
# Sanskrit Manuscript Fragment Reconstruction Platform

## *Requirements Standard Plan*

Dylan Garner

Version 1, 2025-10-01

# Table of Contents

# Academic Integrity Disclaimer

We would like to acknowledge that as a dedicated students of McMaster University, we have thoroughly read and comprehended the Academic Integrity Policy published by the university. We are committed to upholding the principles of academic honesty and integrity in all aspects of our educational journey. We understand the importance of acknowledging the work and ideas of others, and we pledge to ensure that all our academic endeavors are conducted with the utmost originality and compliance with the university's policy.

**We affirm that the content presented in this document is entirely our own, and any external sources used have been appropriately cited and referenced.**

# Control Information

| Version | Delivery | | Feedback | |
|---|---|---|---|---|
| | *Deadline* | *Delivered* | *Received* | *Integrated* |
| **V1** | | | | |
| **V2** | | | | |
| **V3** | | | | |

# (G) Goals

*Goals are "needs of the target organization, which the system will address". While the development team is the principal user of the other books, the Goals book addresses a wider audience: essentially, all stakeholders* [1]

*It must contain enough information to provide — if read just by itself — a general sketch of the entire project. To this effect, chapter G.3 presents a short overview of the system and G.1 will typically include some key properties of the environment. As it addresses a wide readership, it should be clear and minimize the use of specialized technical terms. Together, G.1, G.2 and G.3 describe the rationale for the project. It is important to state these justifications explicitly. Typically, they are well understood at the start of the project, but management and priorities can change* [1]

## Control Information

*Table 1. Sanskrit Manuscript Fragment Reconstruction Platform — Versionning Information — Goal Book*

| Section | Version | Lead | Delivered on | Reviewer | Approved on |
|---|---|---|---|---|---|
| G.1 | | | | | |
| G.2 | 1.0 | Umar | October 2nd, 2025 | | |
| G.3 | | | | | |
| G.4 | | | | | |
| G.5 | | | | | |
| G.6 | | | | | |
| G.7 | | | | | |

## (G.1) Context and Overall Objectives

*High-level view of the project: organizational context and reason for building a system. It explains why the project is needed, recalls the business context, and presents the general business objectives.* [1]

The textual history of Indian Buddhism is fragmented across thousands of manuscript folios, preserved only in partial, damaged, or scattered forms. Traditionally, scholars reconstruct these texts manually through paleographic study, transcription, and content comparison; a slow and error-prone process. Scholars in Religious Studies increasingly have access to digitized collections of ancient manuscripts, such as the approximately 21,000 photographed fragments in one major archive. While some existing computational tools exist for pattern recognition and Optical Character Recognition (OCR) none can tackle this problem as they are not designed for irregular, damaged, or arbitrarily oriented manuscript fragments. Without computational assistance, much of the potential in these digital collections remains

locked and underutilized. Our goal is to build an end to end system that can take these fragments as inputs and return a list of the most likely matches based on certain Artificial Intelligence (AI) algorithms. This coupled with a User Interface (UI) that can be easy for users to learn and help them with their research in constructing these fragments.

### The general objectives of the project are to:

- Develop AI-supported software capable of preprocessing, analyzing, and matching manuscript fragments to accelerate scholarly reconstruction.
- Create tools to automatically regularize image orientation, detect edges and shapes, identify paleographic scripts, and perform OCR transcription tuned for ancient Sanskrit manuscripts.
- Provide a searchable, structured database that enables scholars to track, organize, and connect fragments across collections.

*Given the complexity of the project, the work will be divided into clusters, each addressing a distinct but interdependent stage of the scholarly workflow (More information on this in section (S.1) Components):*

- **Front-end Cluster** — Scholar-facing web app for secure login, batch uploads, an interactive canvas to arrange fragments, view AI matches, and save sessions.
- **Backend Cluster** — Core services (Application Programming Interface (API) gateway, authentication, preprocessing orchestration, database access) that power and persist all workflows.
- **AI/Machine Learning (ML) Cluster** — Model services for edge/damage matching, script/handwriting classification, OCR text extraction, and content similarity scoring.

Together, these clusters form an integrated system that not only assists in reconstructing ancient manuscripts but also builds a model for applying computational puzzle-solving approaches to the humanities at large.

## (G.2) Current situation

*Current state of processes to be addressed by the project and the resulting system. It describes the current situation, upon which the system is expected to improve* [1]

As of now, the reconstruction of Indian Buddhist text is a manual and scholar driven process. This process relies on comparing the edges, shapes, and damage of the fragments to to try and find relationships between them. Scholars have to traverse over thousands of fragments across multiple collections with little to no aid from existing computational tools. The current workflow typically involves:

- Visual inspection of the fragments to identify features like number of lines, edges, shapes and damage patterns to possibly find relationships between them
- Paleographic study of the fragments to identify the script type and writing style. This can help determince region and time period of the fragments
- Manual transcription of the fragments to identify the content and compare it with other fragments
- Cross referencing fragments with other known manuscripts to infer relationships or reconstruct missing sections

As touched upon previously, current computational tools provide insufficient support for these tasks and

are limited to simple OCR on clean text, basic string searches, or pattern recognition methods that fail when applied to irregular, damaged, or rotated manuscript fragments. As a result, the reconstruction process remains slow, error-prone, and highly dependent on the availability and judgment of individual scholars. This bottleneck not only delays large scale reconstruction efforts but also leaves significant portions of these manuscript collections underutilized, limiting progress in uncovering the cultural and historical insights they contain.

# (G.3) Expected Benefits

*New processes, or improvement to existing processes, made possible by the project's results. It presents the business benefits expected from the successful execution of the project.* ***This chapter is the core of the Goals book****, describing what the organization expects from the system. It ensures that the project remains focused: if at some stage it gets pushed in different directions, with "creeping featurism" threatening its integrity, a reminder about the original business goals stated in those chapters will help.* [1]

## Overview

The Artificial Intelligence (AI)-powered manuscript fragment reconstruction platform for Buddhist Studies scholars addresses a challenge in digital humanities by transforming an overwhelming manual process into a scalable, collaborative, AI-enhanced workflow.

## Primary Business Benefits

### 1. Dramatic Scale Increase in Reconstruction Capability

- **Current limitation**: Manual reconstruction is extremely time-consuming and tedious
- **Expected benefit**: AI-assisted matching will enable processing of hundreds to thousands of fragments systematically
- **Impact**: Unlocks the potential to meaningfully progress through the 21,000+ fragments possessed by the british library as well as other institutions

### 2. Acceleration of Scholarly Research

- **Time savings**: Reduce fragment matching from hours/days per potential match to minutes through AI-powered suggestions
- **Quality improvement**: Multiple AI-generated hypotheses allow scholars to evaluate more possibilities than humanly feasible
- **Research velocity**: Enable completion of reconstruction projects within significantly shorter timelines

### 3. Methodological Innovation in Digital Humanities

- **Technical advancement**: Pioneer AI applications for irregular archaeological artifact reconstruction
- **Reproducible research**: Establish systematic methodologies for similar reconstruction challenges in

other domains
- **Research infrastructure**: Create reusable computational tools for manuscript studies globally

### Strategic Value Proposition

The platform represents a shift from individual scholarly craft to collaborative, AI-augmented research infrastructure. This transformation enables:

- **Scalability**: Moving from individual fragment analysis to systematic processing of entire collections
- **Sustainability**: Creating methodology that can be maintained and improved over time
- **Innovation**: Developing computational approaches that extend human expertise rather than replacing it

