KudoSpace

Software Requirements Specification (SRS)

Version #2
Prepared By: Mariah Galli, Meryem Salih, Alek White
Prepared For SUNY Oswego
Fall 2025

Table of Contents

Change Log

Name	Date	Reason For Changes
Galli, Salih, White	9/19/25	Introduction and Home Page Requirements.
Alek White	9/28/25	Added to Sections 1, 2, and 3, added card creation page requirements.
Requirements Team	10/30/25	Added new requirements.
Requirements Team	11/15/25	Updates to requirements
Requirements Team	11/20/25	Added new requirements.
Requirements Team	12/03/25	Finishing the remaining sections.

1. Introduction

This document specifies the requirements for a classroom-based digital recognition system. The system provides a safe, structured process for students to exchange kudos cards, which are short messages of acknowledgement and encouragement. Instructors will be able to review and approve the created kudos cards before they are delivered to the recipient.

1.1 Purpose

The system's primary objective is to facilitate the sharing of digital kudos cards by automating the process and providing an efficient and accessible interface. It will allow students to send short messages to share feedback, recognize their peers' work, and motivate their classmates. This service is intended for use in an upper-level college course and will also help the course instructor(s) gauge student performance and participation.

The current solution requires the instructor(s) to use multiple websites and spend significant time reviewing and sending out review cards. This solution aims to simplify and enhance the overall process.

1.2 Scope

This SRS document primarily outlines the requirements for developing the KudoSpace system, including functional, usability, performance, non-functional, user requirements, and constraints.

1.2.1 Document Scope:

- The document will include functional requirements (what the system should do).
- The document will include non-functional requirements (such as usability, performance, and reliability).
- The document will describe user requirements, focused on how students and instructors interact with the system.
- The document will provide a product overview to explain how the system fits into the classroom environment.
- The scope does not include developing a mobile app or a special mobile-only version of the system.

1.2.2 System Scope:

- The system will be available to students and instructors who are currently enrolled in a course.
- The system will store each card a student receives, along with statistics regarding student activity, in a persistent database.

- The system will deliver notifications via email.
- The system will provide a web interface for students to generate cards to be shared.
- The system will provide a web interface for the instructor(s) to approve each card and monitor student activity.

1.3 Product Overview

1.3.1 Product Perspective

The tool is designed to replace the existing system of sharing Kudo cards by consolidating the process into a dedicated platform. It operates in a closed environment and is only available to the students and instructors of a single course. This service is a standalone product built with standard software components and is not an extension of an existing product.

1.3.1.1 System Interfaces

- The system shall use an external email service to send all email notifications regarding newly received cards and changes in approval request status.
- The system's web application shall interface with its persistent database program to provide for cards, users, statuses, and statistics.
- The system will use Google's OAuth service for all authentication functions.

1.3.1.2 User Interfaces

Users will be able to access the system's web interface using a standard modern web browser. Once successfully logged in, users are provided with an interface that allows them to complete all kudo-related functions in the browser. The desktop web interface will follow standard usability and accessibility principles and must be WCAG 2.0 Compliant. Users can also interact with the system by viewing all their notifications through email.

1.3.1.3 Hardware Interfaces

The system is primarily a software-based application and will not interface with devices other than the host server and client devices. The web application must be designed for desktop web browsers, not mobile clients. The web application must be available on any operating system a user is utilizing.

1.3.1.4 Software Interfaces

- Open Liberty: Used to facilitate the development and deployment of a microservices-based system
- SQL Database: Stores all necessary data persistently.
- Google OAuth: user authentication
- Microsoft Azure: application deployment and hosting.

1.3.1.5 Communication Interfaces

Internal communication between front-end and back-end components will occur using RESTful APIs. External communication between the system and the user's web browser will take place using HTTP/HTTPS protocols.

1.3.1.6 Memory Constraints

The system has no applicable memory constraints.

1.3.1.7 Site Adaptation

The system does not require any site adaptations.

1.3.1.8 Interfaces with Services

The system has no applicable interfaces.

