.
  Collaborate on shared albums, comment on photos, and react with emojis.



1.2. Non-Functional Features

### Non-Functional Features

### Usability

The application offers an intuitive, multilingual, and accessible interface with responsive design for seamless use across devices.

#### Performance

Ensures fast response times, scalability, and efficient caching for handling large albums and high user concurrency.

### Security

Implements robust authentication, role-based access control, and encryption to protect user data and ensure secure access.



## Non-Functional Features (Cont.)

- Safety & Privacy
   Complies with data protection regulations, provides data deletion options, and ensures secure handling of sensitive information.
  - Maintainability
    Features modular code, comprehensive documentation, automated testing, and CI/CD pipelines for easy maintenance and updates.



## 1.3. Data Requirements

## Data to be Collected and Managed

**Photo files:** Formats: JPG, PNG, HEIC, etc.

**Metadata:** EXIF data: Timestamp, Geolocation, Camera/device info

**User data:** Tags: events, people, places; Albums: names, descriptions; Sharing: public/private/collaborative

Al-generated tags: Facial recognition, Object/scene detection

**Edited content:** Modified images: crops, filters; Generated videos: moments, timelines



### How to Collect and Process Data

- Image Metadata Extraction
  Automatically extract EXIF data using libraries (e.g., ExifTool, Python Pillow).
- Al Tagging & Classification
   Use OpenCV and GPT (or other AI/ML services) for face recognition, object detection, and tagging.
- Photo Editing Metadata
   Store edit history and versioning to allow rollback or undo changes.
- Search Indexes
   Build search indexes in databases (e.g., ElasticSearch) to enable fast retrieval based on time, location, people, objects, and tags.



## Data Storage

- Use MINIO for object storage (photos, videos).
- Use MySQL/PostgreSQL for relational data (user accounts, tags, album relationships).
- Backup and disaster recovery plans to protect user data.



1.4. Technical Requirements

# Operating Environment

Category	Technologies/Tools
Frontend	React.js, TailwindCSS/Bootstrap
Backend	Node.js (Express/Koa), Python (FastAPI/Django)
Database	MySQL/PostgreSQL, MINIO
AI/ML Services	OpenCV, GPT, Third-party APIs (Remove.bg, DeepAI)
Video Generation	MoviePy, FFmpeg
Authentication	OAuth 2.0, JWT, HTTPS
Deployment	Docker, Kubernetes, Ali Cloud/On-premise



## 2. Task & Plan

## 2. Task & Plan

## 2.1. Task Decomposition

### **User Stories**

### Photo Management

- Manage photos by time, location, event, or tags.
- Automatically create albums based on photo characteristics.

### Photo Editing \* It

- Crop, rotate, and adjust brightness/contrast.
- Remove unwanted objects to improve photo quality.

#### Advanced Search Q

- Search by date, location, tags.
- ► Search by image content (face/object recognition).

#### Moment Video & Timeline ⊞

- ► Automatically generate highlight videos with music.
- View photos in chronological order.

### Sharing & Social <</li>

- ► Share photos/albums on social media or with friends.
- Create collaborative albums for team/family.



#### Tasks

## Photo Management

- Design database tables (photos, albums, tags).
- ► Implement photo upload & storage.
- ► Categorize albums by time & location.

#### Advanced Search Q

- ► Implement search by time & location.
- ► Research image search API.
- ► Design search UI/UX.

### • Photo Editing 🗷

- ► Add crop & rotate features.
- ► Investigate image processing libraries (OpenCV, Pillow).

### ● Sharing & Social **<**

- Design access control for shared albums.
- ► Implement social sharing links.



## 2. Task & Plan

### 2.2. Plan

## Task Assignment

#### Frontend

- ► Zhang Zhanwei: Frontend pages for photo/video and memory timeline generation.
- ► Zhou Yixian: Frontend pages for image editing.
- ► Zhang Shuwei: Frontend pages for image management (create, read, update, delete).

#### Backend

- ► Gao Ziran: Image storage, authentication, user login (frontend & backend), photo editing tools.
- ► Liu Shengding: Tagging, advanced search, photo/video and memory timeline generation (backend).



## **Project Timeline**

Week 5: Requirements Analysis, Database Design, Basic UI Design

**Week 6**: Photo upload function, Search function, Basic photo editing function

Week 7: Collaborative Albums & Social Sharing, Testing & Debugging

Week 8: Sprint 1 is over, Ready for presentation



## Thank You!