## (G.4) Functionality overview

*Overview of the functions (behavior) of the system. Principal properties only (details are in the System book). It is a short overview of the functions of the future system, a kind of capsule version of book S, skipping details but enabling readers to get a quick grasp of what the system will do.* [1]

### Core System Functions

The Sanskrit Manuscript Fragment Reconstruction Platform provides a web-based interface for Buddhist Studies scholars to systematically search, filter, and reconstruct ancient texts from the British Library's collection of 21,000 manuscript fragments. The system operates through three primary functional domains:

### 1. Fragment Preprocessing and Analysis

**Image Processing Pipeline** * Automatically segment uploaded fragment images to isolate manuscript content from background * Determine actual physical size measurements of fragments from image data * Extract edge patterns, damage signatures, and physical characteristics for matching * Normalize images across different photography conditions and lighting for consistent analysis

**Metadata Extraction** * Extract script characteristics, writing style patterns, and paleographic features * Identify number of text rows, folio information, and other codicological features * Generate searchable metadata attributes for filtering and discovery * Process both British Library collection and user-submitted images through the same pipeline

### 2. Criteria-Based Search and Filtering System

**Multi-Criteria Filtering Interface** * Allow researchers to enter specific fragment IDs for targeted searches * Filter fragments by script type, number of text rows, folio numbers, physical dimensions * Apply damage pattern matching and edge compatibility criteria * Combine multiple filter criteria to narrow search results to relevant fragments

**Puzzle-Like Matching Engine** * Compare physical edge patterns and damage signatures for potential fits * Identify fragments with compatible breaks, tears, and wear patterns * Generate ranked lists of potential matches based on physical compatibility * Support both automated matching suggestions and manual criteria specification

### 3. Interactive Reconstruction Workspace

**Canvas-Based Assembly Interface** * Provide drag-and-drop canvas for virtually arranging and testing fragment combinations * Enable real-time manipulation of fragment positions and orientations * Support snap-to-fit functionality based on edge pattern analysis * Allow researchers to build and test reconstruction hypotheses visually

**Session Management** * Save and restore reconstruction work sessions * Support multiple concurrent reconstruction projects * Enable export of reconstruction layouts and fragment arrangements * Track reconstruction progress and fragment relationship discoveries

### Integration and Data Management

**British Library Integration**: Primary integration with British Library's collection of 21,000 manuscript fragment images, with automated processing pipeline for the complete collection.

**User Image Submission**: Support for researchers to upload their own fragment images, which are automatically processed through the same segmentation, measurement, and feature extraction pipeline as the core collection.

**Metadata and Provenance Management**: Maintain comprehensive metadata including fragment IDs, provenance information, physical measurements, and extracted features while preserving original catalog information.

### Workflow Integration

The system supports a streamlined workflow: researchers enter fragment IDs or apply filter criteria → system returns matching fragments based on specified characteristics → researchers use the interactive canvas to test physical combinations → save reconstruction sessions for future work. The platform architecture enables efficient processing of the 21,000-fragment collection with responsive search and real-time canvas manipulation.

## (G.5) High-level usage scenarios

*Fundamental usage paths through the system. It presents the main scenarios (use cases) that the system should cover. The scenarios chosen for appearing here, in the Goals book, should only be the **main usage patterns**, without details such as special and erroneous cases; they should be stated in user terms only, independently of the system's structure. Detailed usage scenarios, taking into account system details and special cases, will appear in the System book (S.4).* [1]

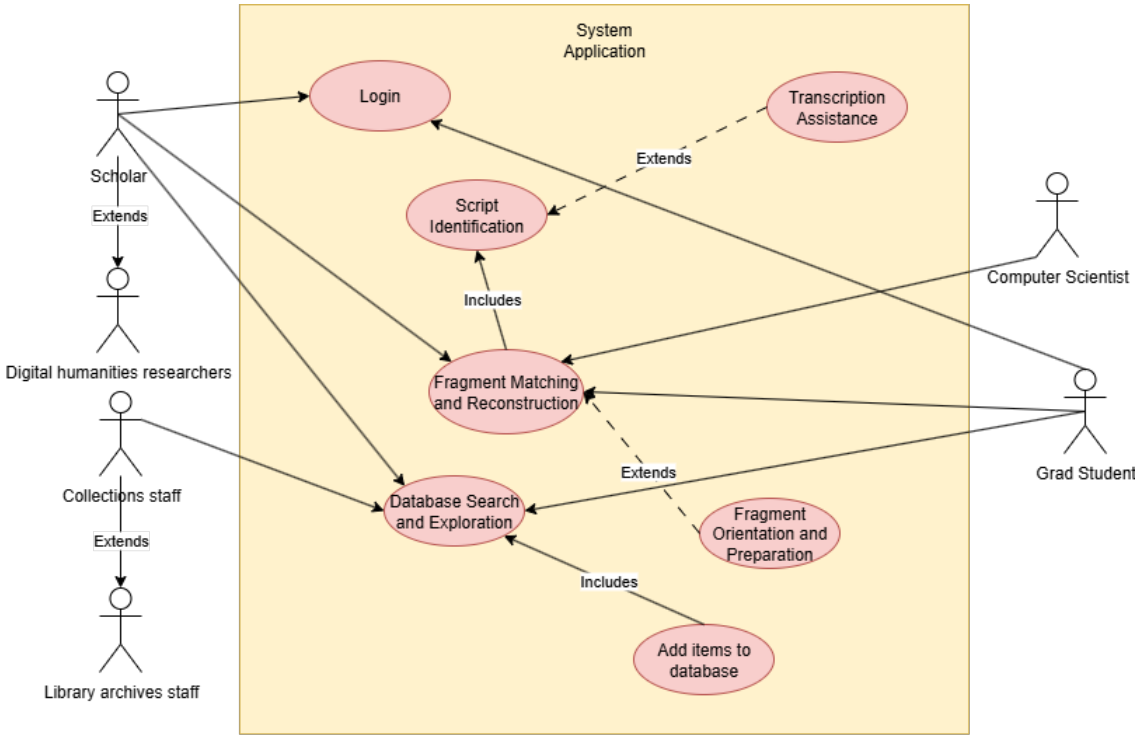


*Figure 1. High Level use cases diagram*

## System Overview

The system will support scholars of Buddhist textual history by providing tools that automate and accelerate the process of reconstructing fragmented manuscripts. The following main usage scenarios capture the essential ways in which users will interact with the system:

## Fragment Orientation and Preparation

- A scholar uploads a set of digitized manuscript images.
- The system automatically cleans, crops, and regularizes the orientation of the fragments so that they are ready for analysis.

## Fragment Matching and Reconstruction

- The scholar requests possible matches for a selected fragment.
- The system scans edges, shapes, and damage patterns, and proposes a ranked set of likely matches.
- The scholar inspects the proposed matches and confirms or rejects them.

## Script Identification

- The scholar views a fragment of uncertain origin.
- The system automatically identifies the paleographical script (e.g., Sanskrit, Tibetan, Chinese) and displays the classification to the user.

### Transcription Assistance

- The scholar opens a fragment and requests an initial transcription.
- The system provides a machine-generated transcription of visible characters, which the scholar can review, correct, and annotate.

### Database Search and Exploration

- The scholar queries the system's database (e.g., "find all fragments in Gupta script with more than 5 visible lines of text").
- The system returns results that can be sorted, compared, and linked with other fragments.

