CS319 Course Project
System Modeling Package

Deliverable 1

Team Members:

Ali Şen / 22203368

Barış Peksak / 22201659

Berkay Şimşek / 22303338

Gökay Nuray / 22302913

Muhammet Furkan Demir / 22302644

Contents

1 Project Overview and Scope

The system enables researchers to run human-subject studies comparing software engineering artifacts (source code, tests, UML, requirements, etc.), with blinded or non-blinded evaluation, annotation, ratings, and dashboards. We expand the instructor's summary use case into Level-1 diagrams, choose a justified tech stack, and provide INVEST-compliant user stories with acceptance criteria.

Core capabilities: artifact management, participant competency assessment, artifact comparison (2 or 3-way) with annotation, study orchestration, dashboards and data collection.

2 Level-1 Use Case Diagrams

2.1 (Shared) Manage Users and Roles

2.1.1 Actors

Researcher: Registers, signs in, updates profile; needs proper role to access study features.

Participant: Registers, signs in, updates profile; needs Participant role to access tasks.

Admin: Assigns/removes roles, deactivates/reactivates accounts, oversees user lifecycle and audits.

External Email Service: Sends verification emails and password-reset links.

2.1.2 Sub-Use Cases (Scenarios)

Register Account: User signs up with email/password; must verify email before full access.

Sign In: User authenticates; optional 2FA challenge if enabled.

Reset Password: User requests a reset link via email and sets a new password.

Update Profile: User edits display name, contact info, and notification preferences.

Manage Roles (Admin): Admin assigns/changes/revokes roles; all changes are audited.

Deactivate/Delete Account (Optional): Account is deactivated or deleted per policy (admin-gated).

View Audit History (Optional): Admin reviews who changed which roles and when.

2.1.3 Relationships and System Boundary

System Boundary: A rectangle labeled **"Manage Users and Roles"** encloses all sub-use cases above. Actors (Researcher, Participant, Admin, External Email Service, 2FA Provider) sit outside and interact via the system's interfaces.

Relationships: Register Account includes email validation and verification. **Sign In** and **Reset Password** may extend to a **2FA Challenge**. **Manage Roles** includes auditing every change. **Deactivate/Delete Account** extends **Manage Roles** as an admin-only lifecycle action.

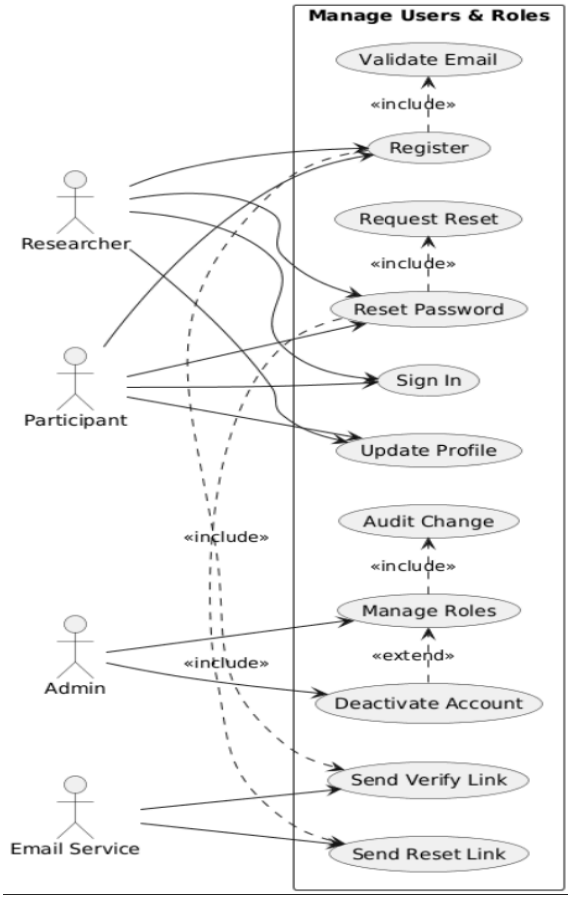

Figure 2.1: Manage Users and Roles

2.2 Upload and Organize Artifacts

2.2.1 Actors

Researcher (Primary): Uploads, manages, and organizes artifacts.

Admin (Supporting): Defines allowed file types, size limits, and storage quotas.

External Storage Service: Handles the physical storage of uploaded files.

