Winter 2026

Software Requirements Specification

Github : https://github.com/Seulgilet/PRJ566NBB_2261_Group6

SmartMatch – Student Team Project Matching Platform

Seulgi Lee, Moe Thet Paing

Yahya Osman, Jackson Pedvis

SMART MATCH | PRJ566 NBB - Group 6

Executive Summary

Background

Description

Company Value Add

End-User Value Add

Scope

What is Included

What is Not Included

Justification

Table of Contents

Section 1

1.1 Document Authors

- Seulgi Lee
- Moe Thet Paing
- Yahya Osman
- Jackson Pedvis

1.2 Document Revision History

WEEK	DATE	Revisions
1	Jan 25, 2026	● Section 1 ● Section 2.1
2	Feb 1, 2026	● Section 2.2, 2.3, 2.4, 2.5
3	Feb 8, 2026	● Section 2.6, 2,7
4		●
5		●
6		●
7		●
8		●
9		●
11		●
12		●
13		●
14		●

1.3 Document Purpose

The purpose of this Software Requirements Specification (SRS) document is to formally define and document the functional and non-functional requirements of the **SmartMatch** – *Student Team Project Matching Platform*. This document serves as the primary reference for all stakeholders by clearly specifying what the system is expected to accomplish, the boundaries of the system, and the conditions under which it must operate.This SRS establishes a common understanding between the project team, faculty, and other stakeholders regarding system objectives, scope, and constraints before the commencement of system design and implementation activities. By capturing requirements in a structured and standardized format, the document aims to minimize ambiguity, reduce the risk of misinterpretation, and prevent scope creep throughout the project lifecycle.

1.4 Audience

This Software Requirements Specification is intended for a diverse audience involved in or affected by the SmartMatch project.

The primary audience includes:

- **Project Team Members**, who will use this document to guide analysis, design, development, and testing activities in alignment with defined requirements.

- **Faculty and Course Evaluators**, who will review the document to assess academic rigor, requirement completeness, and adherence to professional software engineering standards.

- **Project Stakeholders and Client Representatives**, who will validate that the documented requirements accurately reflect the intended problem domain and proposed solution.

A secondary audience may include future developers or maintainers who require an understanding of system requirements and rationale to support system enhancement or maintenance. The document is written to balance technical clarity with accessibility, ensuring it remains understandable to both technical and non-technical readers.

1.5 Group Agreement

Team:

Project Planning and Management Group #6

Project Title

SmartMatch - Student Team Project Matching Platform

Project Time Frame:

Approximately 6 month time frame

Winter 2026 - Summer 2026

Team Members:

Seulgi Lee

Yahya Osma

Moe Thet Pain

Jackson Pedvis

Team Leadership:

Project Manager: Seulgi Lee

Mainly responsible for project coordination, timeline management, final tests and project refinement.

Team Functions/Roles:

Seulgi Lee : Project Manager/ frontend development

Yahya Osman : Documentation/ Frontend development/ Business case Development

Moe Thet Paing: Backend Development/ UI/UX Development

Jackson Pedvis: Documentation/Quality Assurance / Team coordination

Team Meetings

Meeting #1: In person, Mondays 13:30

Meeting   #2: Online, Thursdays 11:40

Team Problems

Some potential problems the team might face:

1. Schedule conflicts
2. Workload balance
3. Technical issues
4. Communication issues

Team Commitment

The undersigned members agree to work together on the project until the end of the PRJ666 next Semester. They recognize that as a team and individually they are equally responsible for the quality of all deliverables.

Name	Date	Signature
Seulgi Lee	25/01/2026	SL
Yahya Osman	25/01/2026	YO
Moe Thet Pain	25/01/2026	MP
Jackson Pedvis	02/02/2026	JP
Rohit SivaKumar	08/02/2026	RS

Section 2

2.1 Project Proposal

2.1.1 Project Background

Market Overview

- Many programs (computer science, engineering, business) rely on **graded group work** as a core assessment method.

- Research indicates that **well-structured team** learning can improve educational outcomes compared to individual learning.

- Growth of **hybrid and online learning** environments has increased reliance on digital coordination and collaboration tools.