1.3.2 Product Functions

- Allows users to create, send, and receive cards.
- Notifies users when they receive new cards via email (Stretch Goal).
- Allows instructors to approve/reject all cards.
- Persistently stores all cards indefinitely.
- Generates and displays stored kudos cards as images.
- Calculates and displays participation-related statistics to the instructor(s) (Stretch Goal).

1.3.3 User Characteristics

The intended user group is college-level students and college professors. Although initially designed for a Computer Science and Human Computer Interaction class, users can be students in any class. It is expected that users are familiar with modern websites and applications. Users can possess a wide range of skills, technical expertise, and accessibility needs. The application needs to cater to a diverse range of users and offer accessibility tools and features, allowing them to utilize it effectively.

The primary concern of users of this system is the expediency of the card creation and approval process. When creating a card, the users are not expected to have any advanced knowledge of graphic design principles. This process should not overburden the users with creative choices; instead, it should minimize the users' creative decision-making and lead to a faster overall process. Students expect the process of creating and sending Kudos to require minimal time, with most of the effort being devoted to writing the message itself.

Instructors are primarily concerned with the card approval process, since the system cannot send cards without their manual approval. The instructors have expressed interest in reducing the time they spend processing approvals.

1.3.4 Limitations

- The system will only be accessible through a web browser; no mobile app or dedicated mobile interface is included in the current scope.
- The system depends on a stable internet connection; offline use is not supported.
- The system will store messages and activity in a database, and long-term archival beyond one semester is required.

1.4 Definitions

Term	Definition
Kudos Card / Card	A stylized digital image containing a short message of encouragement to be sent between students.
Card Approval Request	A request from a student to an instructor to review the content of the given card before it is sent to the recipient.
Classroom / Class	A collection of students, belonging to the same course at the same time. Students can only send and receive cards from students and instructors in the same class as them.
Card Design / Card Template	A selectable background image used to generate a card.
Join-Code	A six digit code such as "123456" that can be distributed by the instructor to their students, allowing students with the code to request enrollment in the associated class.

2. References

- WCAG Accessibility Guidelines
- Wireframes

3. Requirements

Each requirement is organized into one of five sections: functions, performance, usability, interface, and database requirements. The IDs start with an abbreviation that corresponds to the sprint the requirement is associated with (HP-Sprint2: Homepage, CC-Sprint3: Card Creation).

3.1 Functions

ID	Natural Language Requirement	Requirement Type	Source
HP.5	The homepage shall provide users access to a notification tray, showing the count of unread notifications for that user.	Functional	Client Interview
HP.6	The instructor(s) shall be provided access to approve or reject a pending card from their home page.	Functional	Instructor Interview
HP.8	There will be “Filter” and “Sort” options for the card inboxes on the homepage.	Functional	Client Interview
HP.G3	The system must ensure users have successfully logged into their account before displaying that user’s home page.	Functional	Requirements Team
CC.4	The system shall generate and display a preview of the Kudos card on the card creation page before the submission step.	Functional	Client Interview
CC.5	The CC page must provide the user with a button labeled "Submit”, which, when pressed, the system shall check that the user has filled in all required fields.	Functional	Client Interview
CC.6	The system shall allow users to confirm, discard, or edit the Kudos after previewing the Kudos card through a confirmation menu.	Functional	Client Interview

CC.G1	The Kudos creation process must take place within a dedicated page.	Functional	Client Interview
AC.1	The system must provide a single account creation process for students and instructors.	Functional	Usability
AC.2	The account creation process should have fields for Display Name and Role (Student/Instructor).	Functional	Usability
LP.1	The system must provide a single login page to authenticate users.	Functional	Usability
LP.2	The system must use Google's OAuth service for all authentication functions.	Functional	Full Stack
CM.1	The Course Management Page shall provide instructors with the ability to view, create, edit, and delete courses.	Functional	Client Interview
CM.2	The course creation process will generate a join code for each course. The join codes will allow users to request enrollment in a given class.	Functional	Client Interview
CM.3	The system must allow users to request enrollment in a course by using a valid join code.	Functional	Client Interview
CM.4	The course management page must require instructors to manually approve a user's enrollment request before that user is added to the course.	Functional	Client Interview
CM.5	The system must automatically archive courses once the class end date is reached.	Functional	Client Interview

