Survey Platform

Functional Requirements

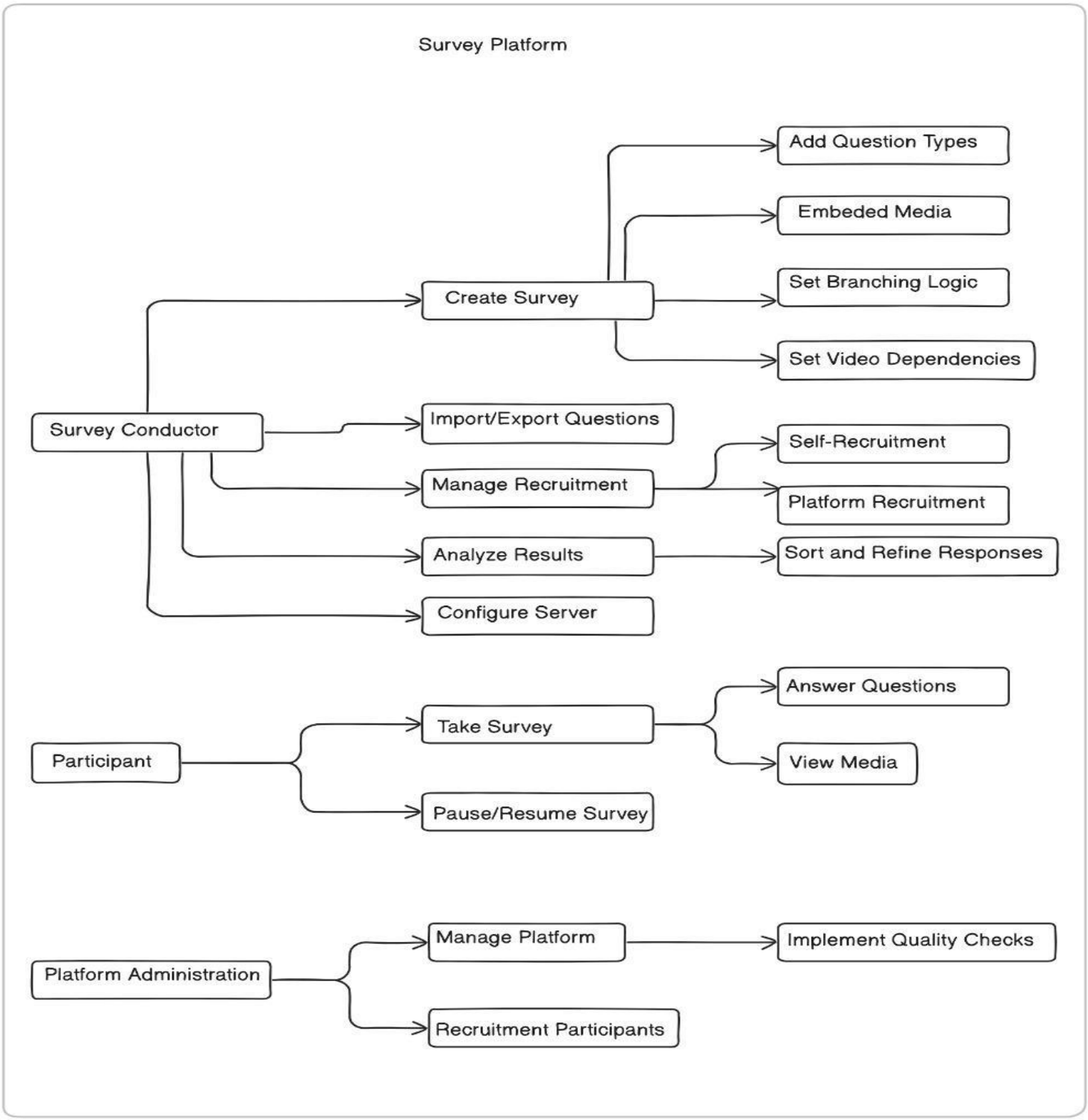
1. User Management
    - 1.1 User registration and authentication
    - 1.2 User roles (survey conductor, participant, administrator)
    - 1.3 User profile management
2. Survey Creation and Management
    - 2.1 Create, edit, and delete surveys
    - 2.2 Import/export survey questions (CSV format)
    - 2.3 Support various question types (MCQ, MSQ, rating scales, open-ended text, image ranking,coding)
    - 2.4 Embed media (images, videos, audio) in surveys
    - 2.5 Implement branching logic based on question responses
    - 2.6 Set video-dependent questions (locked until video playback criteria met)
3. Participant Recruitment
    - 3.1 Self-recruitment mode (generate shareable survey links)
    - 3.2 Platform recruitment service
        - 3.2.1 Define participant criteria
        - 3.2.2 Match surveys with registered participants based on criteria
4. Survey Taking
    - 4.1 Access surveys via web and mobile interfaces
    - 4.2 Pause and resume survey progress
    - 4.3 Save responses in real-time
    - 4.4 Support offline survey taking (sync when online)
5. Data Collection and Storage
    - 5.1 Collect and store individual responses
    - 5.2 Ensure response anonymity
    - 5.3 Option for self-hosted or cloud-based data storage
6. Analytics and Reporting
    - 6.1 Generate basic survey result reports
    - 6.2 Provide tools to sort and refine responses
    - 6.3 Export results in various formats (CSV, PDF, etc.)
7. Administration
    - 7.1 Platform configuration and management
    - 7.2 User management (create, modify, delete accounts)
    - 7.3 Monitor system usage and performance

