SmartResearch

AI-powered Academic Paper Summarisation & Clustering

CSIT321 Project Specification Presentation

Meet the team

Dr. Jack Yang

Advisor

Chelsea

Project Manager / QA / Swing Developer

Noor

Frontend Lead (UI/UX)

Meet the team

Nishad

Technical Lead
(NLP & Clustering)

Azwad

Infrastructure Lead
(Integration)

Why SmartResearch Is Needed

Millions of papers published yearly

Flood of academic output makes staying current hard to manage

Manual screening is slow & fragmented

Researchers waste hours scanning abstracts and filtering by hand

Existing tools solve only part of the workflow

Current solutions don’t integrate summarisation, clustering, and export

What SmartResearch Delivers

01

Batch PDFs

Process large sets at once.

02

Summarise

Objective, Method, Findings, Limitations.

03

Clustering

Group research into topics.

04

Export

CSV/JSON for integration.

Functional Requirements

Batch Upload

Upload ≥10 PDFs at once for processing.

Summarisation

Extract Objective, Method, Findings, Limitations

Clustering

Group papers by theme with keyword labels

User Interface

Upload View

Drag-and-drop or select PDFs.

Browse View

Search, filter, and review summaries.

Cluster View

Visualise groups by themes and labels.

Export

Download results as CSV/JSON.

Non-Functional Requirements

Performance

≤60s per 10 PDFs processed.

Usability

≤3 clicks to view each paper.

Reliability

No crashes with malformed PDFs.

Privacy

Local-only processing, no third-party API calls.

Constraints

Language

English, text-based PDFs only.

Technology

Open-source stack.

Timeline

Limited to semester project scope.