### Login

- The scholar opens the system's login page.
- The scholar enters their username and password.
- The system verifies the credentials.
- If valid, the system grants access and displays the main dashboard.

## (G.6) Limitations and Exclusions

> *Aspects that the system need not address. It states what the system will not do. This chapter addresses a key quality attribute of good requirements: the requirements must be delimited (or "scoped"). G.6 is not, however, the place for an analysis of risks and obstacles, which pertain to the project rather than the goals and correspondingly appears in chapter P.6.* [1]

To clearly delimit the project scope, the following activities are explicitly excluded from the system's objectives:

### Manual Scholarly Judgment

- The system will not replace the expertise of scholars in making final decisions about fragment matches, transcriptions, or interpretations. It will only provide automated suggestions and tools for assistance.

### Non-Buddhist Manuscripts

- The system will not initially support manuscripts outside the field of Indian, Tibetan, or Chinese Buddhist textual traditions. Other languages, scripts, or literary traditions are out of scope.

### Full Translation into Modern Languages

- The system will not provide full automatic translations of reconstructed texts into modern languages (e.g., English, French, Hindi). Its role ends at transcription and cross-referencing.

### Physical Conservation or Restoration

- The system will not engage in physical preservation, repair, or handling of manuscripts. It applies only to digital images of fragments.

### Non-Standard Inputs

- The system will not accept non-image inputs (e.g., handwritten notes, modern typed transcriptions) for automated processing. Only digitized images of fragments will be handled.

### Error Correction of Scholarly Input

- The system will not validate or correct scholars' manual annotations, transcriptions, or metadata edits. Responsibility for the accuracy of human input lies with the user.

## (G.7) Stakeholders and requirements sources

*Groups of people who can affect the project or be affected by it, and other places to consider for information about the project and system. It lists stakeholders and other requirements sources. It should define stakeholders as categories of people, not individuals, even if such individuals are known at the time of writing. The main goal of chapter G.7 is to avoid forgetting any category of people whose input is relevant to the project. It also lists documents and other information that the project, aside from soliciting input from stakeholders, can consult for requirements information.* [1]

*Table 2. Table — Stakeholder categories*

| Stakeholder | Persona | Category |
|---|---|---|
| Scholars studying Buddhist texts | Scholar in Buddhist Text Fragmentation (Shayne) | Direct |
| | Textual Critic of Sanskrit Buddhist Literature (Arun) | |
| Graduate students & research assistants (RAs) | Graduate Student in Buddhist Studies (Stacey) | Direct |
| | Graduate Research Assistant in Manuscript Studies (Pema) | |
| Other scholars working on different manuscript reconstructions (non-Sanskrit) | — | Indirect |
| Collection staff looking to store fragments online | — | Indirect |
| Digital humanities researchers studying textual transmission | — | Indirect |

# (G.7.1)Direct Stakeholders

## Scholars Studying Buddhist Text

Scholars that are working on the reconstruction of Buddhist texts will be directly interacting with the web application we plan on building. They will be using the web application to match fragments together based on many factors such as transcription, shape matching and other criteria they decide to choose. Scholars will be using the main functionality of this web application and therefore are direct stakeholders.

**Shayne (Scholar in Buddhist Text Fragmentation)** Shayne is a professor in Religious Studies at McMaster University specializing in Indian Buddhist manuscripts. He is quite busy as he splits his time between teaching and supervising graduate students and when he can he reconstructs folios from digitized fragments sourced across multiple archives. The process of reconstruction is quite slow and takes Shayne way too much time. He is looking for something that would help him cut down the time of this process to help him with his progress on his reconstruction.

**Arun (Textual Critic of Sanskrit Buddhist Literature)** Arun is a senior professor and series editor who balances teaching, and frequent research travel. His work focuses on reconstructing dispersed Sanskrit folios and checking them against Tibetan and Chinese parallels to establish reliable readings. Progress with this reconstruction is slow due to time-consuming manual checks such as dating scripts, aligning shapes of fragments, and looking for possible joins across multiple archives. He needs a way to speed this process up as he feels this takes too much of his time.

## Graduate Students and Research Assistants (RAs):

Graduate students and RAs support the reconstruction work of scholars by looking though large batches of fragments and trying to figure out if they can find matches or not. Sometimes they are given a certain task from the scholars to help them learn certain skills that will help them with future work in reconstruction. They will also be using the main functionality of this web application and therefore are direct stakeholders.

**Stacey (Graduate Student in Buddhist Studies)** Stacey is a second year masters student juggling school work as well as learning from her supervising professor on how to tackle problems like the reconstruction of Buddhist text. She is still learning Sanskrit and is sometimes having trouble with figuring out which characters are which. She is also having a hard time learning Sanskrit overall and wants something that can help her with the character recognition and learning Sanskrit.

**Pema (Graduate Research Assistant in Manuscript Studies)** Pema is a RA who also helps scholars and professors with their work with reconstruction of Buddhist texts. He enjoys sports such as basketball and hockey but is often finding it hard to find time to do either given that he spends too much time trying to find matching pieces of fragments to help his supervising professor. He would like something that could help him with the time commitment of his research to spend some time on the hobbies he enjoys.

# (G.7.2) Indirect Stakeholders

- **Other scholars working on different manuscript reconstructions (specifically ones not in Sanskrit):** While they benefit from the shape matchmaking functionality of the system the system was not built to support other languages that aren't Sanskrit. But they could definitely use the functionalities that don't require Sanskrit.
- **Collection staff looking to store fragments online:** If some researchers out there want a place to store their manuscript fragments online somewhere safe they could also do it using this web application. Though this is not the intended use of the system it is definitely a way in which someone can use the system
- **Digital humanities researchers studying textual transmission:** If given access to a scholars workspace they can look into the formed manuscripts and that could help them with their research in textual transmission.

# (G.7.3) Requirements sources

- **Personal Information Protection and Electronic Documents Act (PIPEDA)** - If the system collects or uses personal information (e.g., user accounts, contact details, usage logs), it must comply with PIPEDA: obtain consent, limit use to stated purposes, safeguard data, enable access/correction, and follow breach-notification rules.
  - https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/pipeda_brief/
- **Web Accessibility** - Validate the UI against WCAG 2.2 Level AA and Ontario's AODA (IASR) web accessibility requirements to ensure inclusive access (keyboard navigation, contrast, alt text, captions).
  - https://www.w3.org/TR/wcag/
  - https://www.ontario.ca/laws/regulation/110191
- **Protection of Restricted/Private Images (Data Security & Access Control)** - Restricted or unpublished manuscript images must be protected via role-based access, least-privilege permissions, encryption in transit and at rest.
  - https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final

# (E) Environment

*The Environment book describes the application domain and external context, physical or virtual (or a mix), in which the system will operate.* [1]

## Control Information

*Table 3. Sanskrit Manuscript Fragment Reconstruction Platform — Versionning Information — Environment Book*

| Section | Version | Lead | Delivered | Reviewer | Approved |
|---|---|---|---|---|---|
| E.1 | | | | | |
| E.2 | | | | | |
| E.3 | | | | | |
| E.4 | | | | | |
| E.5 | | | | | |
| E.6 | | | | | |

## (E.1) Glossary

*Clear and precise definitions of all the vocabulary specific to the application domain, including technical terms, words from ordinary language used in a special meaning, and acronyms. It introduces the terminology of the project; not just of the environment in the strict sense, but of all its parts.* [1]

**AI (Artificial Intelligence)**

The simulation of human intelligence processes by computer systems, including learning, reasoning, and problem-solving capabilities.