3.2 Performance Requirements

ID	**Natural Language Requirement**	**Requirement Type**	**Source**
PR-1	The system shall allow users to create and send a Kudos card at an appropriate time after submission.	Performance	Usability Standard
PR-2	The system shall support up to 20 concurrent users without performance degradation.	Performance	Anticipated Class

			Size
PR-3	The system shall handle # Kudos cards per class per week without exceeding database limits.	Performance	Expected Usage Data

3.3 Usability Requirements

ID	**Natural Language Requirement**	**Requirement Type**	**Source**
CC.7a	The templates shall be offered in a variety of visual styles and colors, and will have a standardized image size and font type.	Usability	Usability Demo
CC.7b	Each template must have ruled lines on which the message body text is placed on. Ruled lines must be standardized across templates.	Usability	Usability Demo
CC.G3	The CC page must have a button to return to the homepage, which discards the user's selections when pressed.	Usability	Client Interview

3.4 Interface Requirements

ID	**Natural Language Requirement**	**Requirement Type**	**Source**
HP.1	The user's home page shall display a clickable thumbnail preview for each card received by that user.	Functional	Client Interview
HP.2	The homepage for the instructor shall display all pending card submissions for the active class.	Functional	Instructor Interview
HP.3	The user's home page shall display a clickable thumbnail preview for each card submitted by that user, which includes its status (Approved, Denied, Pending Approval, or Received).	Functional	Client Interview

HP.4	The Navigation Menu shall be available at the top of all pages, and will allow the user to navigate to any page available to that user. (Home Page and Generate Kudos, Sign-in / Sign-out, Instructor Statistics Page).	Functional	Client Interview
HP.7	On the home page, in the user's Inbox and Outbox, newly updated or received Kudos Cards shall be visually distinct from previously viewed Kudos Cards.	Usability	Usability team
HP.9	The user's home page shall display the numeric values for the total number of cards received and the total number sent for that user using separate labels.	Functional	Client Interview
CC.G2	The CC page shall only provide input fields for: Message, Recipient, Card Design, and an Anonymous checkbox.	Functional	Client Interview
CC.1	The CC page shall provide a text box for entering the Kudos message with a limit of 250 characters.	Usability	Client Interview / Usability Testing
CC.2	The system shall provide users with a selection menu to choose a card design.	Functional	Client Interview
CC.3	The system shall provide a dropdown menu to select recipients from all users in the active class, including all instructors.	Functional	Client Interview
CM.1.1	The CM interface will allow instructors to view all classes they created and to see a roster of all users in each class.	Functional	Client Interview

3.5 Logical Database Requirements

ID	**Natural Language Requirement**	**Requirement Type**	**Source**
HP.G1	Created Kudo cards should not be stored as image files; the system shall reconstruct each card from the information in the database when needed.	Constraint	Client Interview

HP.G2	The system must store all cards in the database indefinitely.	Database	Instructor Interview
AC.3	All users in the system should have a unique email address.	Non-Functional	Requirements Team

4. Verification

4.1 Functions

ID	Verification Process
HP.5	- Verify that the system uses a button to provide access to an expanded notifications pop-up menu. - Verify that the notification menu contains all notifications the user has received in the last month. - Verify that the notifications are displayed in order, with the most recently received notifications on top. - Verify that the notification menu will indicate which notifications are unread.
HP.6	- Verify that when the Instructor selects a Kudos Card, the options to “Approve” or “Reject” are listed. - Verify that when “Reject” is selected, the rejection section becomes available. There will be a field for the Instructor to specify a Rejection Reason. - Verify that the reason for rejection section shall include: a drop-down menu with predefined reasons, and a comment text box (with a limit of 250 characters) for additional notes. - Verify that if “Other” is selected from the dropdown, the comment text box becomes required before submission. - Verify that when approving, the card status updates to “Approved” for the student without a refresh. - Verify that when a card is rejected, the student's card status updates to “Denied” without a refresh. - Verify that after an instructor approves or rejects a Kudos card in the “Submitted” section, the Kudos will move to the “Reviewed” Kudos section on their home page.
HP.8	- Verify that the “Received Kudos” and “Submitted Kudos” have a “Sort” button. - Verify that the “Sent Kudos” and “Reviewed Kudos” have both “Sort” and “Filter” buttons. - Verify that the “Sort” options reorder cards in either ascending or descending order according to the selected criteria. - Verify that the “Filter” options only show cards with the specified criteria.
HP.G3	- Verify that any page accessed requires a valid login to be displayed. - Verify that, when not logged in, the system will navigate to the Logon Page.

