Software Requirements Specification

for

SkillsBridge SG

Version 1.0 approved

Prepared by <author>

<organization>

<date created>

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1. Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>

1. Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

1. Intended Audience and Reading Suggestions

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

1. Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

1. References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

2. Overall Description

2. Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

2. Product Functions

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

2. User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

2. Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

2. Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

2. User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2. Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external

factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

3. External Interface Requirements

3. User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

3. Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

3. Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

3. Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>

4. System Features

4.1 Visualizing future paths

4.1.1 Description and Priority

This feature shows a clear, data-backed connection between a student's current choices (e.g., subject combinations, CCAs, polytechnic courses) and their potential future academic and career opportunities. The system leverages government datasets (e.g., Graduate Employment Survey, university/polytechnic enrolment data) to generate predictive pathways.

- **Benefit:** Helps students/parents align passions with realistic future opportunities (UC02)
- **Penalty (if absent):** Students risk making uninformed, stressful choices (UC02)
- **Cost:** Moderate — Requires integration with multiple datasets and recommendation logic (UC02)
- **Risk:** Data freshness/accuracy dependency on government sources (UC02)
- **Priority:** High

4.1.2 Stimulus/Response Sequences

Scenario A (Student User):

1. Student inputs current interests, subjects, or CCA.
2. System matches inputs with datasets available.
3. System displays multiple pathways (secondary → poly/JC → university → career).

System Responses:

- Show pathways as an interactive flowchart/timeline.
- Display pathways in comparison view.

4.1.3 Functional Requirements

- **REQ-1.1:** The system shall allow users to input student profile.

- **REQ-1.2:** The system shall map these inputs to relevant post-secondary courses using data.gov.sg datasets.
- **REQ-1.3:** The system shall map these inputs to relevant jobs using data.gov.sg datasets.
- **REQ-1.4:** The system shall build potential career pathways using the mapped courses and jobs.
- **REQ-1.5:** The system shall display these potential career pathways in a visual, interactive manner (timeline or flowchart).
- **REQ-1.6:** The system shall save all potential career pathways upon command from the user.
- **REQ-1.7:** The system shall generate a sharable link for the page containing the career pathways upon request from the user.
- **REQ-1.8:** The system shall display a fallback message when no matching pathway is found and suggest related areas.

4.2 Exploring options holistically

4.2.1 Description and Priority

This feature enables students and parents to explore multiple academic and career pathways beyond simple school/course listings. It integrates skills, experiences, and industry relevance into each option, helping users see the **bigger picture** rather than making decisions in isolation.

- **Benefit:** Encourages informed decision-making by showing a complete landscape (UC03/ UC05)
- **Penalty (if absent):** Students may make choices with a narrow or short-term view (UC03/ UC05)
- **Cost:** Moderate — requires data integration and additional visualization modules (UC03/ UC05)
- **Risk:** Users may feel overwhelmed with too much information (UC03/ UC05)
- **Priority:** High

4.2.2 Stimulus/Response Sequences

Scenario A (Student User):

1. Students select a subject combination (e.g., Pure Sciences).
2. System retrieves related post-secondary options (Polytechnic diplomas, JC subject streams).
3. System highlights not only courses, but also **soft skills gained** along the way (e.g. teamwork skills etc.).
4. Student clicks into one pathway → sees a combination of **academic modules + soft skills + industry links**.

Scenario B (Parent User):

1. Parent explores a career domain (e.g., Healthcare).
2. The system shows the full range of educational paths leading to that career (e.g., nursing diplomas, biomedical science degrees, alternative allied health roles).
3. Parent compares holistic paths to discuss with child.

System Responses:

- Present options as **cards** showing courses, soft skills, real-world applications.
- Provide a **"zoom-out" view** of multiple pathways across industries.

4.2.3 Functional Requirements

- **REQ-2.1:** The system shall recommend options based on student profile.
- **REQ-2.2:** The system shall retrieve the skills and experiences available at the recommended educational options.
- **REQ-2.3:** The system shall present the skills and experiences available at the recommended educational options.
- **REQ-2.4:** The system shall further retrieve any other possible career fields that could be branched out based on the student profile.
- **REQ-2.5:** The system shall allow users to view a "zoomed-out" multi-pathway map across the interconnected industries.
- **REQ-2.6:** The system shall display up to 2 recommended courses in comparison view for the user.

- **REQ-2.7:** The system shall display a fallback message when no matching pathway is found and suggest related areas.