**Codicological**

Relating to codicology, the study of books as physical objects, especially manuscripts. This includes the analysis of materials, construction, decoration, and other physical aspects of manuscripts such as folio numbers, text row counts, and structural features.

**Fragment**

A piece or portion of an ancient manuscript that has been separated from the original complete text through damage, deterioration, or other processes over time.

**Palaeography**

The study of ancient and historical handwriting, including the deciphering, analysis, and dating of historical manuscripts based on writing style, script characteristics, and other textual features.

**Reconstruction**

The process of piecing together manuscript fragments to restore or approximate the original arrangement and content of ancient texts.

# (E.2) Components

*List of elements of the environment that may affect or be affected by the system and project. It includes other systems to which the system must be interfaced. These components may include existing systems, particularly software systems, with which the system will interact — by using their APIs (program interfaces), or by providing APIs to them, or both. These are interfaces provided to the system from the outside world. They are distinct from both: interfaces provided by the system to the outside world (S.3); and technology elements that the system's development will require (P.5).* [1]

The system will use different components from the environment and will provide components to the user to interact with the system.

**Secure Authentication System**: This component will only give the user access to the system if the user is involved in the field of Buddhist studies which includes professors, researchers, scholars, specialists, and students. The system will use the API provided by the authentication system to verify the user's credentials and grant access if their credentials are adequate.

**Fragment Management System**: This component will manage the storage, retrieval, and organization of text fragments. The system will interact with the fragment management system to store new fragments, retrieve existing ones, and organize them based on the user's preference.

**Segmentation Tool**: This component will assist users in segmenting fragments so that they can be combined into larger texts. The system will interact with the segmentation tool to provide users with the ability to segment fragments effectively.

# (E.3) Constraints

*Obligations and limits imposed on the project and system by the environment. This chapter defines non-negotiable restrictions coming from the environment (business rules, physical laws, engineering decisions), which the development will have to take into account.* [1]

The system will be bounded by several constraints which will restrict its design and implementation.

**Data Privacy Regulations**: The system must comply with relevant data privacy laws. This includes ensuring that user data is collected, stored, and processed in a manner that respects user privacy. Furthermore, the fragments must be stored securely to prevent unauthorized access.

**Performance Requirements**: The system must be capable of handling a large number of concurrent users without significant degradation in performance. This includes ensuring that response times for user

actions remain within acceptable limits, even under high load conditions.

**Security Standards**: The system must adhere to established security standards to protect against potential threats. This includes implementing robust authentication and authorization mechanisms to prevent access to unauthorized users.

# (E.4) Assumptions

*Properties of the environment that may be assumed, with the goal of facilitating the project and simplifying the system. It defines properties that are not imposed by the environment (like those in E.3) but assumed to hold, as an explicit decision meant to facilitate the system's construction.* [1]

The environment is assumed to have properties that will influence the design and implementation of the system.

**User Expertise**: The users of the system will have a basic understanding of Buddhist studies and familiarity with digital tools for text analysis. This assumption allows the system to be designed with a focus on advanced features without needing to provide extensive tutorials or guidance for basic usage.

**Dataset Availability**: The number of high-quality text fragments available for the system is large enough for classification. The system's functionality depends on this assumption, as it trains the model on a dataset of text fragments. Although it is a subset of all existing fragments, this data set is expected to represent all fragments properly so that the model will be able to process new fragments effectively.

**Technological Infrastructure**: The users will have access to modern computing devices and reliable internet connections. This assumption enables the system to implement advanced computational resources without being constrained by the limitations of users' hardware.

# (E.5) Effects

*Elements and properties of the environment that the system will affect. It defines effects of the system's operations on properties of the environment. Where the previous two categories (E.3, E.4) defined influences of the environment on the system, effects are influences in the reverse direction.* [1]

## Environmental Elements Affected by System Operation

The Sanskrit Manuscript Fragment Reconstruction Platform will have measurable effects on Buddhist scripture research and the intersection of modern technology with humanities academia.

## 1. Buddhist Studies Research Environment

**Buddhist Scripture Reconstruction**: Acceleration of text discovery and reconstruction timelines, enabling scholars to reassemble ancient Buddhist texts that have been fragmented for centuries. The systematic processing capabilities allow meaningful progress through the British Library's 21,000

fragments and similar collections worldwide.

**Research Methodology in Buddhist Studies**: Establishment of new computational approaches to paleographic research specifically for Buddhist manuscripts, with integration of AI-powered fragment matching into traditional scholarly methods. This creates hybrid research methodologies that maintain scholarly rigor while leveraging computational efficiency.

**Publication and Knowledge Recovery**: Increased volume of Buddhist manuscript reconstruction research publications due to accelerated discovery processes, potentially recovering lost Buddhist wisdom and bridging historical gaps in Buddhist literary tradition through systematic reconstruction.

### 2. Technology-Humanities Academic Integration

**Digital Humanities Evolution**: Buddhist Studies becomes a pioneering example of successful AI integration in humanities research, demonstrating how computational methods can augment rather than replace traditional scholarly expertise. This establishes new paradigms for technology adoption in humanities disciplines.

**Academic Research Culture Transformation**: Shift from individual, craft-based manuscript analysis to systematic, evidence-based reconstruction practices. Development of new forms of scholarly evidence that combine computational analysis with traditional philological and paleographic reasoning.

**Cross-Disciplinary Methodology Development**: Creation of transferable methodologies applicable to other manuscript reconstruction challenges in papyrology, epigraphy, and archaeological artifact analysis, establishing Buddhist manuscript reconstruction as a model for technology-assisted humanities research.

## (E.6) Invariants

> *Properties of the environment that the system's operation must preserve, i.e., properties of the environment that operations of the system may assume to hold when they start, and must maintain.* [1]

### Environmental Properties That Must Be Preserved

The Sanskrit Manuscript Fragment Reconstruction Platform operates as a non-deterministic AI assistance tool that augments scholarly work without making critical decisions. These invariants ensure the system maintains its supportive role while preserving essential academic and research environment properties.

### 1. Scholarly Authority and Decision-Making Invariants

**Human Scholar Final Authority**: All reconstruction conclusions and academic decisions must be made by qualified human scholars. The AI system provides suggestions and pattern matching assistance only, with scholars retaining complete authority over interpretation and validation of results.

**Non-Deterministic AI Acknowledgment**: The system operates with the understanding that AI-powered fragment matching is inherently non-deterministic and probabilistic. Results are presented as suggestions

requiring scholarly evaluation rather than definitive answers.

**Academic Independence**: System operation cannot substitute for scholarly judgment or critical analysis. Researchers must maintain their role as the primary interpreters of manuscript evidence and cultural context.

**Scholarly Evaluation Requirement**: All system outputs must be clearly presented as requiring academic evaluation and validation. The interface and documentation must emphasize the need for scholarly review of all AI-generated suggestions.

## 2. Data and Research Integrity Invariants

**Original Data Preservation**: Digital fragment images and metadata must remain unaltered by system processing. All analysis is performed on copies, ensuring original cultural heritage materials are never modified or compromised.

**Research Reproducibility**: While AI outputs may vary due to their non-deterministic nature, the underlying methodologies and data processing steps must be documented to support academic transparency and scholarly review.

**Attribution and Provenance**: Original fragment attribution, provenance information, and institutional ownership must be preserved and clearly displayed. The system cannot obscure the source or ownership of cultural heritage materials.

