

Student Life HUB

Team J.A.D.E.

Ava Tonini, Dylan Dunmire, Joshua Partridge

Software Requirements Specification Document

Version: 1.0

Date: 10/12/2025

Table of Contents

1 Purpose	3
2 Scope.....	3
3 User characteristics.....	4
3.1 Key users	4
3.2 Secondary users	4
3.3 Unimportant users	4
4 Product perspective.....	4
4.1 System Context	4
4.2 User interfaces.....	5
4.3 Software interfaces.....	6
4.4 Hardware interfaces and Memory constraints.....	6
4.5 Deployment requirements	7
5 Assumptions and Dependencies.....	7
6 Specific requirements	7
6.1 System Functional Requirements	7
6.2 Logical Database Requirements	8
6.3 Software System Attributes.....	8
6.3.1 Usability	8
6.3.2 Performance	8
6.3.3 Reliability/Dependability	8
6.3.4 Security	8
6.3.5 Maintainability	9

1 Purpose

Business Background:

College students today face increasing demands from academic, personal, and social responsibilities. Surveys show that:

- 86% of U.S. students struggle with time management (Admin, 2023).
- 70% experience insufficient sleep (Hershner & Chervin, 2014).
- 60% report academic burnout, while 80% feel overwhelmed by stress (Ingham, 2024).

These challenges directly affect academic performance, health, and overall quality of life. Existing tools are fragmented as students must juggle multiple apps (calendars, fitness trackers, meal planners, shopping lists, sleep apps, etc.) to manage their daily responsibilities. This lack of integration leads to disorganization, inefficiency, and added stress.

Why a New System is Needed:

The deficiencies in current systems stem from:

- Fragmentation: No unified solution tailored to the student lifestyle.
- Inefficiency: Switching between apps consumes time and increases the likelihood of missing tasks.
- Burnout risk: Poor organization leads to unhealthy sleep, eating, and study habits.

A new system, Student Life Hub, is needed to consolidate tools into a single, student-centered platform. By addressing time management, organization, and well-being in one place, the system will reduce stress, improve productivity, and support healthier lifestyles.

2 Scope

Goals & Objectives

The Student Life Hub will:

- Provide a centralized platform for managing classes, assignments, and exams.
- Integrate wellness tools (sleep, fitness, and meal tracking).
- Support lifestyle management, such as shopping lists, tutoring sign-ups, and event reminders.
- Deliver analytics and reminders that encourage balanced academic and personal life management.

Application of the Software:

The Student Life Hub will be used daily by students to:

- Plan coursework, assignments, and deadlines.
- Track personal health goals such as sleep schedules and fitness routines.
- Manage day-to-day tasks like meal planning and shopping.
- Receive intelligent reminders and insights to encourage healthier time allocation.

Potential Benefits:

Beyond direct users (students), the software offers value to:

- Academic Advisors & Support Staff: Access to aggregated (non-identifiable) data to identify student struggles and provide support.
- Universities: Improved student retention and satisfaction by addressing burnout and wellness concerns.
- Parents & Guardians: Peace of mind knowing students have access to tools that support time management and balance.

3 User characteristics

3.1 Key users

- **User Role Responsibilities:** Undergraduate and graduate students will use the product daily to manage academic schedules, wellness, and personal responsibilities.
- **Subject Matter Experience:** Journeyman – familiar with academic workload but often lacks advanced organizational strategies.
- **Technological Experience:** Journeyman to Master – most students are comfortable with mobile and web apps.
- **Other Characteristics:**
 - Age group: 18-30.
 - Education: College-level, postgraduate.
 - Attitude toward technology: Generally positive, expect intuitive and fast interfaces.
 - Linguistic skills: Fluent in English; international students may require multilingual support.
 - Potential disabilities: Some students may have ADHD, dyslexia, or vision impairments, making accessibility features critical.

3.2 Secondary users

- **User Role Responsibilities:** Academic advisors and campus support staff may use aggregated reports to identify student trends (e.g., stress, workload imbalance).
- **Subject Matter Experience:** Master – strong knowledge of student challenges and support mechanisms.
- **Technological Experience:** Novice to Journeyman – Advisors may not be as tech-savvy as students.
- **Other Characteristics:**
 - Age group: 30-60.
 - Education: Advanced degrees (master's, doctorate, counseling certifications).
 - Attitude toward technology: Neutral to positive, value data insights but prefer simplified dashboards.
 - Physical/Intellectual disabilities: Rare but should be accommodated for accessibility.