2.2.2 Sub-Use Cases (Scenarios)

Upload Artifact: The Researcher navigates to the upload page, sees limits/accepted types, chooses a file, and if checks the pass, uploads it; metadata is captured, a version number is assigned (v1 if first), and the upload completes. (Entry: authenticated Researcher. Exit: artifact stored and indexed or rejection shown.)

File Type Validation: The system verifies MIME/extension against the allowed list defined by Admin and rejects unsafe types before completing the upload. **Allowed (examples):** .pdf, .png, .jpg, .jpeg, .docx, .csv, .xlsx, .txt, .md, .json, .py, .java, .cpp, .ipynb. **Rejected (examples):** .exe, .dll, .bat, .cmd, .sh, .apk, .msi, .iso, .vbs, .jar. (Entry: file selected by an authenticated Researcher. Exit: accepted for processing or rejected.)

Extract Metadata: The system records file name, size, hash, type, and upload time for indexing and auditing. (Entry: validation passed. Exit: metadata stored in the database.)

Store Artifact: After validation, the file is written to permanent storage via the external storage service, and a storage URL is assigned; success or error is shown to the user. (Entry: metadata extracted and file verified. Exit: file stored with a stable URL.)

Tag and Categorize: The Researcher adds tags, folders, and descriptive categories to organize the artifact for search and filtering. (Entry: artifact stored. Exit: metadata and search index updated.)

Version Artifact: First upload creates **v1**; duplicate uploads prompt the user to **create a new version** or **replace the existing** file, preserving version history. (Exit: version history maintained.)

Generate Preview and Index: The system produces a text/code preview or thumbnail (when applicable) and updates the search index to speed up browsing and comparisons. (Entry: file stored. Exit: preview available and searchable.)

2.2.3 Relationships and System Boundary

System Boundary: Artifact Comparator Web Application, the web-based interface and backend services responsible for artifact management, validation, and organization.

Relationships: Upload Artifact includes **File Type Validation**, **Quota/Size Check**, **Extract Metadata**, and **Store Artifact**; **Store Artifact** includes **Generate Preview & Index** (when previewable). **Upload Artifact** extends **Tag & Categorize** (optional post-upload organization). On duplicates, **Duplicate Detection** extends **Version Artifact**, where the Researcher chooses **Create New Version** or **Replace Existing**. If limits are hit, **Quota/Size Check** extends **Quota Exceeded**. **File Type Validation** enforces the **Admin-configured allowed file types** list.

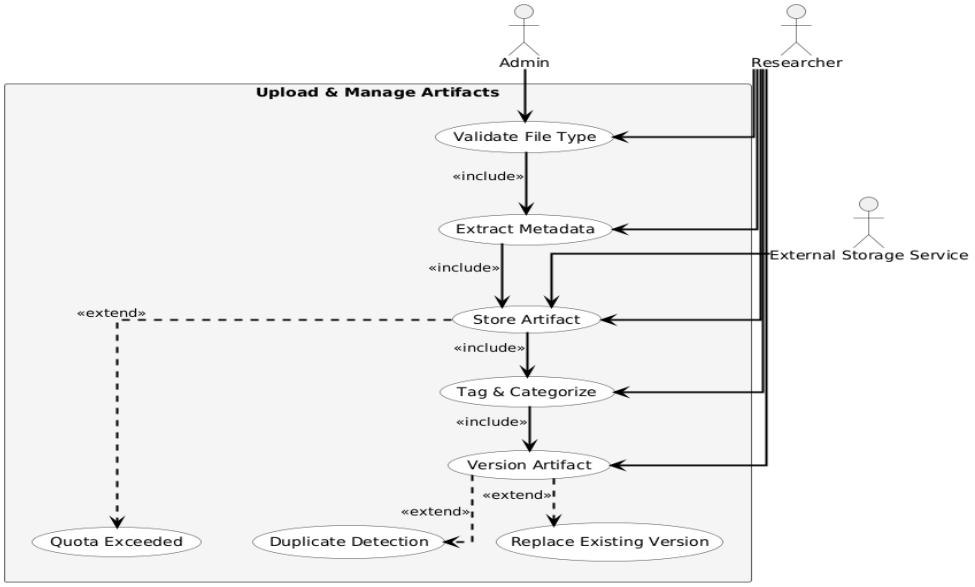

Figure 2.2: Upload and Organize Artifacts

2.3 Create and Manage Studies

2.3.1 Actors

Researcher: The primary actor who creates studies, sets blinded/compare-mode settings, defines tasks and criteria, invites/enrolls participants, tracks progress, and exports results.