# (S) System

*the System book refines the Goal one by focusing on more detailed requirements.*

## Control Information

| Section | Version | Lead | Delivered | Reviewer | Approved |
|---|---|---|---|---|---|
| S.1 | | | | | |
| S.2 | | | | | |
| S.3 | | | | | |
| S.4 | | | | | |
| S.5 | 1.0 | Umar | October 2nd, 2025 | | |
| S.6 | | | | | |

## (S.1) Components

**Overall structure expressed by the list of major software and, if applicable, hardware parts.** [1]

The system consists of three main clusters that work together to enable scholars to reconstruct ancient Buddhist manuscript fragments through computational analysis and interactive manipulation.

### Front-end Cluster

- **User Authentication Interface**: Login and session management components for secure user access.
- **Fragment Upload Interface**: Components for uploading manuscript fragment images with support for batch uploads and image format validation.
- **Interactive Canvas Module**: A drag-and-drop workspace where users can display, move, rotate, zoom, and inspect one or multiple fragment images simultaneously.
- **Match Discovery Module**: Interface for triggering AI model analysis and viewing match suggestions for selected fragments.
- **Progress Management Module**: Tools for saving and loading user study sessions, including fragment arrangements, annotations, and match evaluations.

### Backend Cluster

- **API Gateway**: RESTful API endpoints for handling frontend requests and coordinating backend operations.
- **Authentication Service**: User authentication, authorization, and session management.
- **Image Processing Service**: Preprocessing pipeline for fragment normalization, orientation correction, and format standardization.

- **Database Access Layer**: Service layer managing interactions with all database systems.
- **Match Orchestration Service**: Coordinates requests to AI/ML models and aggregates results into ranked match suggestions.

### Data Storage Cluster

- **Fragment Image Database**: Stores high-resolution fragment images, preprocessed versions, and image metadata (resolution, format, upload date).
- **User Database**: Maintains user accounts, authentication credentials, preferences, and access permissions.
- **Project Database**: Tracks reconstruction projects, fragment arrangements, match history, confidence scores, user progress states, and study session snapshots.

## AI/ML Cluster

- **Edge Pattern Matching Model**: Analyzes fragment edge shapes and contours to identify physical matches where fragments may have been torn or broken from the same page. Generates edge signature vectors for comparison.
- **Handwriting Style Classifier**: Analyzes script characteristics, writing style, and paleographic features to group fragments by scribe or writing period. Outputs style vectors for similarity comparison.
- **Damage Pattern Recognition Model**: Identifies and characterizes damage signatures including tears, stains, creases, and deterioration patterns. Creates damage pattern fingerprints that indicate fragments exposed to similar conditions.
- **Text Extraction Model**: Extracts Sanskrit text from fragment images, handling various script styles and deterioration levels. Outputs transcribed text and character-level confidence scores.
- **Content Similarity Model**: Performs semantic analysis on extracted text to identify thematic and linguistic connections between fragments. Generates content embeddings for matching.

# (S.2) Functionality

***This is the bulk of the System book, describing elements of functionality (behaviors)**. This chapter corresponds to the traditional view of requirements as defining "**what the system does**". It is organized as one section, S.2.n, for each of the components identified in S.1, describing the corresponding behaviors (functional and non-functional properties).* [1]

# S.2.1 Front End

## Functional Properties:

**User Authentication**: The system will provide a secure authentication mechanism to ensure that only authorized users can access the system. This will involve integrating with an existing authentication system that verifies user credentials.

**Fragment Management**: The system will allow users to upload, store, retrieve, and organize text fragments. Users will be able to create collections of fragments and manage them through a user-friendly interface.

**Display Fragments**: The system will display all of the text fragments in a clear and organized manner, allowing users to view and navigate through their collections easily. It will allow users to view multiple fragments in the same display and provide tools for zooming, panning, and rotating fragments.

**Match Suggestions**: The system will suggest matches for text fragments based on the edge patterns, damage patterns, and content. Users will be able to view these suggestions and determine their validity. The system will provide an approximate probability score for each match suggestion.

**Session Management**: The system will allow users to save their current study sessions, including the arrangement of fragments, annotations, and any match evaluations they have made. Users will be able to load previous sessions to continue their work.

## Non-Functional Properties:

**Usability**: The front-end interface will be designed to be intuitive and user-friendly, ensuring that users can easily navigate and utilize the system's features without extensive training.

**Performance**: The front-end will be optimized for performance, ensuring that images load quickly and that user interactions are responsive, even when handling large collections of fragments.

# S.2.2 Back End

## Functional Properties:

**API Gateway**: The back-end will provide a RESTful API that serves as the primary interface for interactions between the front-end and the back-end. This API will handle requests related to user authentication, fragment management, match suggestions, and session management.

**Authentication Service**: The back-end will include an authentication service that integrates with the existing secure authentication system to verify user credentials and manage user sessions.

**Access Database**: The back-end will manage a user database that stores user accounts, authentication credentials, and access permissions.

## Non-Functional Properties:

**Scalability**: The back-end will be dynamically scaled based on the workload required by the system, allowing it to handle an increasing number of users and requests without significant performance degradation.

**Reliability**: The back-end will be designed with mechanisms in place for error handling, data integrity, and recovery from failures. The database will be backed up to avoid data loss if system failures were to occur.

# S.2.3 AI/ML Cluster

## Functional Properties:

**Edge Pattern Matching**: The system will include a ML model that analyzes fragment edge shapes to identify potential physical matches. This model will combine fragments with similar edges and output the combined fragment for the user.

**Handwriting Style Classification**: The system will include a ML model that analyzes script characteristics and writing styles to group fragments. This model will classify the fragments for comparison.

**Damage Pattern Recognition**: The system will include a ML model that identifies and characterizes damage signatures on fragments. This model will create damage pattern fingerprints to indicate fragments exposed to similar conditions.

## Non-Functional Properties:

**Accuracy**: The ML models will be trained with a training data set and then validated with a separate validation data set to ensure high accuracy. The model parameters will be adjusted to minimize error and will reduce false positives and false negatives in match suggestions.

**Efficiency**: The system will be using efficient classification models, which will ensure that match suggestions are generated quickly to reduce the wait time for users.

**Maintainability**: The ML models will be designed to allow for easy updates and retraining as new data becomes available, ensuring that the system can adapt to evolving requirements and datasets.

# (S.3) Interfaces

> *How the system makes the functionality of S.2 available to the rest of the world, particularly user interfaces and program interfaces (APIs). It specifies how that functionality will be made available to the rest of the world, including people (users) and other systems. These are interfaces provided by the system to the outside; the other way around, interfaces from other systems, which the system may use, are specified in E.2.* [1]

### Overview

The system will provide several interfaces to facilitate interaction between users and the system's functionality, as well as integration with other systems.

**User Interface**: The system will provide a web-based user interface that allows users to interact with the system's features. This interface will be accessible via modern web browsers and will be designed to be user-friendly and intuitive. It will include features for uploading, managing, and viewing text fragments, as well as tools for evaluating match suggestions and saving study sessions.

**API Interface**: The system will provide a RESTful API that allows external systems to interact with its functionality programmatically. This API will support operations for managing fragments, retrieving match suggestions, and accessing user session data. The API will be secured using authentication and authorization mechanisms to ensure that only authorized systems can access it.