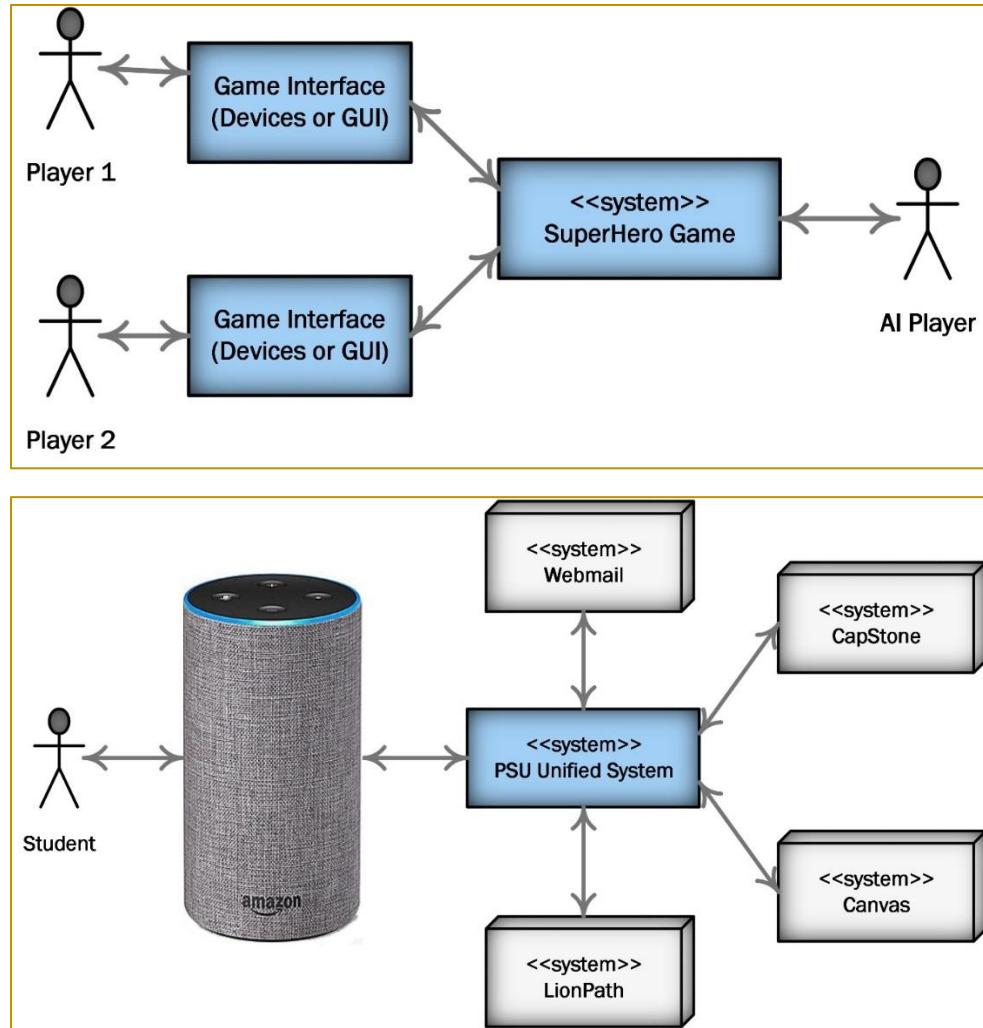
3.3 Unimportant users

- **User Role Responsibilities:** Parents, guardians, or visitors may occasionally access the app, but their input is not essential for long-term success.
- **Subject Matter Experience:** Novice – limited understanding of student life challenges.
- **Technological Experience:** Novice – may not frequently use modern productivity/wellness apps.
- **Other Characteristics:**
 - Age group: 40-65+.
 - Education: Varies.
 - Attitude toward technology: Mixed; some may resist digital tools

4 Product perspective

4.1 System Context

Define the system's relationship to users or other related systems. If the system is an element of a larger system, then identify the interfaces between the system covered by this SRS and the larger system. A block diagram showing the major elements of the larger system, interconnections, and external interfaces can be helpful. Below are two example block diagrams (with the system to be developed highlighted):