Admin: Supports policy/limits (e.g., quotas, data retention), manages email templates/domains if needed, and can archive or delete studies.

Participant (Indirect): Receives invites and assignments created by the Researcher (does not configure studies).

External Email Service: Sends invitation links and reminders triggered by the system on the Researcher's request.

2.3.2 Sub-Use Cases (Scenarios)

Create Study: The Researcher creates a new study with title, goals, blinded mode on/off, and comparison mode (2-way or 3-way).

Configure Tasks: The Researcher selects artifacts for each task, writes instructions, and sets rating criteria (e.g., readability, correctness, completeness) with optional weights.

Randomize Order (Blinded): If blinded mode is enabled, the system randomizes artifact order and hides origin/author info for each task.

Set Deadlines and Reminders: The Researcher sets due dates and optional automated reminders (e.g., 24h before due).

Manage Enrollment: The Researcher invites participants (email or shareable token), enrolls them, and assigns tasks (optionally by competency tier).

Send Email Invites: The system sends invitation emails with secure tokens; expired/used tokens are rejected.

Monitor Progress: The Researcher views completion percentages, per-task status, and basic QC flags (e.g., too fast submissions).

Pause or Resume Study (Optional): The Researcher pauses the study (no new submissions) and later resumes it.

Reassign or Replace Tasks (Optional): The Researcher reassigns a task to a different participant or swaps an artifact if it's broken.

Export Results: The Researcher exports ratings, annotations, and metadata (CSV/XLSX) for analysis.

Close and Archive Study (Optional): The Researcher marks the study finished; data is locked and archived for later review.

2.3.3 Relationships and System Boundary

System Boundary: A rectangle labeled **"Create and Manage Studies"** encloses all sub-use cases above. Actors (Researcher, Admin, Participant, External Email Service) interact from outside this boundary.

Relationships: Manage Enrollment includes **Send Email Invites** (invites are part of enrolling). **Configure Tasks** includes **Randomize Order (Blinded)** when the study is blinded (guard: blinded = true). **Monitor Progress** extends **Pause/Resume Study**, **Reassign/Replace Tasks**, and **Export Results** (optional controls used as needed). **Admin** may extend **Close and Archive Study** for policy/lifecycle actions outside the normal researcher flow.

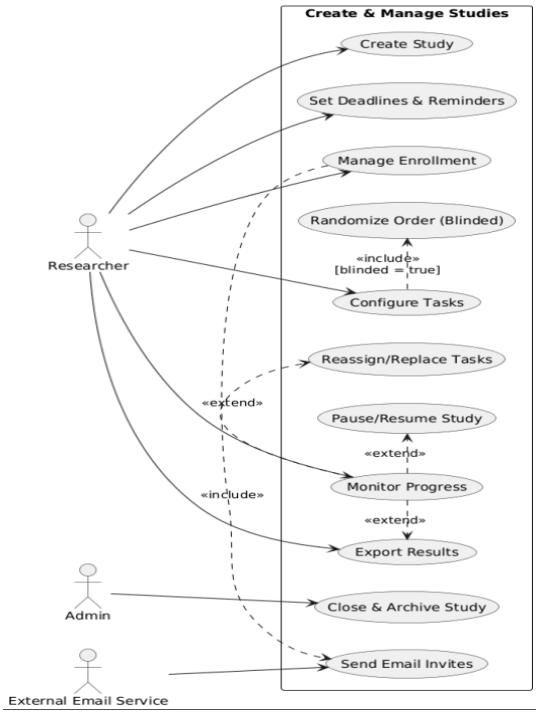

Figure 2.3: Create and Manage Studies

2.4 Assess Participant Competency

2.4.1 Actors

Researcher: The primary actor who starts the process. The Researcher creates the questionnaires and quizzes, views the collected data, and uses this data to select suitable participants for studies.

Participant: The actor who interacts with the assessments which were created by the Researcher. The Participant completes questionnaires about their background and takes technical quizzes to demonstrate their skills.

External Tool API: The system interacts with this API to dynamically generate quiz questions using an LLM, as requested by the Researcher.

2.4.2 Sub-Use Cases(Scenarios)

Create Background Questionnaire: The Researcher designs and saves a questionnaire to collect information and test it about a participant's background and experience.