The global Learning Management System (**LMS**) market is valued in the **tens of billions of USD** and is projected to grow steadily through the next decade.Institutions continue investing in digital learning infrastructure that supports course delivery and student management.

Current Team Formation Methods

- Student self-selection
- Manual instructor assignment
- Random assignment via Learning Management Systems (LMS)

Traditional student grouping methods frequently struggle to balance efficiency with equity: self-selection often fosters bias, manual assignment fails to scale for large cohorts, and **random assignment** ignores essential variables like **skill sets and schedules.** This gap is further widened by current LMS platforms, which provide basic automation but lack the sophisticated, multi-criteria optimization and transparent logic necessary to create truly balanced and effective teams.

Competitor Landscape

Overview	Capabilities	Limitations
Learning Management Systems (LMS)	- Provide group creation and management tools - Support manual or random student assignment - Integrated into existing academic workflows	- Do not optimize team composition using multiple student attributes - Limited support for skill-based or role-based matching
Cross-Domain Grouping and Matching Systems	- Use constraint-based matching with availability and preferences - Demonstrate scalability of multi-criteria assignment approaches	- Not designed for academic assessment contexts - Do not address grading fairness or educational learning outcomes - Lack support for student-specific roles or project skills
Post-Formation Team Management Tools	- Support peer evaluation and team performance monitoring - Enable instructor intervention after teams are formed	- Do not address issues arising from poor initial team formation - Focus on corrective measures rather than prevention

Market Gap

- Existing solutions focus on **administrative convenience**, not **team quality optimization**.
- There is no lightweight system that:
    1. Uses structured student profile data

2. Applies multi-factor matching (skills, availability, role preference)
3. Provides explainable team formation outcomes

2.1.2 Problem Statement

The problem of	In many universities and colleges, team projects are an essential part of coursework. However, when forming teams, institutions still rely on random assignment or student self-selection methods, **resulting in groups being formed without any meaningful information about group members**. In these cases, students are assigned to teams without knowing their teammates' skills, availability, work styles, or expectations, which creates uncertainty at the start of projects and makes effective collaboration difficult from the beginning.
Affects	This problem affects **students, professors, and educational institutions** alike. Students frequently encounter **unfair workload distribution**, lack of leadership, **scheduling conflicts**, and interpersonal issues. Professors and institutions must invest additional time and effort in managing team conflicts, addressing complaints, and ensuring fair evaluation.
The impact of which is	As a result, **teams are unable to work effectively from the beginning of the project**. Collaboration becomes inefficient, time and effort are wasted, learning outcomes diminish, and the operational burden on institutions increases.
A successful solution would be	An ideal solution should provide a team formation process that considers **individual characteristics and circumstances, such as students' skills, availability, work styles, and role preferences**, along with a transparent team formation process, promotion of balanced participation, and reduction of early-stage project conflicts. Through this approach, it should improve students' learning experience and enable institutions to manage team-based courses more effectively.

2.1.3 Product Vision

For (target customer)	For educational institutions such as universities and colleges
Who	Face challenges caused by random or informal team formation methods, including team conflicts, unbalanced workload distribution, reduced learning outcomes, and increased administrative effort related to managing team-based projects
The SmartMatch	SmartMatch is a B2B web-based team matching platform designed to support the structured formation of student teams for academic, course-based team projects by using student-provided data rather than random or manual assignment methods
That	Considers individual skills, availability, role preferences, and collaboration styles to form balanced teams, provides transparency in the team formation process, reduces conflicts at the early stages of projects, and improves student participation and overall learning experience,
Unlike	Unlike manual, random, or instructor-driven team assignment methods, or basic group creation features commonly found in learning management systems
Our product	SmartMatch offers clear matching criteria and understandable team composition outcomes, enabling institutions to manage team-based courses through consistent, data-driven team formation processes, fair workload distribution based on student attributes, and reduced administrative time spent on manual team assignment and conflict management.

2.2 Stakeholders and Users

Stake Holder Overview

SmartMatch involves multiple stakeholder groups who **interact with**, are **impacted by,** or **influence the system**. These stakeholders have different goals, responsibilities, and levels of involvement. Identifying them helps ensure that system requirements align with real-world academic and operational needs.