Market & Analysis

Competitor Tools

Scholarycy, ResearchRabbit, and Elicit = partial solutions only

Gap in Workflow

No single tool covers the full pipeline

Our Advantage

End-to-end: upload -> summarise -> cluster -> export

Impact

Cuts screening from days to minutes

Design Principles

Simplicity

Workflow stripped to essentials only

Accessibility

Any document in ≤3 clicks

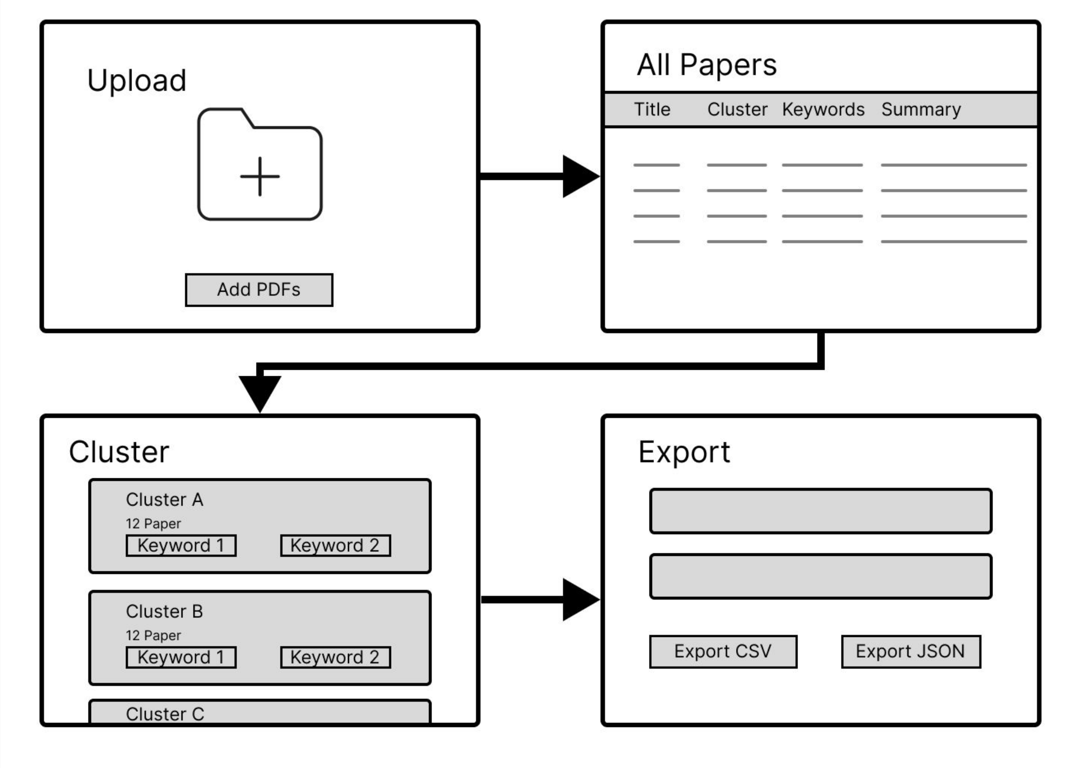
Clarity

Summaries in a 4-field structure

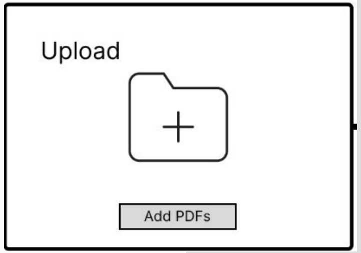
Consistency

Uniform layout across all views

User flow & Mockups

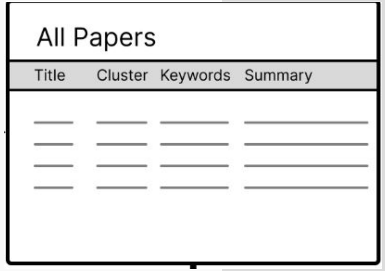


Features



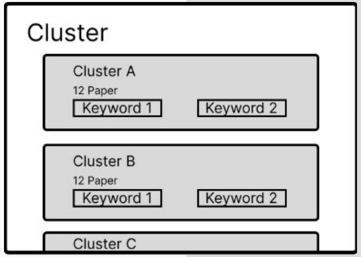
Fast Uploads

Drag-and-drop batches with progress



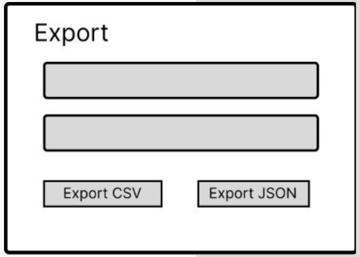
Smart Summaries

Four-field format for quick scan



Thematic Clusters

Auto groups with keyword labels



Easy Export

CSV/JSON for reuse in tools

Technical Requirements (Backend)

Summariser

Four-field NLP summaries

Clustering

Groups with keyword labels

API layer

FastAPI links front and back

Database

SQLite stores docs and results

Technical Requirements (Infrastructure)

