Software Requirements Specification for Software Engineering: subtitle describing software

Team 4, EvENGage
Virochaan Ravichandran Gowri
Omar Al-Asfar
Rayyan Suhail
Ibrahim Quraishi
Mohammad Mahdi Mahboob

October 2, 2025

Contents

Revision History

Date	**Version**	**Notes**
Date 1	1.0	Notes
Date 2	1.1	Notes

1 Purpose of the Project

1.1 User Business

Insert your content here.

1.2 Goals of the Project

Insert your content here.

2 Stakeholders

2.1 Client

The client for this project is the supervisor, Luke Schuurman. He is a member of the MES and has first hand experience plannig and hosting events with the MES. As the supervisor he will play a crucial role by ensuring the project aligns with objective of the MES and integrate the platform seamlessly with existing systems. He will also provide us with feedback and guidance throughout the project and will help define the project requirements in this document.

2.2 Customer

Insert your content here.

2.3 Other Stakeholders

Insert your content here.

2.4 Hands-On Users of the Project

MES Executives and Council Members: They will be utilizing this system to create and manage events, configure forms and surveys, monitor event data and generate data analytics reports. They can be characterized as primarily undergraduate students who value their time greatly. They aim to reduce the time taken to do administrative tasks as well as provide a better experience during MES events. Their experience with systems like this can range from Journeyman - Master as they can be experienced in event planning and student engagement. There may be a slight learning curve to utilizing the new technology but these users have experience performing these functions.

McMaster Engineering Students: They will be utilizing this app to to register for events, purchase tickets, sign waivers, check in at venues, and complete feedback surveys. They want to enjoy their university experience and connect with other students, They are also very busy and value their time greatly so are looking for an intuitive and straightforward user experience. Their general experience with systems like this is Journeyman as they may have used similar systems for other use cases.

Other Students and Guest Event Attendees: This group includes non-engineering students, alumni, and invited guests who participate in large-scale MES events such as the Fireball Formal and Graduation Formal, which extend beyond the core engineering community. It could also include students and guests from other universities as well as industry which attend the engineering conferences that the MES helps host. They are generally looking for an easy and seamless experience registering and attending these events. Their general experience with systems like this is Journeyman as they may have used similar systems for other use cases.

2.5 Personas

1. **Matthew Cruise (Engineering Student)**: Matthew is a 20-year-old second-year Mechanical Engineering student at McMaster University. He lives in a shared house near campus with two close friends. He enjoys attending MES events like pub nights and the Fireball Formal as a way to balance his heavy academic workload with some enjoyment and entertainment. He usually hears about events through word-of-mouth or social media and would prefer an easy and effortless way to find and register for events. Matthew is generally tech savvy and is comfortable using online platforms but doesn't want to be bothered by too many notifications and forms. He has many ideas on how he can generally improve his university experience but doesn't believe he has an outlet to convey them. For Matthew his priority is convenience as he wants to enjoy himself but doesn't want to spend too much time or

effort doing so.

2. **Adam Clooney (MES Executive)**: Adam Clooney is a 22-year-old final-year Civil Engineering student who currently serves as VP Social on the MES. He is responsible for coordinating large-scale events like Fireball Formal, working closely with other council members to handle logistics, advertising, and student engagement. Adam is outgoing and enjoys bringing people together, but often feels the strain of balancing his role with academic responsibilities. He is proficient with common digital tools such as Google Drive, spreadsheets, and design platforms for promotions, but he's not highly technical. Adam appreciates structure and tools that keep things organized because he dislikes wasting time fixing errors or repeating work. He is motivated by the sense of accomplishment that comes from hosting a successful event and wants tools that help him stay on top of details.

3. **Margot Watson (McMaster Student)**: Margot Watson is a 21-year-old undergraduate student in Political Science at McMaster University. Although she is not part of the engineering faculty, she often attends large MES-hosted events such as the Fireball Formal and other socials because many of her friends are in engineering. Margot lives in an off-campus apartment with two roommates and enjoys being involved in student life across faculties. She has a relaxed attitude toward technology since she uses her phone daily for social media and messaging but she prefers things to be straightforward and intuitive. She is cautious with her money, balancing tuition and living expenses, but she's willing to spend on experiences with friends. She is motivated by spending time with her friends and having good experiences to have a fullfiling student life.

2.6 Priorities Assigned to Users

Key Users: Luke Schuurman, MES Council Members and Executives, McMaster Engineering Students.
Secondary Users: Other Students and Guest Event Attendees.

2.7 User Participation

Our requirements will primarily be derived through meetings with our supervisor Luke Schuurman. If we have the need to clarify or elicit more requirements we will look to engage other MES members either through our supervisor or directly.