CC.4	- Verify that the preview displays a fully readable image of the card as the recipient will see it. It displays the full message body, which is overlayed onto the selected card design. - Verify that the preview updates dynamically when users change the card's design or message body. - Verify that the text will automatically reformat to fit properly on the card, ensuring it is readable and each word fits entirely on one line.
CC.5	- Verify that all required fields (Message and Recipient) must be completed before proceeding to the confirmation step. - Verify that all non-required fields (card design) must have a default option pre-selected. - Verify that after a failed submission attempt, the user shall be informed about the error, including the possible cause and solution.
CC.6	- Verify that the CC page shall display a preview of the created card, featuring the user's written message on the chosen card style. - Verify that "Confirm" submits the Kudos card for approval by an instructor and then redirects the user to their homepage. The Kudos card will appear in the instructor's "Submitted Kudos" list, and the Kudos card will appear in the student's "Sent Kudos" list. - Verify that when an Instructor creates a new Kudos and selects "Confirm," the card is automatically added to the instructor's sent list and the recipient's received list. Instructor-created cards do not require approval. - Verify that "Discard" discards the card entirely, then redirects the user to their homepage.
CC.G1	- Verify that the Kudos creation process occurs on a dedicated page.
AC.1	- Verify that after successful account creation, users should be automatically logged in and redirected to their homepage. - Verify that new users will not be members of any classes when their accounts are created.
AC.2	- Verify that the system must add a new user record to the database when a new account is successfully created. - Verify that all unsubmitted data is discarded when a user exits the AC process. - Verify that the system allows users to set a display name different from the name associated with their Google account.
LP.1	- Verify that students and instructors are redirected to their homepage after a successful login. - Verify that only authenticated users can access data associated with their account.

LP.2	- Verify that the system uses Google's OAuth service for all authentication functions.
CM.1	- Verify that he course management page and all functions on it are only available to instructors.
CM.2	- Verify that the code expires after the class's end date and does not work past that date. - Verify that each class only has one active join code at a time. - Verify that the system requires users to provide a join code only when they first enroll in a class; once enrolled, the join code is no longer needed.
CM.3	- Verify that the system must provide access to a pop-up for instructors and students to enter a join code. - Verify that the system must validate the join code and ensure it has not expired or is invalid. - Verify that if an expired or invalid code is entered, the system outputs an error.
CM.4	- Verify that all pending enrollment requests must be displayed, and either approved or denied within the instructor's course management page. - Verify that a user's enrollment request must be approved for that course to appear in the course selection dropdown menu on the card creation page. - Verify that only approved users can appear in the recipients dropdown menu on the card creation page for that class.
CM.5	- Verify that when the end date passes, the class is considered to be "Archived" by the system. - Verify that archived courses must not appear in the course selection dropdown menu on the card creation page. - Verify that students and professors can only view old cards; new cards cannot be sent to archived classes. - Verify that users cannot be added or removed from an archived class. - Verify that archived classes cannot be edited or un-archived; they can only be deleted.

4.2 Performance Requirements

ID	**Verification Process**
PR.1	- Verify that the system shall allow users to create and send a Kudos card at an appropriate time after submission.
PR.2	- Verify that the system supports up to 20 concurrent users without degrading performance.

PR.3	- Verify that the system shall handle # Kudos cards per class per week without exceeding database limits.

4.3 Usability Requirements

ID	Verification Process
CC.7a	- Verify that each card image uses the same font style for the message body. - Verify that each template must have a unique, clearly visible title embedded in the design. - Verify that each card image and template will not display any information pertaining to the sender, recipient, or the date/time sent.
CC.7b	- Verify that each template must have ruled lines on which the message body text is placed. Ruled lines must be standardized across templates.
CC.G3	- Verify that the CC page must have a button to return to the homepage, which discards the user's selections when pressed.