4.2 User interfaces

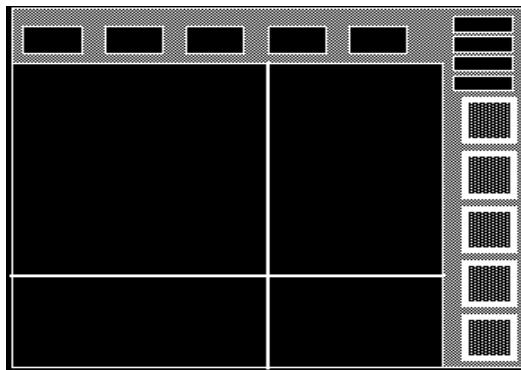
Specify the *required* characteristics of each interface between the software product and its users. Keep in mind that we are specifying requirements, so we here only care about the requirements on interfaces (this is NOT the place to put user interface designs by the software developers). For each requirement on an interface, it may include:

- Required screen formats (text only or multi-media),
- Required page or window layouts,
- Required content of any reports, menus, error messages,
- Required backup and recovery operations, or
- Required user-initiated operations or function keys.

Some examples:

- 1) It is required that the system provides the option of long and short error messages.
- 2) It is required that the system allows users to save work-in-progress and load previous works

- 3) It is required that the system interfaces have the following screen layout:



4.3 Software interfaces

Specify the use of other required software systems (e.g., a data management system, an operating system, or a mathematical package), and interfaces with other software systems.

*[Note: This is only for **customer-specified systems** that you **have** to interact with. Choosing SQL Server 7 as a DB without a customer requirement is a Design choice, not a requirement. This is a subtle but important point to writing good requirements and not over-constraining the design.]*

- For each required software product, if possible, specify: a) Software name; b) Software specification number; c) Software version number; d) Source.
- For each interface specify the required message formats. If the interface is well-documented elsewhere, provide a reference to the document defining the interface.

For instance if your customer uses SQL Server 7 DB and you are required to use that, then you need to specify

- The system must use SQL Server 7 as its database component.
- Communication with the DB is through ODBC connections.

4.4 Hardware interfaces and Memory constraints

Specify the logical characteristics of each interface between the software and the hardware elements of the system. This includes configuration characteristics (number of ports, instruction sets, etc.). It also covers such matters as what devices are to be supported, how they are to be supported, and protocols.

Specify required characteristics and constraints on primary and secondary memory.

For example,

- The terminal output should support full-screen of 1080X800 pixel.
- Because the target devices have only 2G of RAM, it is required that the design footprint should not exceed 2G.

This is mainly applicable to embedded systems. If this is not applicable to your course project, simply remove this subsection. Don't just make up something here.

4.5 Deployment requirements

Specification of the deployment environment that is required for the installation and operation of the software.

If any modifications to the customer's work area would be required by your system, then document that here. Any equipment the customer would need to buy or any software setup that needs to be done so that your system will install and operate correctly should be documented here.

This could be hardware-specific, For instance, "A 100Kw backup generator and 10000 BTU air conditioning system must be installed at the user site prior to software installation".

This could also be software-specific like, "New data tables created for this system must be installed on the company's existing DB server and populated prior to system activation."

5 Assumptions and Dependencies

- A.1 It is assumed that users possess basic computer literacy and familiarity with common mobile and web applications.
- A.2 It is assumed that users will have access to a stable internet connection to support cloud-based data storage and synchronization.
- A.3 It is assumed that the system will be deployed on modern devices (desktop and mobile) running current versions of major operating systems (Windows 10+, macOS 12+, Android 12+, iOS 15+).
- A.4 The system depends on reliable third-party APIs for certain integrations, such as calendar synchronization, fitness tracker data, and email or push-notification delivery.
- A.5 It is assumed that institutional users (e.g., universities) will provide secure access to academic data through approved APIs or export files if such integrations are implemented.
- A.6 The system depends on a cloud database service for storage, backup, and analytics functions.
- A.7 It is assumed that data privacy regulations (FERPA, HIPAA, GDPR where applicable) will remain consistent for the duration of development; major legislative changes could alter system requirements.
- A.8 It is assumed that the average user device will have sufficient memory and processing power to run a standard web application or native mobile application.
- A.9 The project depends on continued support and maintenance from the development team or hosting provider after deployment.
- A.10 It is assumed that students will provide accurate and truthful data inputs (e.g., sleep, meals, assignments) for analytics and recommendations to remain meaningful.