**Fragment Upload Interface**: The system will provide an interface for uploading text fragments, supporting various image formats and batch uploads. This interface will include validation mechanisms to ensure that uploaded files meet the required specifications.

**Segmentation Tool Interface**: The system will provide an interface for integrating with the external segmentation tool. This interface will allow users to segment fragments effectively, facilitating the combination of smaller fragments into larger texts.

# (S.4) Detailed usage scenarios

*Examples of interaction between the environment (or human users) and the system, expressed as user stories. Such scenarios are not by themselves a substitute for precise descriptions of functionality (S.3), but provide an important complement by specifying cases that these behavior descriptions must support; they also serve as a basis for developing test cases. The scenarios most relevant for stakeholders are given in chapter G.5 in the Goals book, at a general level, as use cases; in contrast, S.4 can refer to system components and functionality (from other chapters of the System book) as well as special and erroneous cases, and introduce more specific scenarios.* [1]

In this section we take the use cases from section G.5 and give a detailed flow to them here.

# (S.4.1) Login

**Normal flow**

- The scholar navigates to the login page.
- The scholar enters a valid username and password.
- The system validates the credentials against the user database.
- Access is granted; the scholar is directed to the main dashboard.

**Alternate flow**

- If the scholar has no account, they click "Create Account" and complete registration (name, institutional email, password).
- If the scholar has forgotten a password, they click "Forgot Password" to receive a reset link by email.

**Error flow**

- If invalid credentials are entered, the system denies access and displays: "Invalid username or password. Please try again."
- After 5 failed attempts, the account is temporarily locked for 15 minutes.

## (S.4.2) Fragment Orientation and Preparation

**Normal flow**

- The scholar uploads one or more digitized manuscript images (JPG, PNG, TIFF).
- The scholar clicks "Normalize Images."
- The system cleans the images (crop, remove background noise, standardize resolution).
- The system regularizes fragment orientation (align text horizontally ±5°).
- The system notifies the scholar that the images are ready.

**Alternate flow**

- If metadata (e.g., EXIF orientation tags) is available, the system uses it to assist orientation.
- The scholar may manually adjust orientation if the automated correction is unsatisfactory.

**Error flow**

- If a file type is unsupported, the system rejects it and displays: "Unsupported format. Please upload JPG, PNG, or TIFF."
- If an image upload fails due to size limits, the system prompts: "File exceeds maximum size of 50 MB."

## (S.4.3) Fragment Matching and Reconstruction

**Normal flow**

- The scholar selects a fragment.
- The scholar requests "Find Matches."
- The system analyzes edges, shapes, and damage patterns.
- The system generates a ranked list of probable matches (with similarity scores).
- The scholar inspects the matches and accepts or rejects each suggestion.

**Alternate flow**

- The scholar may adjust sensitivity thresholds (e.g., only show matches above 0.8 probability).
- The scholar may compare two fragments side by side.

**Error flow**

- If no probable matches are found, the system displays: "No strong matches detected. Try lowering the match threshold."
- If processing fails due to corrupted images, the system flags them and skips analysis.

## (S.4.4) Script Identification

**Normal flow**

- The scholar views a fragment.
- The system analyzes visible text features.
- The system classifies the fragment's script (e.g., Gupta Brahmi, Tibetan Uchen, Chinese).
- The classification result and confidence score are displayed.

**Alternate flow**

- The scholar may override the system's classification with a manual label.

**Error flow**

- If the system cannot classify with ≥50% confidence, it displays: "Script could not be determined. Manual input required."

## (S.4.5) Transcription Assistance

**Normal flow**

- The scholar opens a fragment.
- The scholar requests transcription.
- The system applies OCR tuned for ancient manuscripts.
- The transcription is displayed alongside the fragment image.
- The scholar edits and annotates the transcription.

**Alternate flow**

- The scholar may export the transcription as text or XML.
- The scholar may request partial transcription (e.g., line by line).

**Error flow**

- If OCR confidence is <70%, the system highlights uncertain characters in red.
- If OCR fails entirely, the system prompts: "Unable to transcribe. Please attempt manual transcription."

## (S.4.6) Database Search and Exploration

**Normal flow**

- The scholar opens the database search interface.
- The scholar enters a query (e.g., "Gupta script, 5+ lines").
- The system retrieves and displays relevant fragments.
- The scholar sorts, filters, and links fragments as needed.

**Alternate flow**

- The scholar may export results to CSV or PDF.
- The scholar may save queries for future use.

**Error flow**

- If the query is invalid, the system prompts: "Invalid search format. Please correct your query."
- If no results match, the system displays: "No fragments found matching your criteria."

# (S.5) Prioritization

*Classification of the behaviors, interfaces and scenarios (S.2, S.3 and S.4) by their degree of criticality. It is useful in particular if during the course of the project various pressures force the team to drop certain functions.* [1]

## Critical (Must-Have)

- Secure login and authentication (S.4.1)
  - required for system access and user management.
- Fragment upload, orientation, and normalization (S.4.2)
  - enables the core workflow of preparing digitized manuscripts.
- Fragment matching and reconstruction (S.4.3)
  - essential for achieving the project's scholarly goals.
- Script identification (S.4.4)
  - foundational to understanding and categorizing fragments.
- Database search and exploration (S.4.6)
  - critical for enabling scholars to retrieve and compare fragments efficiently.

## Important (Should-Have)

- Transcription assistance (S.4.5)
  - significantly enhances usability and research speed, but manual transcription remains possible if OCR fails.
- Match sensitivity tuning and side-by-side comparison (alternate flow of S.4.3)
  - improves researcher control and reliability of results.
- Manual overrides for classification and orientation
  - ensures system flexibility where automation falls short.
- Exporting search results and transcriptions
  - supports collaboration and documentation.

## Optional (Nice-to-Have)

- Session management (save, load, version history)
  - valuable for long-term projects but not mandatory for demonstrating proof-of-concept.
- Annotation features on fragments
  - enhances usability but not required for minimal viable system.
- Integration with external segmentation tools
  - useful for advanced workflows but not critical for initial delivery.

- Read-only sharing links
  - convenient for collaboration but not essential for project success.

# (S.6) Verification and acceptance criteria

> *Specification of the conditions under which an implementation will be deemed satisfactory. Here, "verification" as shorthand for what is more explicitly called "Verification & Validation" (V&V), covering several levels of testing — module testing, integration testing, system testing, user acceptance testing — as well as other techniques such as static analysis and, when applicable, program proving.* [1]

## Overview

This section specifies conditions under which the Sanskrit Manuscript Fragment Reconstruction Platform will be deemed satisfactory for academic research use. The V&V framework focuses on core functionality while recognizing the platform serves as a non-critical AI assistance tool for scholarly research.

## 1. Functional Verification Criteria

**Fragment Processing Pipeline**

- **Image Segmentation and Measurement**: System processes British Library collection and user-submitted images with consistent segmentation and size measurement. Success threshold: ≥85% successful processing across diverse image conditions.
- **Metadata Extraction**: System extracts script type, row count, and codicological features for filtering. Success threshold: ≥75% accuracy for primary metadata attributes.

**Search and Filter Interface**

- **Multi-Criteria Search Performance**: Fragment ID lookup and filter combinations perform responsively. Success threshold: ≤3 seconds response time for searches across 21,000-fragment collection.
- **Canvas Workspace Functionality**: Drag-and-drop interface operates smoothly with session save/restore capability. Success threshold: Responsive manipulation (≤200ms) with reliable session management.