4.4 Interface Requirements

ID	Verification Process
HP.1	- Verify that when a user has not received any cards, the system displays a message such as "No Cards Received" when checking the inbox. - Verify that when clicking on a preview, the page displays that entire card in a large view centered on the screen. - Verify that the card previews are displayed in order by default, with the most recently received card being first. - Verify that when a user receives a new card, its preview is displayed on the home page without requiring a manual refresh.
HP.2	- Verify that if no pending cards, display "No cards to review." - Verify that pending cards display the student name, class, and card preview. - Verify that the instructor can click a card to open the full view.
HP.3	- Verify that when a user has not sent any cards, the system displays a message such as "No Cards Sent". - Verify that the card requests are displayed in order, with the most recently sent card being first.

	- Verify that the system displays the card title, a preview of the text, and the status of each request. - Verify that when the status of any of the user's card requests changes, the new status is shown automatically without a refresh. - Verify that when a user submits a new card request, it is displayed on the user's home page with the status "Pending". - Verify that when rejected cards are clicked on the home page, an overlay opens that displays card information and the rejection reason.
HP.4	- Verify that each available page is accessible with a separate labeled button. - Verify that from the home page, the "Logout" button redirects the user to the login page, and the "Send Kudos Card" button redirects the user to the card generation page. - Verify that the mail button redirects instructors to a page containing their received and sent inboxes.
HP.7	- Verify that when a user receives a new Kudos or a sent Kudos changes status, it appears at the top of the user's inbox. It will have a distinct indication and coloring. - Verify that after a user views the new notification, its color and indicator change to indicate it has already been viewed.
HP.9	- Verify that when a user receives a card, the received count is incremented by one without requiring a manual refresh. - Verify that when a user's card approval request is approved, the sent count is increased by one without requiring a manual refresh.
CC.G2	- Verify that the CC page shall only provide input fields for: Message, Recipient, Card Design, and an Anonymous checkbox.
CC.1	- Verify that users see a counter showing the number of characters used as they type. - Verify that the system provides the users with a text box for entering the message body, separate from the card preview. - Verify that users are limited to 250 characters. - Verify that the system shall block users from typing once they reach 250 characters. - Verify that the text box will ignore the tab and enter keys when pressed. - Verify that the system requires the message to be more than 10 characters long. - Verify that the text box allows the user to paste text from the clipboard. If the pasted text exceeds the character limit, the excess text will be ignored.

CC.2	- Verify that users can choose from 3–4 card designs (more card designs may be available later), each design displayed as an image in the card design selection menu.
CC.3	- Verify that the dropdown lists all available users (first and last name) for the selected course - Verify that users can not select more than one recipient per card.
CM.1.1	- Verify that the CM interface allows instructors to view all classes they created and see a roster of all users in each class. - Verify that the roster shows each student's display name - Verify that each course entry displays the Class Name, join code, and End Date.

4.5 Logical Database Requirements

ID	**Verification Process**
HP.G1	- Verify the database is not storing PNG or JPEG Kudo Cards. The system should construct cards on demand.
HP.G2	- Verify that cards are not deleted when the course they are associated with ends.
AC.3	- Verify that if an email is already in use, account creation should be blocked, and the user is informed that an account already exists with that email.

5. Appendix A – Tailoring Policies

5.1 Acronyms and Abbreviations

Term	Abbreviation
Account Creation	AC
Card Creation Page	CC
Course Management	CM
Homepage	HP
Login Page	LP
Performance Requirements	PR

6. Appendix B – Copyright

This document is based on a template meeting the ISO/IEC/IEEE 29148-2018 standard, available at https://www.iso.org/standard/72089.html. Template authors are:

Dr. rer. nat. Bastian Tenbergen,
Associate Professor of Software Engineering
bastian@tenbergen.org

Mikayla Conner-Spagnola, MA
Independent Consultant
mconner@oswego.edu

Department of Computer Science
State University of New York at Oswego
Oswego, NY 13126, United States