4.3 Providing Context (Industry & Workforce Landscape)

4.3.1 Description and Priority

This feature gives students and parents a **macro-level understanding** of industries, workforce demand, and how education ties into employment opportunities. Instead of just showing “where you can study,” it shows “why these matter in the real world.” It bridges the classroom and workplace using government datasets (e.g., Graduate Employment Survey, MOM labor reports, SkillsFuture).

- **Benefit:** Builds awareness of future-proof skills and industries (UC04)
- **Penalty (if absent):** Students risk pursuing irrelevant or declining fields (UC04)
- **Cost:** High — requires integration with multiple job/industry datasets (UC04)
- **Risk:** Job market forecasts may shift rapidly, affecting accuracy (UC04)
- **Priority:** High

4.3.2 Stimulus/Response Sequences

Scenario A (Student User):

1. Students select an interest area (e.g., Artificial Intelligence).
2. System retrieves industry data showing AI-related roles, growth projections, and median salaries.
3. System displays which university/polytechnic courses lead to those roles.
4. Students can “bookmark” promising industries to track future trends.

System Responses:

- Show dashboards with **employment rates, job demand growth, skills in demand**.
- Highlight **risks** (e.g., declining demand in certain industries).

4.1.3 Functional Requirements

- **REQ-3.1:** The system shall retrieve employment and industry data from trusted sources (e.g., data.gov.sg, MOM, SkillsFuture).
- **REQ-3.2:** The system shall display workforce insights including employment rates, job demand growth, and salary benchmarks.
- **REQ-3.3:** The system shall provide visual dashboards for industry trends (graphs, charts, infographics).
- **REQ-3.4:** The system shall allow users to bookmark or save industries/careers for later review.
- **REQ-3.5:** The system shall generate warnings when users select industries marked as declining or highly competitive.

5. Other Nonfunctional Requirements

5. Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

- The system must render the two-course comparison view (UC05 Compare Two Courses) within 2 seconds for at least 95% of user requests.
- The onboarding flow (UC01 Onboard & Enter Profile → UC02 Generate Pathway Roadmap) must complete within 2 minutes for at least 80% of first-time users.
- Each Roadmap view (UC02 Generate Pathway Roadmap) must generate within 2 seconds for at least 95% of user requests.
- The system must generate a shareable link for Saved Plans (UC07 Save/Share Plan) within 2 seconds of user request.

5. Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product's design or use. Define any safety certifications that must be satisfied.>

- The system must safeguard against misleading users by displaying the data source and dataset reference year (vintage) for every employment or salary statistic shown across Visualizing Future Paths (UC02), Exploring Options Holistically (UC03), and Providing Context (UC04).
- If dataset values are older than 3 years, the User Interface must display a warning badge.
- The system must generate a warning when users select an industry flagged as declining or highly competitive (UC04 Providing Context).
- Unsaved selections (skills and chosen courses from UC01 Onboard & Enter Profile and UC03 Explore Options Holistically) must persist through a single page refresh using local storage, preventing accidental data loss during normal use.

5. Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

- All data transmission must occur over HyperText Transfer Protocol Secure (HTTPS) using Transport Layer Security version 1.2 or higher (TLS 1.2+).
- User accounts must be protected with secure password hashing using bcrypt (Blowfish Crypt) or argon2 (Password Hashing Competition Algorithm).
- The system must enforce role-based access control (RBAC), where Admin users can upload datasets and manage glossary entries (UC08 Update Data), while End-users (Students, Parents, Counsellors) can only use Visualizing Future Paths (UC02), Exploring Options Holistically (UC03), Providing Context (UC04), Compare (UC05), and Save/Share Plan (UC07).
- Only minimal personally identifiable information (PII) will be stored (email, age band), and no sensitive data will be collected.
- Telemetry (anonymous usage analytics, e.g., number of comparisons, roadmap saves) must be opt-in only, with no data collected until the user explicitly consent

5. Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

- Usability: At least 80% of first-time users must reach their first Roadmap (UC02 Generate Pathway Roadmap) in ≤ 2 minutes.
- Reliability: Following a backend restart, the system must fully restore the features described in Section 4 — Visualizing Future Paths (UC02), Exploring Options Holistically (UC03), Providing Context (UC04), Roadmap (UC02), and Compare (UC05) — within 5 minutes.
- Maintainability: The datasets (courses, employment, skills) used in Visualizing Future Paths (UC02) and Providing Context (UC04) must be

configurable via a single schema configuration file without requiring application code changes.