Create Competency Quiz: The Researcher creates a technical quiz, consisting of questions like multiple-choice or code snippets, to evaluate a participant's skills.

Generate Quiz Questions via LLM: An optional feature where the Researcher can prompt the system to use an external LLM to help generate questions for a competency quiz.

Complete Background Questionnaire: The Participant fills out and submits the background questionnaire assigned to them as part of a study enrollment.

Take Competency Quiz: The Participant completes the assigned technical quiz to demonstrate their expertise.

View Competency Data: The Researcher reviews the submitted questionnaire answers and quiz scores for each participant.

Filter Participants: The Researcher uses the stored competency data to filter and select participants who meet specific criteria for a study.

2.4.3 Relationships and System Boundary:

System Boundary: The diagram uses a rectangle to clearly define the boundary of the **"Assess Participant Competency."** All sub-use cases occur within this system, while the actors interact with it from the outside.

Relationship: The diagram shows an arrow from **Generate Quiz Questions via LLM** to **Create Competency Quiz** with an "**<<extend>>"** label. This correctly models that AI-assisted question generation is an **optional** functionality. A researcher can fully create a quiz manually, but has the option to extend that process by using the AI feature.

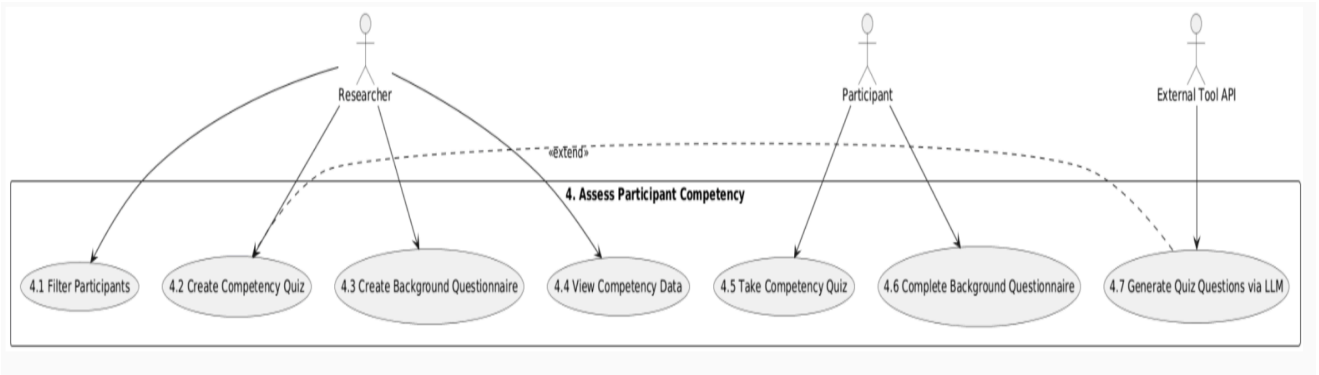

Figure 2.4: Assess Participant Competency

2.5 Evaluate Artifacts

2.5.1 Actors

Participant: The primary actor who performs artifact evaluation and comparison tasks within assigned studies. They read provided instructions, review artifacts in a side-by-side interface, annotate and rate them based on defined criteria, and submit their evaluations.

Researcher: Configures evaluation criteria and task types, defines blinded or comparison modes, monitors participant submissions, and may review flagged artifacts or inconsistencies.

Admin: Oversees data integrity, enforces evaluation policies, manages access control, and reviews flagged content or incomplete evaluations if required.

External Analysis Tool: Provides automatically generated metrics or reports that appear alongside artifacts during participant evaluation.

Authentication and Notification Service: Handles secure login for participants and sends system notifications such as submission confirmations or evaluation reminders.

2.5.2 Sub-Use Cases(Scenarios)

Start or Resume Evaluation: The participant accesses the evaluation environment for a given study, authenticates securely, and continues any saved progress.

Read Instructions and Criteria: The system displays the researcher-defined task description, evaluation mode, and rating criteria such as readability, correctness, and completeness.

View and Interact with Artifacts: Participants open the side-by-side artifact viewer, which supports synchronized scrolling, zooming, and safe viewing of code, diagrams, or documents.

Annotate and Comment: Participants highlight specific portions of an artifact and add inline comments, annotations, or tags to justify their evaluations.

Rate Artifacts: Using the given rubric, participants assign scores or qualitative ratings for each criterion. The system automatically checks that all required fields are completed.