Can be segmented into two groups:

- **Internal stakeholders** are directly involved in **building, operating, or managing** SmartMatch.

- **External stakeholders use**, are **affected by**, or **influence outcomes**, but do not build or maintain the system.

Internal StakeHolders

StakeHolder	Role	Interest
Project Development Team	Designs and builds the system	Clear requirements, feasible scope
System Administrator	Maintains system availability	Security, uptime, data integrity
Academic Institution	Host the system	Policy compliance, consistency

External Stakeholder

StakeHolder	Role	Interest
Students	Receive team assignments	Fairness, workload balance
Course Instructors	Configure and approve teams	Transparency, reduced conflict
Teaching Assistants	Support instructors and students	Visibility , issue monitoring

Stakeholder & User Avatars

This represents typical stakeholders, not specific individuals.

Avatar 1 – Student User

- **Role:** Undergraduate student in team-based course
- **Goals:** Fair teams, predictable workload
- **Concerns:** Free-riders, schedule conflicts
- **Interaction:** Profile creation, team review

Avatar 2 – Course Instructor

- **Role:** Faculty member managing group projects
- **Goals:** Balanced teams, fewer disputes
- **Concerns:** Transparency, grading fairness
- **Interaction:** Set criteria, review assignments

Avatar 3 – System Administrator

- **Role:** IT staff maintaining platform
- **Goals:** Stability, data protection
- **Concerns:** Security, access control
- **Interaction:** System monitoring

Avatar 4 – Project Development Team

- **Role:** Student developers
- **Goals:** Meet SRS requirements
- **Concerns:** Scope creep, ambiguity
- **Interaction:** Implement features

2.3 Project Scope

This section explains what the smartMatch system is responsible for, what it does and what it doesn't. Defining the project scope helps keep the project focused and avoids adding features that are outside the original goals.

What the system will do:

SmartMatch is designed to help make a fair and balanced student teams for course based group projects.

The system modules:

- User Authentication modules:
  - Manages and processes user authentication, login credentials, and role-based access control to ensure that students and instructors can securely access the appropriate features and functionalities of the system
- Student profile management:
  - Enables students to create comprehensive personal profiles that include detailed information about their technical skills, subject matter expertise, weekly availability and scheduling preferences, preferred team roles and responsibilities, and individual working style preferences
- Instructor module:
  - Enables instructors to define and configure team formation rules, including parameters such as team size, matching constraints, and grouping criteria
- **Algorithm-Based Team Matching module (Core)**:
  - Automatically generates balanced and fair project teams by analyzing student data and instructor-defined rules. This module creates optimal team combinations by considering students' skills, preferences, and availability.
- Web Interface Module:
  - Provides a responsive, browser-based interface that ensures accessibility across a wide range of devices, including desktop computers, laptops, tablets, and smartphones, allowing users to interact with the system seamlessly regardless of their device choice.

What the system will not do:

- Project Management Component
  - It does not include project execution and management features such as task list management, deadline management, and progress tracking after teams are formed.
- Communication Component
  - It does not provide built-in communication tools such as chat, messaging, video calls, or discussion boards.
- Assessment and Grading Component
  - Grading, scoring, and performance evaluation features for instructors are not supported by the system.
- Learning Management System Integration Component

- Integration features with external learning management systems such as Moodle or Blackboard are not included in the current project scope.

Scope summery:

The SmartMatch system will maintain a singular and dedicated focus exclusively on the process of creating, forming, and organizing student teams in a manner that ensures fairness, balance, and equitable distribution of skills and resources, specifically for the purpose of course-based group projects within academic settings.

2.4 System Risks

This section describes potential problems that could happen while building, launching, and using the SmartMatch system. Each problem is reviewed based on how likely it is to happen (Probability) and how serious it would be (Impact). SmartMatch is an academic project that helps form teams in settings where this was usually done by hand. Problems are grouped into five types: Market, Financial, Technology, People, and Structural/Process. For each problem, we suggest ways to reduce or prevent it.