6 Specific requirements

6.1 System Functional Requirements

- R.1 The system shall allow students to create, edit, and manage academic schedules, including classes, assignments, and exams.
- R.2 The system shall send automated reminders for upcoming deadlines, exams, and events.
- R.3 The system shall allow users to log and track sleep duration and quality.
- R.4 The system shall include a fitness tracking feature for recording workouts and progress.
- R.5 The system shall provide a meal tracking module for logging daily food intake.
- R.6 The system shall allow students to create and manage shopping lists.
- R.7 The system shall include a dynamic calendar view integrating academic, personal, and wellness events.
- R.8 The system shall generate analytical insights to help students balance academic and personal activities.
- R.9 The system shall allow academic advisors to view aggregated, anonymized wellness and performance data.

- R.10 The system shall include a notification system for personalized alerts and recommendations.
- R.11 The system shall support user authentication through secure login credentials.
- R.12 The system shall allow users to edit, delete, and update all stored personal data.
- R.13 The system shall support accessibility features including screen reader compatibility and adjustable text size.
- R.14 The system may allow users to sync data from third-party apps such as calendars or fitness trackers.
- R.15 The system may include a tutoring sign-up interface connected to university resources.

6.2 Logical Database Requirements

- R.16 The database shall store user profiles containing unique user IDs, names, and authentication credentials.
- R.17 The database shall maintain academic records including class schedules, assignments, and exams linked to each user.
- R.18 The database shall store fitness data including activity type, duration, and date.
- R.19 The database shall store meal data including date, meal type, and nutritional details.
- R.20 The database shall store sleep records including start time, end time, and duration.
- R.21 The database shall store event details including title, date, time, and category.
- R.22 The database shall maintain relationships between user accounts and their academic, wellness, and personal data entities.
- R.23 The database shall ensure referential integrity between all related tables through primary and foreign keys.
- R.24 The database shall log user interactions for analytics and recommendations.
- R.25 The database shall encrypt all sensitive and personally identifiable information, including passwords and health-related data.

6.3 Software System Attributes

6.3.1 Usability

- R.26 The system shall feature an intuitive interface enabling users to complete common tasks within three clicks.
- R.27 The interface shall maintain consistent navigation, layout, and terminology across all modules.
- R.28 The system shall achieve at least 90% user satisfaction in usability testing.
- R.29 The system shall provide onboarding tutorials for new users.
- R.30 The system shall be fully accessible according to WCAG 2.1 Level AA standards.

6.3.2 Performance

- R.31 The system shall support at least 10,000 simultaneous users without performance degradation.
- R.32 The system shall load all primary dashboard pages within two seconds under normal load.
- R.33 The system shall process 95% of transactions (such as data saves or updates) in less than one second.
- R.34 The system shall handle up to 1 GB of stored data per user account.
- R.35 The system shall update analytical insights within 30 seconds of new data input.

6.3.3 Reliability/Dependability

- R.36 The system shall maintain an uptime of at least 99.5% over any 30-day period.
- R.37 The system shall automatically back up all critical data daily.
- R.38 The system shall recover from unexpected shutdowns without data loss.
- R.39 The system shall notify administrators of any data corruption or failed backups.
- R.40 The system shall undergo reliability and regression testing before each major release.

6.3.4 Security

- R.41 The system shall enforce secure user authentication before granting access to personal or academic data.

- R.42 The system shall use industry-standard encryption (e.g., AES-256, TLS 1.3) for all stored and transmitted data.
- R.43 The system shall automatically log out users after 15 minutes of inactivity.
- R.44 The system shall restrict access to administrative functions using role-based permissions.
- R.45 The system shall maintain an audit log of all login attempts and security-related events.
- R.46 The system shall comply with FERPA, HIPAA, and other applicable data privacy regulations.

6.3.5 Maintainability

- R.47 The system shall use a modular architecture to enable independent feature updates and scalability.
- R.48 The system shall provide internal documentation for all major components and interfaces.
- R.49 The system shall allow bug fixes and feature updates without requiring full system downtime.
- R.50 The system shall support automated testing to verify functionality after updates.
- R.51 The system may allow future plugin or module integrations without requiring substantial redesign.

7. References

Admin. (2023, September 8). The importance of time management: College edition. College LSP. <https://www.collegelsp.com/2023/09/08/the-importance-of-timemanagement-college-edition/>

Hershner, S. D., & Chervin, R. D. (2014, June 23). Causes and consequences of sleepiness among college students. National Institutes of Health. <https://pmc.ncbi.nlm.nih.gov/articles/PMC4075951/>

Ingham, P. (2024, October). Addressing burnout and stress in college healthcare students and educators. ASCLS. <https://ascls.org/breaking-the-silence/>