2.8 Maintenance Users and Service Technicians

Insert your content here.

3 Mandated Constraints

3.1 Solution Constraints

Insert your content here.

3.2 Implementation Environment of the Current System

Insert your content here.

3.3 Partner or Collaborative Applications

Insert your content here.

3.4 Off-the-Shelf Software

Insert your content here.

3.5 Anticipated Workplace Environment

Insert your content here.

3.6 Schedule Constraints

Insert your content here.

3.7 Budget Constraints

Insert your content here.

3.8 Enterprise Constraints

Insert your content here.

4 Naming Conventions and Terminology

4.1 Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project

- **MES**: McMaster Engineering Society

5 Relevant Facts And Assumptions

5.1 Relevant Facts

Insert your content here.

5.2 Business Rules

Insert your content here.

5.3 Assumptions

Insert your content here.

6 The Scope of the Work

The purpose of the section is to define the expected scope of the project, including specifications on how the system is partitioned, business data models, and business use cases.

6.1 The Current Situation

Currently, the MES uses plethora of different platforms to host events and conduct surveys such as tools for registration, signing of waivers, and checking in attendees. The main platforms used are a combination of Google Forms and Google Sheets. There are several pain points to address with the current system.

- **Decentralized System Components** Registration and surverying is split across a plethora of tools and software which may not be compatible with each other. For example, Google Forms is used for registration forms and surveys, Google Sheets stores form data and LinkTree holds links to all the registration forms, and event updates is done by email or social media.

- **Overly Complex Form Logic** The current CFES survey consists of 70 pages of questions linked together through complex branching logic. The Google Forms form building UI makes this very complicated as all form elements are displayed as a linear list of sections making it very hard to track paths through the form.

- **Disorganized Data Visualization** Form response data is currently stored using Google Sheets. While Google Form data is easily imported into Google Sheets, any analytics on the data must be done manually through equations and macros.

- **Lack of Reusability** Google Forms comes with a few templates that provide an initial starting point for many types of forms such as registrant information. However, after the first section of the form, each subsequent section must be made manually. Sections may be imported from other forms but this requires the user to have access to a form with the wanted section and to scour through an unorganized list of forms and sections.

- **Low Response Rates on Surveys** The response rates on the annual CFES survey of undergraduate engineering students have been decreasing due to the long length of the Google Form and the lack of ability to submit a partially completed form.

- **Manual Registration Scheduling** Event registration is managed through a combination of Instragram, Google Forms, and LinkTree. Events are advertised on Instagram, and a link to the signup Google Form is posted on the MES LinkTree. This solution lacks automation since a new Google Form must be made for every event and links have to be manually added and removed for each form when registration is opened or closed.

6.2 The Context of the Work

This section provides an overview of the high-level inputs and outputs between the system and external systems or actors. Figure X illustrates a work context diagram showing the interactions between the system components and external elements.