Probability / Impact Scale

- **Low**: Not likely to happen or won't cause much harm
- **Medium**: Somewhat likely to happen or could cause some harm
- **High**: Very likely to happen or could cause serious harm

Probability / Impact Matrix

		Low	Medium	High
Probability	High		Risk 8 Risk 2 Risk 6	Risk 3
	Medium	Risk 4	Risk 9	Risk 5 Risk 7
	Low		Risk 1	
		Low	Medium	High
			Impact	

Risk No	Type	Risk Category	Risk Description	Probability	Impact	Mitigation Strategy
Risk 1	**General**	**Market Risk**	Schools and universities that have always formed teams by hand may not	Medium	Low	Explain the system's advantages clearly, such as less work for administrators and

			see why they need a new system			better team results than doing it manually.
Risk 2	**Specific**	**Market Risk**	Students may not want to enter their profile details or may enter very little information, which could result in low system usage	High	Medium	Only ask students for the most important information and give them simple instructions to make it easier for them to participate.
Risk 3	**General**	**Financial Risk**	The academic nature of the project limits the budget available for system development and operation	High	Medium	Use open-source technologies and free development tools, and focus development efforts on core MVP functionality.
Risk 4	**General**	**Financial Risk**	Unexpected operational costs may occur during deployment or testing environment setup	Low	Medium	SmartMatch is an academic project with limited scope and controlled usage. It operates only during specific courses or project periods. Costs are minimized by limiting users and operation duration.
Risk 5	**Specific**	**Technology Risk**	The matching rules may not work perfectly in every situation	Medium	High	Begin with basic matching rules that are easy to understand, then make them better step by step based on what you learn from testing.
Risk 6	**Specific**	**Technology Risk**	If students give wrong or incomplete information, the team matching may not work well	High	Medium	Use input validation, define required fields, and restrict data

						formats to ensure data consistency.
Risk 7	**Specific**	**Technol ogy Risk**	Not enough students may be available to meet the team requirements, making it hard to form teams properly	Mediu m	High	Set up basic matching rules that work when there isn't enough data. Allow the system to be more flexible or let someone make changes by hand if needed.
Risk 8	**General**	**People Risk**	Team members may miss work on the project because of their classes or personal reasons	High	Mediu m	Make realistic schedules and check progress regularly to handle schedule changes.
Risk 9	**Specific**	**Structu ral / Process Risk**	Transitioning from manual to system-based team formation may make it hard to know who is responsible when problems happen with teams	Mediu m	Mediu m	Clearly define the system's role and limitations. Document operational guidelines and responsibility boundaries.

Risk Summary

The SmartMatch project has two main challenges. First, we need to switch from forming teams by hand to using an automated system. Second, we need to make sure the matching works well when students provide good information. Since this is a academic project used in specific classes, we can solve these problems by communicating clearly, setting clear system limits, and making improvements step by step.

2.5 Operating Environment

This section describes the environment in which the system is expected to operate.

Hardware / Devices Supported

SmartMatch is a web-based system designed to run on standard user devices without requiring specialized hardware.

The system supports the following devices:

- Desktop and laptop computers
- Tablet devices
- Smartphones (via a responsive web interface)

Users access the system through a web browser, and no high-performance hardware is required.

Operating System

SmartMatch is operating system independent. Since the system runs in a web browser, it can be accessed on any operating system that supports modern web browsers, including:

- Windows
- macOS
- Linux
- iOS
- Android

No operating system–specific installation is required.

Software Dependencies

The system requires a modern web browser to function properly. Supported browsers include, but are not limited to:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Apple Safari

The server needs a basic web setup and a database to run the system.

Users don't need to install anything except a web browser.

Internet Connection

You need an internet connection to use SmartMatch.

You must be online to:

- Log in to your account
- Create or update your profile
- Request team matches
- See your team assignments and match details

You cannot use the system without internet.

2.6 Functional Requirements

Overview

This section defines the functional requirements of the **SmartMatch** system.

Functional requirements describe what the system must do, including system behaviors, features, and user interactions. These requirements are derived from stakeholder analysis and user story interviews conducted with potential end users.