Compare and Rank Artifacts: When comparison mode is active, participants evaluate multiple artifacts simultaneously, rank them, or indicate preferences among alternatives.

Integrate Automated Metrics: The system retrieves results from external analysis tools and displays these metrics next to artifacts to enrich the evaluation context.

Flag or Report Issues: Participants can flag artifacts that are broken, irrelevant, or contain inappropriate content. The researcher or admin reviews and resolves such flags.

Auto-Save and Draft Storage: The platform continuously saves ratings and comments, allowing participants to leave and later resume without data loss.

Validate and Submit Evaluation: Before submission, the system verifies completeness, ensures time-on-task thresholds and required comments are met, and

then securely stores the finalized results.

Review or Calibrate: Researchers may provide calibration examples or feedback summaries after participants submit, allowing participants to see model evaluations and learn expected standards.

2.5.3 Relationship and System Boundary

System Boundary: A rectangle labeled “Evaluate Artifacts” encloses all sub-use cases above. Actors (Participant, Researcher, Admin, External Analysis Tool, Authentication and Notification Service) interact from outside this boundary.

Relationships: Rate Artifacts includes Integrate Automated Metrics (metrics appear during evaluation). Validate and Submit Evaluation includes Auto-Save and Draft Storage (progress is saved continuously). Compare and Rank Artifacts extends Rate Artifacts when comparison mode is active (guard: comparison = true). Flag or Report Issues extends View and Interact with Artifacts when a problem is detected (guard: issue = true). Review or Calibrate extends Validate and Submit Evaluation when calibration is enabled (guard: calibration = true). Resolve Conflicts extends Validate and Submit Evaluation when inconsistent evaluations occur (guard: conflict = true).