## 2. Non-Functional Verification Criteria

**System Performance**

- **Collection Handling**: System processes 21,000-fragment British Library collection and supports concurrent research use. Success threshold: ≤48 hours initial processing; supports 15+ concurrent users.
- **AI Matching Assistance**: Physical compatibility suggestions demonstrate utility for scholarly evaluation. Success threshold: ≥60% of expert users find matching suggestions useful for research workflow.

**Usability and Reliability**

- **Research Workflow Usability**: Scholars can effectively use core features (search, filter, canvas) for manuscript reconstruction research. Success threshold: ≥80% task completion rate in user testing.
- **System Availability**: Platform maintains consistent availability for academic research. Success threshold: ≥95% uptime during operational hours.

## 3. Academic Research Acceptance Criteria

**Research Tool Validation**

- **Scholarly Utility**: Platform demonstrates value as an AI assistance tool for Buddhist manuscript reconstruction research. Success threshold: Positive feedback from ≥3 pilot research projects.
- **Data Integrity**: British Library collection data and user-submitted images are handled appropriately with preserved attribution. Success threshold: 100% preservation of original provenance and catalog information.

## 4. Testing and Validation Framework

**Testing Approach**: Unit testing for image processing, integration testing for search functionality, system testing for complete workflows, and user acceptance testing with Buddhist Studies researchers.

**Acceptance Decision**: Platform ready for deployment when core functionality (processing, search, canvas interface) operates reliably, British Library collection is accessible, and expert users validate research utility. Recognition that AI suggestions are non-deterministic and require scholarly evaluation.

# (P) Project

## Control Information

| Section | Version | Lead | Delivered | Reviewer | Approved |
|---|---|---|---|---|---|
| P.1 | | | | | |
| P.2 | | | | | |
| P.3 | | | | | |
| P.4 | | | | | |
| P.5 | 1.0 | Umar | October 2nd, 2025 | | |
| P.6 | 1.0 | Umar | October 2nd, 2025 | | |
| P.7 | 1.0 | Umar | October 2nd, 2025 | | |

## (P.1) Roles and personnel

*Main responsibilities in the project; required project staff and their needed qualifications. It defines the roles (as a human responsibility) involved in the project.* [1]

**Team:**

- Five software engineering students from McMaster University.
- Rotating biweekly roles: Leader, Organizer, Meeting Chair, Note Taker, Reviewer.
- 10 hours per week per member.

**External Personnel:**

- Dr. Shayne Clarke (domain expert, biweekly consultations).
- Course instructor and teaching assistants.

**Constraints:**

- All personnel are full-time students with concurrent academic obligations.
- The team's limited expertise in Sanskrit text and Buddhist manuscripts.
- Limited time for completing the project.

## (P.2) Imposed technical choices

*Any a priori choices binding the project to specific tools, hardware, languages or other technical parameters. Not all technical choices in projects derive from a pure technical analysis; some result from company policies. While some project members may dislike non-strictly-technical decisions, they are a fact of project life and must be documented, in*

> *particular for the benefit of one of the quality factors for requirements: "requirements must be justified".* [1]

## Mandatory Constraints

**Licensing:**

- General Public License (GPL) must be used for all project code.

**Version Control:**

- GitHub for version control and project management.
- Feature-branch workflow with pull request reviews.

**Programming Languages:**

- Python: ML models and image processing.
- JavaScript/TypeScript: Front-end and back-end development.

## Technical standards:

**Code Quality:**

- All code changes require pull request review and approval before merging.
- Linting and style checking must pass before merge approval.

**Build Automation:**

- Continuous integration pipeline required.

**Documentation:**

- LaTeX for formal project documentation.

# (P.3) Schedule and milestones

> *List of tasks to be carried out and their scheduling. It defines the project's key dates.* [1]

## Major Milestones

- Proof of Concept: Demonstrate technical feasibility of core system capabilities
- Revision 0: First complete system implementation.
- Revision 1: Refined system based on testing and feedback.
- Final Delivery: Complete system with full documentation.

## Course Deliverables

- Team Formation and Project Selection
- Problem Statement, POC Plan, Development Plan
- Requirements Documentation and Hazard Analysis Revision 0
- Verification & Validation Plan Revision 0
- Design Document Revision -1
- Proof of Concept Demonstration
- Design Document Revision 0
- Revision 0 Demonstration
- V&V Report and Extras Revision 0
- Final Demonstration (Revision 1)
- EXPO Demonstration
- Final Documentation (Revision 1)

## Meeting Schedule

**Team Meetings:**

- Weekly meetings required (in-person preferred, virtual if necessary)

**Domain Expert Consultations:**

- Biweekly meetings scheduled in advance
- Duration: 30-60 minutes
- Additional meetings available as needed

# (P.4) Tasks and deliverables

***This is the core of the Project book**. It details the individual tasks listed under P.3 and their expected outcomes. It define the project's main activities and the results they must produce, associated with the milestone dates defined in P.3.* [1]

## Team Formation and Project Selection

**Tasks:**

- Form team of five members
- Select project from available options
- Establish initial contact with domain expert

**Deliverables:**

- Form the team.
- Confirm selected project

## Problem Statement, POC Plan, Development Plan

**Tasks:**

- Define project scope and objectives
- Identify technical risks and mitigation strategies
- Establish development workflow and standards
- Create POC demonstration plan

**Deliverables:**

- Problem statement document
- POC plan document
- Development plan document (this document)

## Requirements Documentation and Hazard Analysis Revision 0

**Tasks:**

- Define functional and non-functional requirements
- Identify potential hazards and safety concerns
- Document system constraints

**Deliverables:**

- Requirements specification document
- Hazard analysis document

## Verification & Validation Plan Revision 0

**Tasks:**

- Define testing strategy and acceptance criteria
- Establish verification methods for requirements
- Plan usability testing approach

**Deliverables:**

- V&V plan document

## Design Document Revision -1

**Tasks:**

- Create preliminary system architecture
- Define component interfaces
- Plan database schema

**Deliverables:**

- Initial design document

## Proof of Concept Demonstration

**Tasks:**

- Implement OCR text extraction from manuscript fragments
- Develop image normalization pipeline (rotation correction, standardization)
- Create feature extraction for edges, text content, and damage patterns
- Implement similarity algorithms with confidence scores
- Build basic web interface for fragment upload and viewing results

**Deliverables:**

- Functional POC demonstrating:
  - 60% OCR text extraction accuracy
  - 95% orientation correction success
  - Feature extraction of edge patterns, damage signatures, and text regions
  - Consistent similarity rankings
- POC demonstration presentation

## Design Document Revision 0

**Tasks:**

- Refine architecture based on POC learnings
- Complete detailed component specifications
- Document API contracts

**Deliverables:**

- Complete design document

## Revision 0 Demonstration

**Tasks:**

- Implement complete frontend (multi-fragment canvas, match discovery, save/load, search/filter, export)
- Build full backend (database operations, model orchestration, async processing, caching)
- Deploy all five AI models (70% OCR accuracy, 70% top-5 match precision)
- Achieve 80% test coverage for frontend and backend
- Conduct usability testing with domain expert

**Deliverables:**

- Fully integrated system
- Revision 0 demonstration presentation

## V&V Report and Extras Revision 0

**Tasks:**

- Execute test plans
- Document test results and coverage metrics
- Report usability findings

**Deliverables:**

- V&V report
- Test coverage reports

## Final Demonstration (Revision 1)

**Tasks:**

- Address issues identified in Revision 0
- Optimize system performance
- Enhance error handling and user feedback