Non-Functional Requirements

1. Performance
    - 1.1 Support concurrent users: minimum 10,000 simultaneous users
    - 1.2 Response time: <2 seconds for 95% of requests
    - 1.3 Page load time: <3 seconds for complex surveys
2. Scalability
    - 2.1 Support for microservices architecture
3. Security
    - 3.1 End-to-end encryption for all data transmissions
    - 3.2 Regular security audits and penetration testing
    - 3.3 Multi-factor authentication option
4. Usability
    - 4.1 Intuitive, user-friendly interface for both conductors and participants
    - 4.2 Mobile-responsive design
5. Maintainability
    - 5.1 Modular architecture for easy updates and maintenance
    - 5.2 Comprehensive logging and monitoring
    - 5.3 Automated testing with >80% code coverage
6. Compatibility
    - 6.1 Support latest versions of major browsers (Chrome, Firefox, Safari, Edge)
    - 6.2 Mobile app support for iOS 13+ and Android 8+
7. Localization
    - 7.1 Multi-language support (initially English, with easy addition of other languages)
    - 7.2 Localization of date, time, and number formats
8. Data Integrity
    - 8.1 Implement data validation checks
9. Compliance
    - 9.1 Adhere to relevant survey and market research industry standards
    - 9.2 Implement ethical guidelines for survey conduct and data usage
10. Interoperability
    - 12.1 Provide RESTful APIs for integration with third-party systems
    - 12.2 Support standard data exchange formats (JSON, XML)