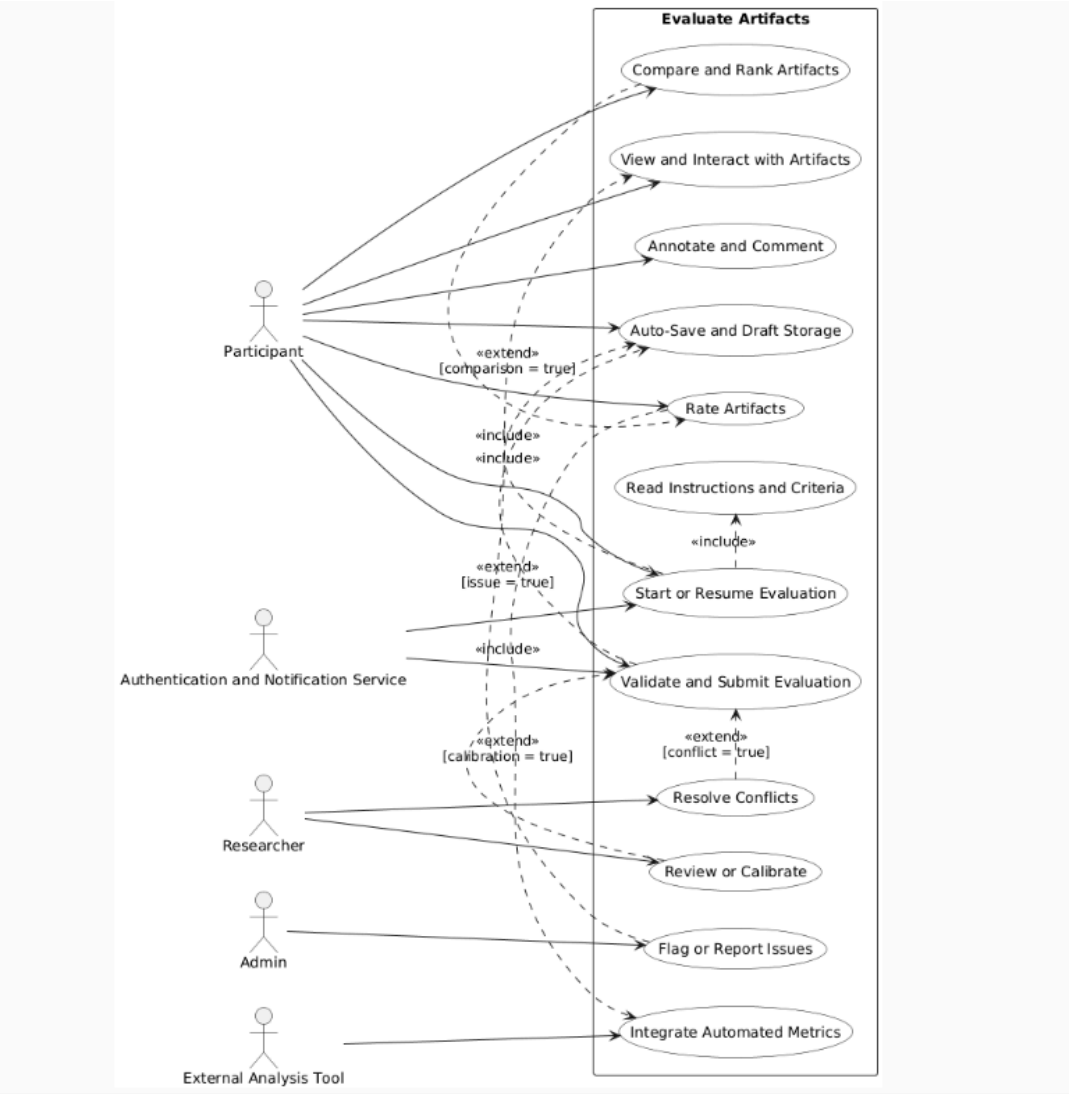

Figure 2.5: Evaluate Artifacts

2.6 Use Dashboards

2.6.1 Actors

Researcher: Uses dashboards to track their studies, see progress/QC flags, and export data.

Admin: Oversees all studies, manages user roles, applies policy/lifecycle actions, and exports reports.

Participant: Sees assigned tasks, due dates, progress, and history.

Notifications Service (Email/Push): Sends reminders and alerts triggered by the system.

2.6.2 Sub-Use Cases(Scenarios)

Researcher/Admin Dashboard Overview: The Researcher or Admin views active studies with completion percentages, quick filters, and quality-control flags.

Participant Dashboard Overview: The Participant sees assigned tasks, due-soon reminders, progress bars, and a history of submitted evaluations.

Filter and Search: The user filters by study, time window, participant, status, or tags, and searches by keyword to quickly find relevant items.

View Study Details: The Researcher or Admin opens a study's details to see tasks, participants, timelines, and per-task status.

View QC Flags: The Researcher or Admin reviews quality-control indicators such as "too fast," missing ratings, or anomalies, and navigates to the affected submissions.

Notifications Center: The user sees reminders and alerts (e.g., upcoming deadlines, reassigned tasks, responses to flags) generated by the system.

Export and Reporting: The Researcher or Admin downloads ratings, annotations, and related metadata as CSV/XLSX/PDF for offline analysis.

Configure Dashboard Widgets (Optional): The Researcher or Admin customizes which cards/metrics are shown on the dashboard and saves a preferred view.

2.6.3 Relationships and System Boundary

System Boundary: A rectangle labeled **"Use Dashboards"** contains the dashboard features above. Actors (Researcher, Admin, Participant, Notifications Service, Export Service) interact from outside.

Relationships: Researcher/Admin Dashboard includes **View Study Details**, **View QC Flags**, and **Filter and Search**, and includes **Export and Reporting** for downloads. **Participant Dashboard** includes **Filter and Search** and **Notifications Center** for reminders. **Configure Widgets** extends **Researcher/Admin Dashboard** as an optional personalization step.