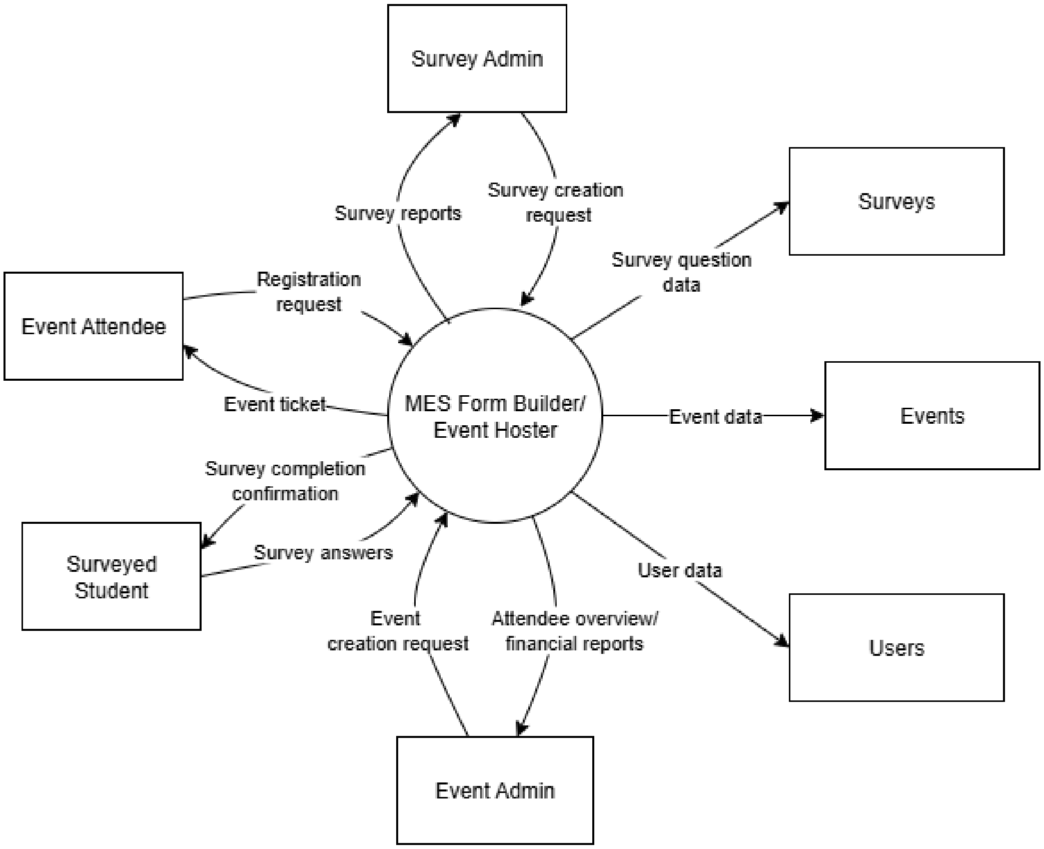

Figure 1: Work context diagram of the form builder and event hosting system

6.3 Work Partitioning

This section describes how the work done by the system can be partitioned into smaller and more manageable workflows. Below is a table describing each of the sub-workflows of the proposed systems. Note that event creation and survey creation have been grouped as a single workflow since they are nearly identical.

No.	Event	Input	Output	Requirements
1	User registers for event	Registration data	Event ticket	
2	User fills out a survey	Survey responses	Survey completion confirmation	
3	Admin creates a form module	Module question data	Reusable form module	
4	Admin creates an event/survey	Event/survey data	Event/survey creation confirmation	
5	Admin views survey/event statistics	Event/survey to view	Event/survey registration/response reports	

Table 1: Partitioning of system workflows

6.4 Specifying a Business Use Case (BUC)

BUC 1: User registers for an event
Input: Registration data
Output: Event tickets
Pre-condition: User has downloaded the application and has created an account
Scenario:

1. The user application receives the registration data
2. The user application verifies the data is filled correctly
3. The user application sends the registration data to the backend server
4. The backend server verifies the event is not full and the deadline has not passed
5. The backend server adds the user to the list of attendees

6. The backend server generates an event ticket and sends it to the user application
7. The user application confirms with the user that the registration was successful and presents the user with the ticket

Sub variations:
2a. The submitted registration data has errors, the user application prompts the user to fix the errors and resubmit
4a. The event is full or the deadline has passed, the backend sends an error code to the user application
4b. The user application alerts the user of the error.

BUC 2: User fills out a survey module
Input: Survey responses
Output: Survey completion confirmation
Pre-condition: User has downloaded the application and has created an account
Scenario:

1. The user application receives the survey data
2. The user application verifies the data is filled correctly
3. The user application sends the survey data to the backend server
4. The backend server updates the survey database with the user's data
5. The backend server sends a confirmation message to the user application
6. The user application confirms with the user that the survey data has been submitted

Sub variations:
2a. The submitted data has errors (i.e. mandatory fields not filled), the user application prompts the user to fix the errors and resubmit

BUC 3: Admin creates a form module
Input: Form fields

Output: Reusable form module
Pre-condition: Admin has access to create custom form modules
Scenario:

1. The admin portal receives the list of form fields and questions from the admin user for the custom module
2. The admin portal verifies the custom module has been created correctly
3. The admin portal sends the custom module data to the backend server
4. The backend server authenticates the admin user
5. The backend server saves the custom module data to the template database
6. The backend server sends a confirmation message to the admin portal
7. The admin portal updates the list of custom modules with the completed module
8. The admin portal confirms with the admin user that the custom module has been created

Sub variations:
2a. The submitted form module has errors (i.e. unfinished fields), the admin user is prompted to fix these errors before resubmitting
4a. Authentication of the admin fails, the admin portal is notified of the request denial

BUC 4: Admin creates an event/survey
Input: Event details
Output: Event creation confirmation
Pre-condition: Admin has access to create events
Scenario:

1. The admin portal receives the event/survey details
2. The admin portal verifies the event/survey details are correct

3. The admin portal sends the event/survey data to the backend server
4. The backend server authenticates the admin user
5. The backend server saves the event/survey data to the database of events/surveys
6. The backend server sends a message to the user application to notify users of the new event/survey
7. The backend server sends a confirmation message to the admin portal
8. The admin portal adds the created event/survey to the event/survey dashboard
9. The admin portal confirms with the admin user that the event/survey has been created