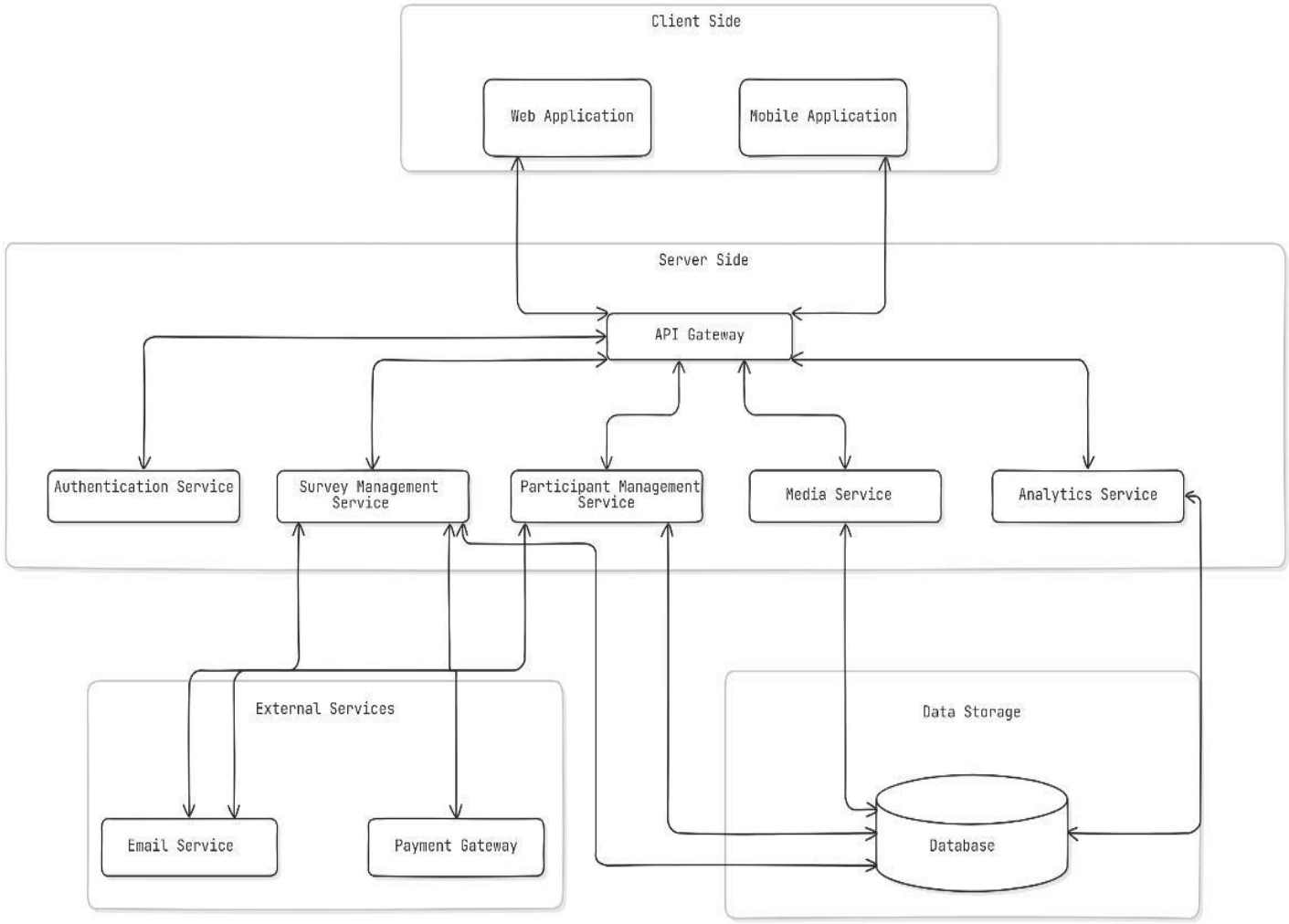
Use Case Diagram

Link: https://app.eraser.io/workspace/ap1aYFlmiXNvKbAXcMVv


Survey Platform
Add Question Types
Embeded Media
Set Branching Logic
Create Survey
Set Video Dependencies
Survey Conductor
Import/Export Questions
Self-Recruitment
Manage Recruitment
Platform Recruitment
Analyze Results
Sort and Refine Responses
Configure Server
Answer Questions
Take Survey
Participant
View Media
Pause/Resume Survey
Manage Platform
Implement Quality Checks
Platform Administration
Recruitment Participants

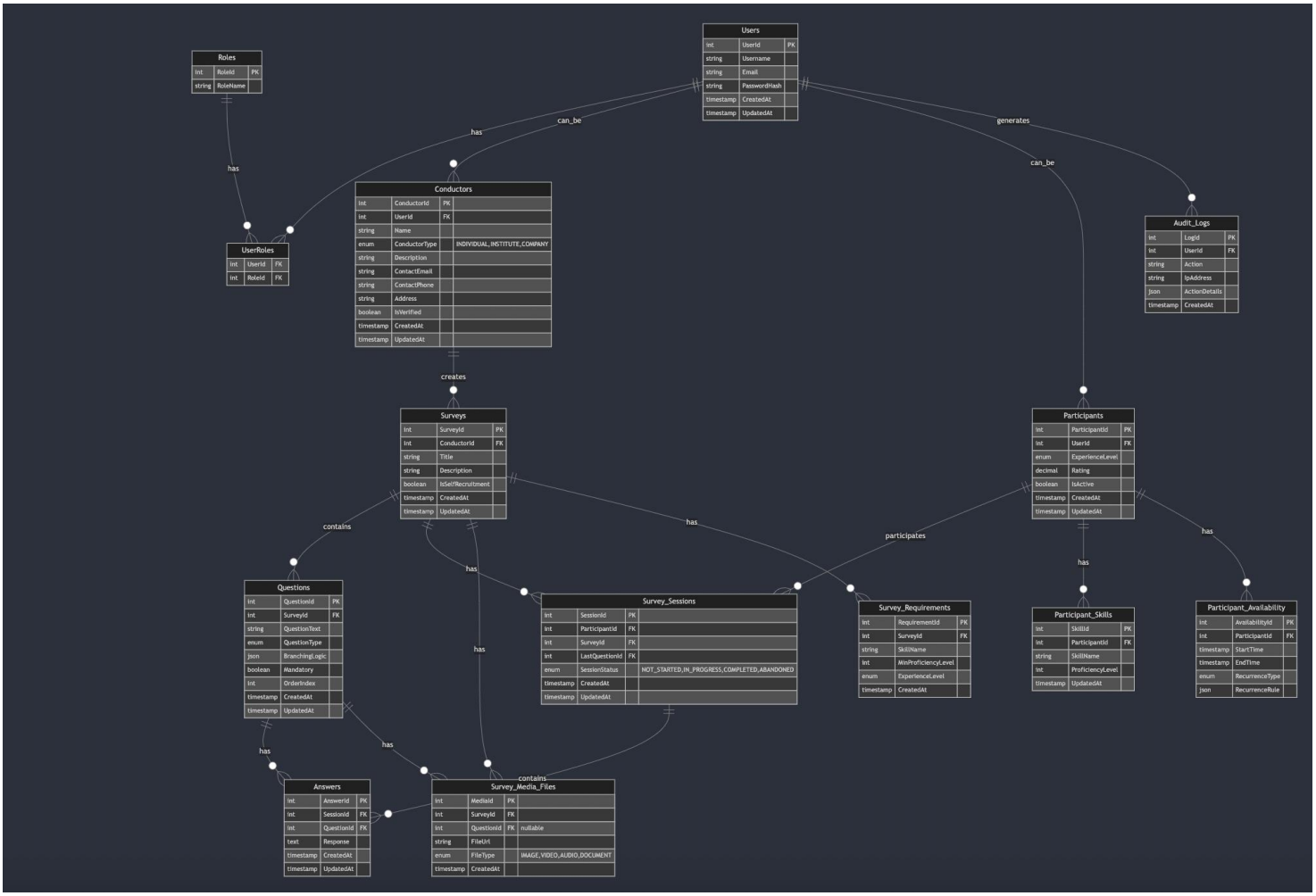
High Level Design:

Link:https://app.eraser.io/workspace/qb3PHL2DqKzO8gRHAS7O



Database Schema:

Link: https://app.eraser.io/workspace/aagyHzT5gYtCyaKQmICf


Roles
int RoleId PK
string RoleName
Users
int UserId PK
string Username
string Email
string PasswordHash
timestamp CreatedAt
timestamp UpdatedAt
has
can_be
generates
has
can_be
UserRoles
int UserId FK
int RoleId FK
Conductors
int ConductorId PK
int UserId FK
string Name
enum ConductorType INDIVIDUAL,INSTITUTE,COMPANY
string Description
string ContactEmail
string ContactPhone
string Address
boolean IsVerified
timestamp CreatedAt
timestamp UpdatedAt
Audit_Logs
int LogId PK
int UserId FK
string Action
string IpAddress
json ActionDetails
timestamp CreatedAt
creates
Surveys
int SurveyId PK
int ConductorId FK
string Title
string Description
boolean IsSelfRecruitment
timestamp CreatedAt
timestamp UpdatedAt
Participants
int ParticipantId PK
int UserId FK
enum ExperienceLevel
decimal Rating
boolean IsActive
timestamp CreatedAt
timestamp UpdatedAt
contains
has
has
participates
has
has
Questions
int QuestionId PK
int SurveyId FK
string QuestionText
enum QuestionType
json BranchingLogic
boolean Mandatory
int OrderIndex
timestamp CreatedAt
timestamp UpdatedAt
Survey_Sessions
int SessionId PK
int ParticipantId FK
int SurveyId FK
int LastQuestionId FK
enum SessionStatus NOT_STARTED,IN_PROGRESS,COMPLETED,ABANDONED
timestamp CreatedAt
timestamp UpdatedAt
Survey_Requirements
int RequirementId PK
int SurveyId FK
string SkillName
int MinProficiencyLevel
enum ExperienceLevel
timestamp CreatedAt
Participant_Skills
int SkillId PK
int ParticipantId FK
string SkillName
int ProficiencyLevel
timestamp UpdatedAt
Participant_Availability
int AvailabilityId PK
int ParticipantId FK
timestamp StartTime
timestamp EndTime
enum RecurrenceType
json RecurrenceRule
has
has
has
contains
Answers
int AnswerId PK
int SessionId FK
int QuestionId FK
text Response
timestamp CreatedAt
timestamp UpdatedAt
Survey_Media_Files
int MediaId PK
int SurveyId FK
int QuestionId FK nullable
string FileUrl
enum FileType IMAGE,VIDEO,AUDIO,DOCUMENT
timestamp CreatedAt