2.6.1 User Story Interviews

User Persona Avatars

Student	A user seeking compatible teammates for academic projects
Instructor & Lecturers	A faculty member managing courses and approving teams
System Administrator	System Config and Management

Interview Detail

Type : Oral Interview

Participants : post-secondary students working on group-based assignments

Objective: Identify **pain points** in forming fair and effective student teams

Interview Summarization

Key Problems Identified

- Unequal workload distribution

- Skill mismatch among teammates

- Last-minute team formation

- Lack of accountability

User Stories

Student User Stories

Id	User Story
1	As a student, I want to enter my skills and strengths so that I can be matched with compatible teammates.
2	As a student, I want to specify my availability so that team members have aligned schedules.
3	As a student, I want to rate past team experiences so that future matches improve.

Instructor User Stories

Id	User Story
1	As an instructor, I want to set team formation criteria so that teams align with learning objectives.
2	As an instructor, I want to view team composition reports so that I can monitor fairness.

2.6.2 Core Functionalities

Id	User Story
CFR - 1	The system shall allow students to create and manage user profiles.
CFR - 2	The system shall collect student skill data and

	availability preferences.
CFR - 3	The system shall generate team recommendations based on compatibility rules.
CFR - 4	The system shall allow students to accept or reject team recommendations.
CFR - 5	The system shall allow instructors to configure team size and matching criteria.
CFR - 6	Data base for future analysis and report.

2.7 Nonfunctional Requirements

Overview

Nonfunctional requirements define **how the system performs**, focusing on quality attributes such as performance, security, usability, and reliability.

Usability Requirements:

- The system will provide a clear and easy to stir user interface for all users
- The system will be accessible through web browsers without requiring any software installation
- The system will have responsive design for all kind of devices (Desktops, phones, tablets)

Performance Requirements:

- The system will generate team matching results within a reasonable time according to class and group sizes
- The system will support multiple access by different users without noticeable performance issues

Security Requirements:

- The system will protect user data from unauthorized access
- User authentication will be stored in cryptic form for protection

Reliability Requirements:

- The system will handle invalid inputs
- The system will recover from system errors
- The system will allow profile updates and team generation while maintaining data consistency

Compatibility Requirements:

- The system will be compatible with different web browsers (Chrome, FireFox, Safari)
- The system will not be dependant on an operating system

Maintainability Requirements:

- The system will support future improvements
- The system code base will follow standard developments practices

2.8 UI/UX Interface Mock-ups

Section 3

3.1 Data Flow Diagrams

3.2 Iser Stories and related Use Case Scenarios

3.3 Activity Diagrams

3.4 Business Rules

Business Rule #	Description	Activity Diagram	Related UCS	UI Mock-up
BR1		AD1	UC1	UI 2.7.2
BR2		AD2	UC2	UI 2.7.3
BR3		AD3	UC3	UI 2.7.4
BR4		AD3	UC3	UI 2.7.4
BR5		AD5	UC4	UI 2.7.6
BR6		AD6	UC5	UI 2.7.6
BR7		AD7	UC6	UI 2.7.7
BR8		AD8	UC7	UI 2.7.8
BR9		AD8	UC7	UI 2.7.8
BR10		AD8	UC7	UI 2.7.8
BR11		AD8	UC7	UI 2.7.8
BR12		AD8	UC7	UI 2.7.8
BR13		AD9	UC8	UI 2.7.9
BR14		AD9	UC8	UI 2.7.9
BR15		AD9	UC8	UI 2.7.9
BR16		AD9	UC8	UI 2.7.9
BR17		AD10	AD9	UI 2.7.9
BR18		AD10	AD9	UI 2.7.9
BR19		AD10	AD9	UI 2.7.9
BR20		AD11	UC10	UI 2.7.10
BR21		AD11	UC10	UI 2.7.11
BR22		AD11	UC10	UI 2.7.11
BR23		AD12	UC11	UI 2.7.10
BR24		AD13	UC12	UI 2.7.12

Section 4 – Domain Class

Section 5 – Database

Section 6 – Project Management

6.1 Work Breakdown Structure

6.2 Milestones & Acceptance Criteria

Section 7 – Product Backlog & Implementation Schedule

Section 8 – Client/Faculty Sign-off