Containerisation

Docker for consistent builds

CI/CD

GitHub Actions for testing + deploy

Privacy

Local-only, no third-party calls

Stability

Error handling + recovery routines

Supervisor/Client Interaction

01

Regular Meetings

Weekly–fortnightly check-ins with Jack

02

Scope Validation

Feedback used to align scope and goals

03

Task Management

GitHub Projects (Kanban) for sprint tracking

04

Accountability

Escalation process for uncompleted tasks

Risks Involved

High Priority

Project Management

Technical

Usability & Org

High Priority

Summary fidelity

Hybrid models, rubric checks, peer review

Cluster coherence

Parameter tuning, pruning, keyword labels

Consistency

Cross-validation, sample benchmarking

Project Management

Schedule Slippage

Milestone gates, fortnightly sprints, progress tracking

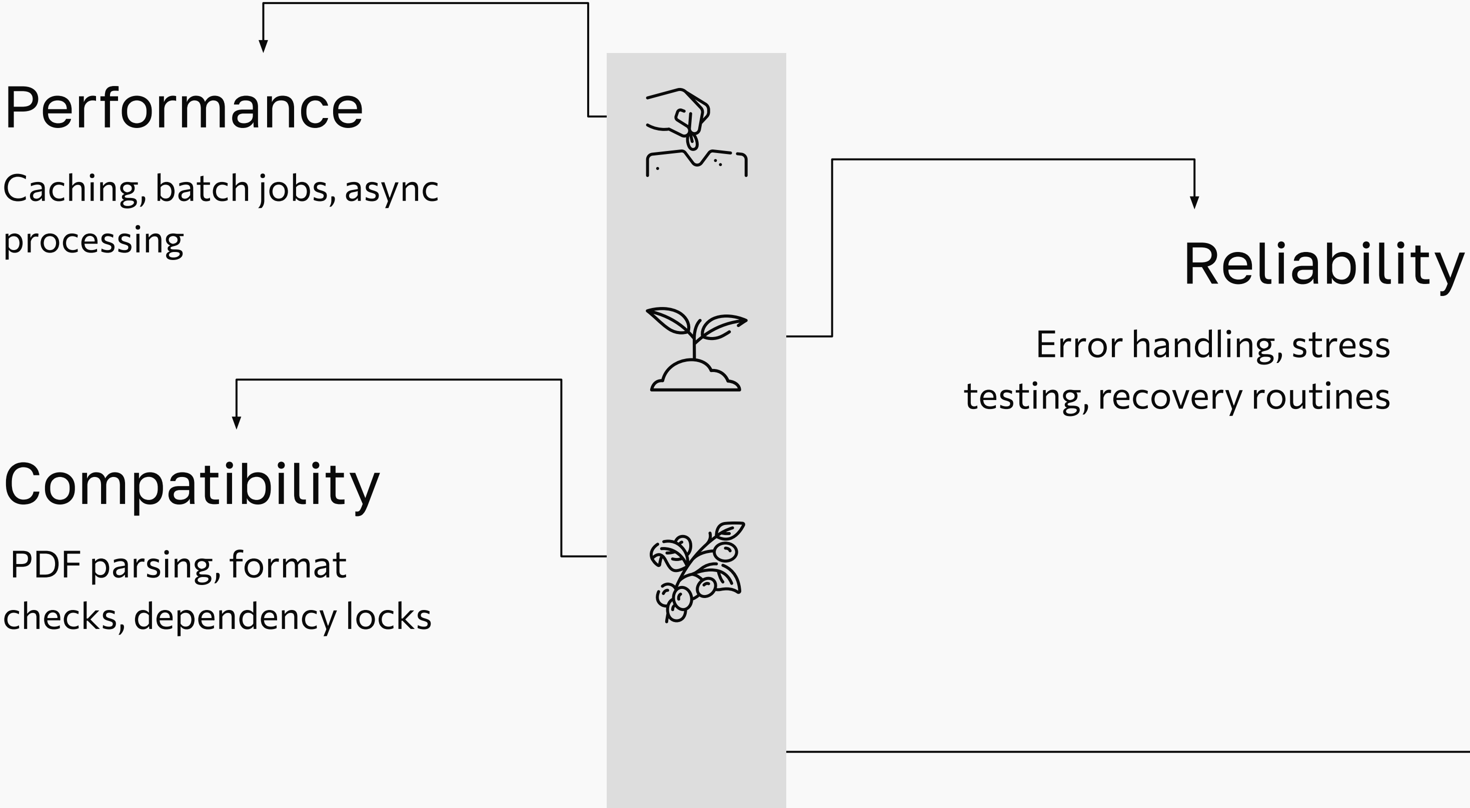
Scope Creep

GitHub issues, acceptance criteria, change control

Team load

Clear RACI, rotating tasks, early escalation

Technical



Performance

Caching, batch jobs, async processing

Reliability

Error handling, stress testing, recovery routines

Compatibility

PDF parsing, format checks, dependency locks

Usability & Organisational

UI Confusion

UX testing, simplified flows, walkthrough guides

Coordination

Discord reminders, shared agendas, written notes

Adoption Risk