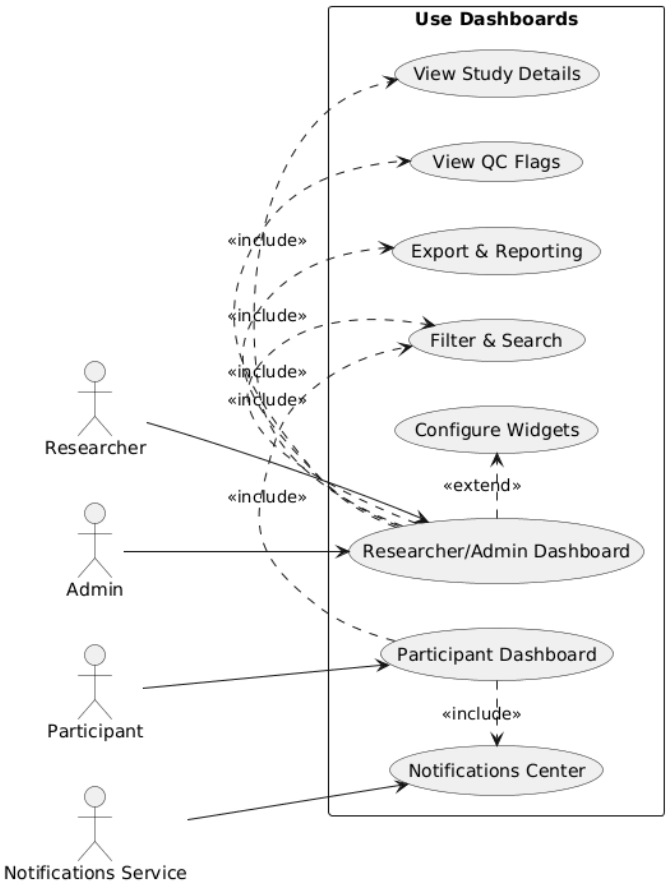

Figure 2.6: Use Dashboards

3 Non-Functional Requirements

3.1 Performance and Responsiveness

- **Page response time**: For task open and dashboard pages, 95% of requests complete <= 2.5s under 100 concurrent users with artifacts <= 1 MB.
- **Annotation latency**: Select/highlight/comment operations apply in UI in <= 150 ms. Demo with browser perf marks.
- **Comparison rendering**: 3-way view initial render <= 1.2s for 500-line code artifacts.

3.2 Security and Privacy

- **Role-based access control (RBAC) enforcement:** All protected endpoints require JWT; role claims gate access. Attempting forbidden action returns **403** and is logged.
- All app traffic uses **HTTPS** (TLS).
- Secrets come from **Spring Boot env/config** (not in code).

- **Data protection:** Passwords are hashed before being stored in the database to protect sensitive user data.

3.3 Usability and Accessibility

- **Accessibility:** Easy to use with keyboard and mouse, readable text with good contrast, tool-tips for UI elements and more.
- **Discoverability:** First-time user tips; help icons on criteria and metrics.

3.4 Reliability and Availability

- **Uptime (hosted mode):** >= **99.5%** weekly during study windows; incident log published.
- **Autosave:** Drafts saved every **10s** or on blur to save time and effort if the page closes unexpectedly.

3.5 Scalability

- **Data scale:** Handle **1000 artifacts** and **20 concurrent study sessions** with p95 page **<= 3.5s**.
  Extensibility: Add a new artifact type by implementing more extension points to existing artifact templates.

3.6 Maintainability and Testability

- **Extensive testing:** Unit + API tests on every PR; **>= 70%** line coverage for core modules.

3.7 Compatibility and Deployment

- **Browsers:** Last 2 stable versions Chrome/Firefox/Edge/Opera; functional fallback for Safari.
- **Local and Hosted deployment:** One-command local (Node.js) and hosted deployment ready; setup **<= 15 min** documented for Windows/macOS/Linux.

3.8 Internal APIs

- **Internal API:** Provide REST endpoints for data submission/retrieval.

4 Selected Tech Stack and Architecture

- **Frontend: React.js (JavaScript):** Component model suits complex evaluator UI (side-by-side/3-way, synchronized scrolling, rich annotation). Strong ecosystem speeds delivery of the usability targets.
- **Backend: Java Spring Boot:** Opinionated, testable structure (controllers/services/repositories/guards), easy JWT/RBAC (Spring Security), and robust integrations for metrics, mail, job queues, and aligns with “internal APIs expected” and OO design emphasis.
- **Database: Supabase (PostgreSQL):** Managed PostgreSQL with great SQL + JSONB, RLS if desired, and **Supabase Storage** for artifact blobs (code/text/images/PDF). Extensible for future artifact types as requested.**Deployment Tools: Vercel**: In order to host the React frontend for a fast global CDN, automatic HTTPS, and zero-config preview deployments.

6 Responsibility Split

1 Project Overview and Scope - Shared Work

2.1 Manage Users and Roles - Shared Work

2.2 Upload and Organize Artifacts - Ali Şen

2.3 Create and Manage Studies - Muhammet Furkan Demir

2.4 Assess Participant Competency - Barış Peksak

2.5 Evaluate Artifacts - Berkay Şimşek

2.6 Use Dashboards - Gökay Nuray

3 Non-Functional Requirements - Shared Work

4 Selected Tech Stack and Architecture - Shared Work