**Deliverables:**

- Production-ready system
- Final demonstration presentation

## EXPO Demonstration

**Tasks:**

- Prepare public-facing demonstration materials
- Create poster and presentation materials

**Deliverables:**

- EXPO poster
- EXPO demonstration

## Final Documentation (Revision 1)

**Tasks:**

- Finalize all technical documentation
- Complete user manual
- Generate final API documentation

- Compile source code repository

**Deliverables:**

- Final requirements document
- Final design document
- Final V&V report
- User manual
- Technical documentation
- Source code repository with README

# (P.5) Required technology elements

*External systems, hardware and software, expected to be necessary for building the system. It lists external technology elements, such as program libraries and hardware devices, that the project is expected to require. Although the actual use of such products belongs to design and implementation rather than requirements, it is part of the requirements task to identify elements whose availability is critical to the success of the project — an important element of risk analysis (P.6).* [1]

The following external technology elements are required for the successful development and deployment of the Sanskrit Cipher system:

## Programming Languages

- **Python** for machine learning models, image preprocessing, and backend pipelines.
- **JavaScript/TypeScript** for frontend and backend development.
- **C++** may be required for performance-critical modules.

## Libraries & Frameworks

- **OpenCV** for image preprocessing and orientation correction.
- **PyTorch / TensorFlow** for machine learning models, including OCR and script identification.
- **scikit-learn** for probabilistic matching algorithms.
- **SQLAlchemy** for database abstraction.
- **React** for frontend development.
- **Tailwind CSS** for frontend styling.
- **Flask, FastAPI, or Django** as the backend API framework.

## Pre-trained Models

- OCR models trained on Sanskrit or related Indic scripts.
- Potential use of lightweight custom models if pre-trained solutions prove insufficient.

### Development Tools

- **VSCode** as the main IDE.
- **LaTeX/Markdown** for documentation.
- **Doxygen** for API documentation.
- **Postman** for API testing.

### Testing & Code Quality Tools

- **Pytest** and **coverage.py** for backend testing and coverage.
- **MyPy** and **Ruff** for Python type checking and linting.
- **Jest** with React Testing Library for frontend testing.
- **ESLint** and **Prettier** for frontend linting and formatting.
- **Valgrind** for C++ modules, if used.

### Build & Automation

- **Makefiles** for automated builds of modules and LaTeX documents.
- **GitHub Actions** for CI/CD, including linting, testing, Docker builds, and PDF generation.

### Deployment & Infrastructure

- **Docker** for containerized environments.
- **GitHub Projects** for version control and project management.
- **PostgreSQL or SQLite** as the database backend.

## (P.6) Risk and mitigation analysis

*Potential obstacles to meeting the schedule of P.4, and measures for adapting the plan if they do arise. It is essential to be on the lookout for events that could derail the project, and devise mitigation strategies. It can include a SWOT analysis (Strengths, Weaknesses, Opportunities, Threats) for the project.* [1]

### Technical Risks

- OCR accuracy on damaged Sanskrit fragments may remain below acceptable thresholds.
  - Mitigation: Evaluate existing OCR models or fine-tune lightweight models on curated samples. Integrate fallbacks like similarity-based matching when OCR fails.
- Integration of multiple models and pipelines could cause performance bottlenecks.
  - Mitigation: Modular architecture with asynchronous execution and caching. Early profiling to identify bottlenecks.
- Image preprocessing may fail for severely degraded or rotated fragments.
  - Mitigation: Implement multi-stage preprocessing (denoising, rotation correction, cropping). Validate on a diverse set of fragments during POC.

## Organizational Risks

- Team workload may conflict with other academic obligations.
  - Mitigation: Task decomposition into small, reviewable units. Buffer periods built into schedule. Weekly progress tracking with GitHub Projects.
- Limited domain expertise in Sanskrit manuscripts within the team.
  - Mitigation: Weekly consultations with Dr. Shayne Clarke (project supervisor). Iterative feedback cycles after each milestone demonstration.

## External Risks

- Hardware and resource constraints may limit model training and testing.
  - Mitigation: Prioritize lightweight models and use batching strategies during development. If larger-scale training or evaluation is required, request access to the clusters provided by the Department of Computing and Software at McMaster University.
- Stakeholder engagement may be restricted by limited availability of the domain expert.
  - Mitigation: Schedule consultations well in advance, prepare concise agendas, and use scholarly publications when direct input is not available.

## SWOT Analysis

- Strengths
  - Multidisciplinary skillset across ML, image processing, and full-stack development.
  - Strong project management practices (CI/CD, GitHub Projects, PR reviews).
- Weaknesses
  - Limited prior exposure to Sanskrit OCR and palaeography.
  - Heavy reliance on external resources (pre-trained models, domain expert, etc.).
- Opportunities
  - Develop a reusable research tool for manuscript studies.
  - Potential collaboration with scholars and institutions holding large fragment collections.
- Threats
  - OCR accuracy may not meet scholarly expectations.
  - Project timeline is constrained by academic calendar deadlines.
  - Risk of overfitting to small or curated test datasets.

# (P.7) Requirements process and report

*Initially, description of what the requirements process will be; later, report on its steps. It starts out as a plan for conducting the requirements elicitation process, but is meant to be updated as part of that process so that it includes the key lessons of elicitation.* [1]

## Planned Process

- We plan on using a couple of methods in order to elicit our requirements.
  - Semi-structured interviews (primary): 4–5 researchers, 20–30 minutes each, in-person or video

call. Focused on pain points, workflows, and domain challenges.
    - Online survey (secondary): 25–30 academic community members. Short questionnaire to validate findings quantitatively.
    - Observational study (supplementary): 3–4 researchers observed for 1–2 hours to identify implicit behaviours and actual practices.
    - Structured interviews (excluded): Not chosen due to insufficient domain knowledge for effective closed questions.

## Target User Groups

- Primary users
    - Buddhist studies professors and researchers.
    - Digital humanities scholars.
    - Manuscript specialists and palaeographers.
- Secondary users
    - Graduate students working with ancient manuscripts and reconstruction tasks.

## Objectives

- Understand current manual workflows and pain points in fragment reconstruction.
- Identify functional needs for automated detection, matching, and transcription support.
- Capture usability and workflow-integration requirements for academic research settings.
- Establish expectations for accuracy, performance, and collaboration/sharing features.

## Instruments & Protocols

- Interview guide
    - Recent tool usage (what/why/learning curve), challenges, desired features.
    - End-to-end walkthrough of receiving and processing a new fragment set.
    - Tracking/organization practices; progress management; collaboration needs.
- Survey
    - Short, closed-ended items to validate priority features and pain points at scale.
- Observation
    - Note actual vs. reported behaviours; tools used; environment constraints; implicit steps.
- Recording & consent
    - Obtain consent for audio recording during interviews; anonymize notes for analysis.

## Reporting and Updates

This section will be updated iteratively as elicitation is carried out:

- After interviews: Summarize recurring themes (pain points, desired features, implicit needs).
- After survey analysis: Report statistical validation of key requirements.
- After observational studies: Note discrepancies between reported and actual workflows.

The evolving results will be documented in the SRS, ensuring traceability between user needs and formal

requirements.

# References


- [1] Bertrand Meyer. *Handbook of Requirements and Business Analysis*. Springer. 2022.
- [2] Ian Sommerville and Peter Sawyer. *Requirements Engineering: A good Practice Guide*. Wiley. 1997.