Feedback loops, quick-start docs, minimal learning curve

Timeline and Sprint Plan

01

Wks 1-3

Group formation, allocation, repo setup

02

Wks 4-5

Specification draft, repo setup, risk register

03

Wks 6-7

Wireframes, workflows, presentation slides

04

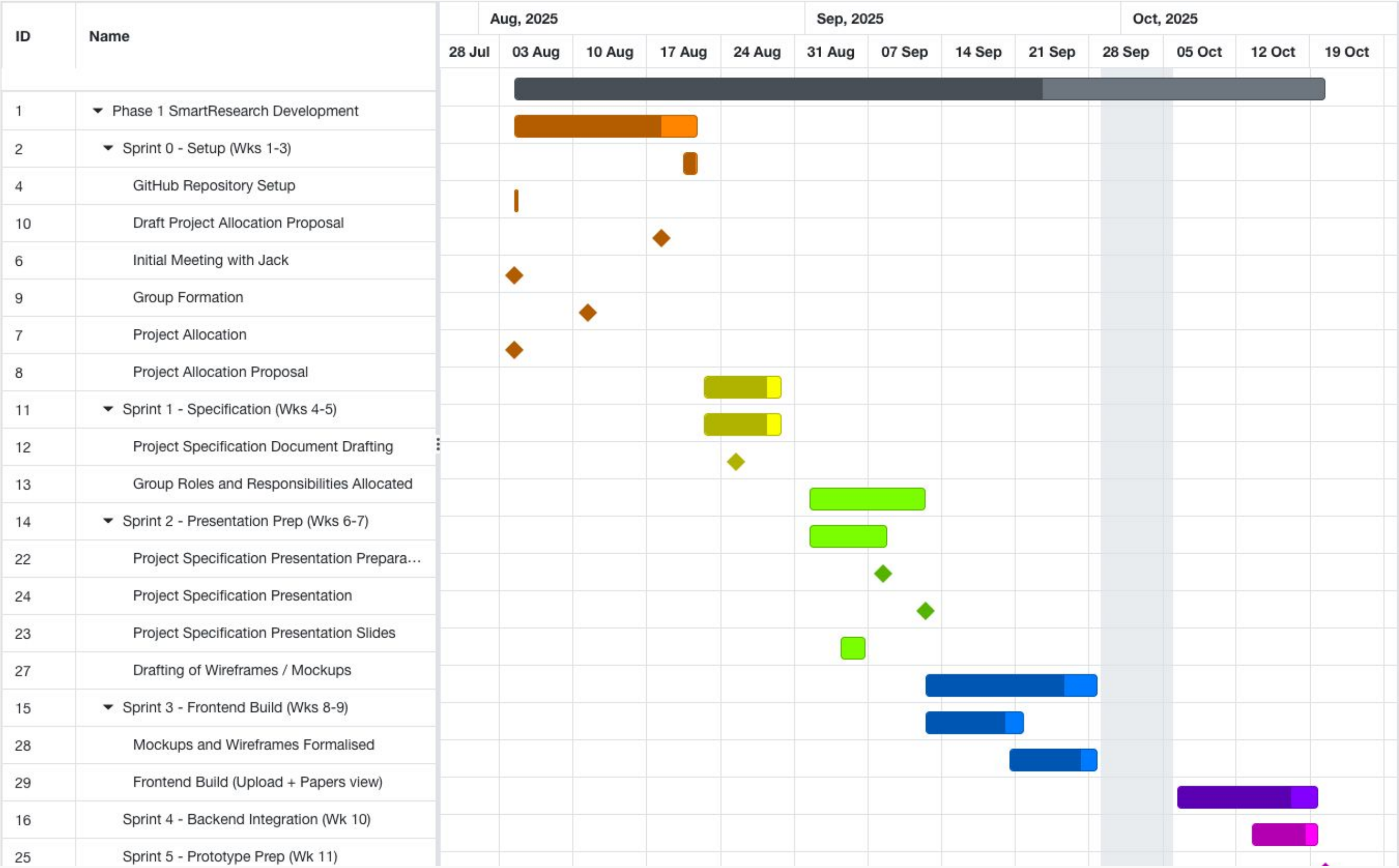
Wks 8-11

Frontend, backend, prototype prep

05

Wk 12

Prototype demo + Phase 1 wrap-up



ID	Name
1	Phase 1 SmartResearch Development
2	Sprint 0 - Setup (Wks 1-3)
4	GitHub Repository Setup
10	Draft Project Allocation Proposal
6	Initial Meeting with Jack
9	Group Formation
7	Project Allocation
8	Project Allocation Proposal
11	Sprint 1 - Specification (Wks 4-5)
12	Project Specification Document Drafting
13	Group Roles and Responsibilities Allocated
14	Sprint 2 - Presentation Prep (Wks 6-7)
22	Project Specification Presentation Prepara…
24	Project Specification Presentation
23	Project Specification Presentation Slides
27	Drafting of Wireframes / Mockups
15	Sprint 3 - Frontend Build (Wks 8-9)
28	Mockups and Wireframes Formalised
29	Frontend Build (Upload + Papers view)
16	Sprint 4 - Backend Integration (Wk 10)
25	Sprint 5 - Prototype Prep (Wk 11)

Gantt Chart

Conclusion

- Workflow: upload → summarise → cluster → export
- Unique: full pipeline vs partial tools
- Simple, clear, consistent design
- Risks managed, sprint plan set