- Portability: The database layer supporting UC01–UC08 must be replaceable with any relational database supporting standard SQL, by updating configuration and migration scripts only.
- Accessibility: The interfaces for Visualizing Future Paths (UC02), Exploring Options Holistically (UC03), Providing Context (UC04), and Compare (UC05) must comply with Web Content Accessibility Guidelines (WCAG) 2.1 AA, including keyboard navigation, sufficient contrast ratios, and alt-text for non-text content.

5. Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

- The system must always display at least one cited data source for every recommendation or comparison in UC02 Generate Pathway Roadmap, UC03 Explore Options Holistically, UC04 Providing Context, and UC05 Compare Two Courses.
- The Compare feature (UC05) must be limited to exactly two courses at a time to preserve clarity.
- Roadmap recommendations (UC02) must always include at least one quantitative indicator (e.g., salary, employment rate) and one qualitative indicator (e.g., skills focus).

6. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

- Internationalization: English is the primary language; currency must be displayed in Singapore Dollars (SGD); metric units must be used.
- Legal/Compliance: The system must attribute all datasets used in UC02–UC05 to their original sources and comply with relevant open data licenses.
- Database: The system must support schema evolution (migration scripts) to adapt to new dataset formats without loss of existing user plans.
- Dataset Refresh: The system must schedule daily dataset refreshes from external sources (UC08 Update Data).

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Term	Definition
Student	Primary end-user who inputs interests/subjects/CCAs/skills to explore options and view pathways/roadmaps.
Parent	Supportive end-user who explores options, industry outlook, and compares courses to guide a student’s choices.
Counsellor	(If applicable) Power user who helps students interpret pathways and industry context using the system’s views.
Student Profile	The information the student provides during onboarding (e.g., interests, subjects, CCAs, strengths, selected skills) used to personalize recommendations and pathways.
Course	A programme of study offered by a post-secondary institution (e.g., Polytechnic diploma, JC stream, University degree) that users can explore and compare.

Diploma / Degree	Post-secondary (diploma) and university (degree) qualifications that appear in pathway options and comparisons.
Pathway	A data-backed sequence of educational milestones (e.g., Secondary → Poly/JC → University → Career Role) generated from datasets and user inputs.
Roadmap	The visual presentation of a chosen pathway (timeline/flow), showing how current selections link to future studies and potential jobs.
Visualizing Future Paths	Feature that displays the connection between current choices and potential academic/career opportunities using public datasets.
Exploring Options Holistically	Feature that shows courses together with skills/experiences and broader opportunities (not only listings).
Providing Context (Industry & Workforce Landscape)	Feature that surfaces industry demand, employment rates, salary benchmarks, growth outlook, and links them to related courses.
Two-Course Comparison (Compare)	View that presents **exactly two** selected courses side-by-side (admissions/skills/outcomes) to support decision-making.
Skills (Checklist)	A set of competencies the user selects; used to tailor recommendations and pathway generation.
Industry Context	Labour-market indicators (employment rate, median salary, demand growth, skills in demand) used to justify recommendations and flag risks.
Evidence Item / Evidence Panel	Data-backed justification attached to a pathway or comparison (e.g., employment rate, salary benchmark, skills gained), with charts/tables and citations.

Dataset Reference Year (Vintage)	The year the statistic represents (e.g., Graduate Employment Survey 2022); shown with every statistic, with warnings if older than three years.
Saved Plan	A persisted roadmap chosen by the student that can be revisited or shared with stakeholders.
Shareable Link	A read-only URL generated for a Saved Plan to share with others.
Trusted Sources	Official public datasets used by the system (e.g., data.gov.sg, MOM, SkillsFuture) that provide education and labour-market data.
Quantitative Indicator	A numeric metric presented as evidence (e.g., employment rate, median salary).
Qualitative Indicator	A descriptive insight (e.g., dominant skills focus, typical experiences) that complements numeric evidence.
Bookmark	An action that lets users save industries/careers of interest for later review in the context feature.
Warning Badge	UI indicator that alerts users when underlying data is stale (older than three years) or when an industry is flagged as declining/highly competitive.
Dataset Refresh (Daily)	The scheduled retrieval and update of datasets from external sources to keep recommendations current (see UC08).
data.gov.sg API	External service supplying public data; appears as an actor in **UC08 Update Data**.

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>

Source: http://www.frontiernet.net/~kwiegers/process_assets/srs_template.doc