Sub variations:
2a. The submitted details have errors (i.e. event date has already passed), the admin user is prompted to fix these errors before resubmitting
4a. Authentication of the admin fails, the admin portal is notified of the request denial

BUC 5: Admin views event/survey statistics
Input: Event/survey to view
Output: Event registration/survey response report
Pre-condition: Event/survey has been created, and users have registered/responded
Scenario:

1. The admin portal receives the request to view event/survey statistics
2. The admin portal sends a request to the backend server containing the identification for the event/survey to view
3. The backend server receives the request and authenticates the admin user
4. The backend server retrieves the registrant/response data for the requested event/survey from the database

5. The backend server generates a statistical report of all the data
6. The data is sent back to the admin portal
7. The admin portal formats all the data into a readable format
8. The admin user is presented with the event/survey statistics

Sub variations:
3a. Authentication of the admin fails, the admin portal is notified of the request denial

7 Business Data Model and Data Dictionary

7.1 Business Data Model

Insert your content here.

7.2 Data Dictionary

Insert your content here.

8 The Scope of the Product

8.1 Product Boundary

Insert your content here.

8.2 Product Use Case Table

Insert your content here.

8.3 Individual Product Use Cases (PUC's)

Insert your content here.

9 Functional Requirements

9.1 Functional Requirements

FR-1: The system shall allow admins to create forms with multiple fields types.

FR-2: The system shall allow admins to organize forms for analytics, and specify how the analysis should be done.

FR-3: The system shall generate QR codes for confirmation and check-in integration.

FR-4: The system shall provide an attendee dashboard for administrators with filters for status updates on payments and waivers, event sign-ins, etc.

FR-5: The system shall store all event data in a secure, centralized database.

FR-6: The system shall store all questionnaire data in a secure, centralized database.

FR-7: The system shall allow administrators to generate analytics dashboards and export CSV/Excel reports.

FR-8: The system shall provide event attendees with a medium to relay feedback for the event.

FR-9: The system shall provide administrators with a medium to view all event feedback once an event has completed.

FR-10: The system shall store all questionnaire data in a secure, centralized database.

10 Look and Feel Requirements

10.1 Appearance Requirements

Insert your content here.

10.2 Style Requirements

Insert your content here.

11 Usability and Humanity Requirements

11.1 Ease of Use Requirements

UE-1: The system shall allow first-time users to understand how to navigate the event registration form within 3 minutes without prior training.

UE-2: The system shall provide survey progress indicators to reduce form abandonment rates.

11.2 Personalization and Internationalization Requirements

Insert your content here.

11.3 Learning Requirements

Insert your content here.

11.4 Understandability and Politeness Requirements

Insert your content here.

11.5 Accessibility Requirements

Insert your content here.

12 Performance Requirements

12.1 Speed and Latency Requirements

SL-1: The system shall have a maximum 2-second latency for form loading, optimized for mobile devices.

12.2 Safety-Critical Requirements

Insert your content here.

12.3 Precision or Accuracy Requirements

Insert your content here.

12.4 Robustness or Fault-Tolerance Requirements

RF-1: The system shall support 2047 concurrent users during peak event times.

RF-2: The system shall auto-save partially completed surveys every 30 seconds.

RF-3: The system shall recover from a server crash within 60 seconds without data loss.

RF-4: The system shall allow new form modules to be added without any downtime.

RF-5: The system shall allow administrators to update event templates without disruption during ongoing events.

12.5 Capacity Requirements

Insert your content here.

12.6 Scalability or Extensibility Requirements

SE-1: The system shall support up to 2047 registrations per event.

12.7 Longevity Requirements

Insert your content here.

13 Operational and Environmental Requirements

13.1 Expected Physical Environment

Insert your content here.

13.2 Wider Environment Requirements

Insert your content here.

13.3 Requirements for Interfacing with Adjacent Systems

Insert your content here.

13.4 Productization Requirements

Insert your content here.

13.5 Release Requirements

Insert your content here.

14 Maintainability and Support Requirements

14.1 Maintenance Requirements

MT-1: The system shall remain operational during the entirety of an event.

MT-2: The system shall have scheduled maintenance performed before and after events.

14.2 Supportability Requirements

SU-1: The system shall run as a native mobile application on Android and iOS devices.

SU-2: The system shall run as a web app accessible through Chromium, Firefox, and Safari browsers.

Insert your content here.

14.3 Adaptability Requirements

Insert your content here.

15 Security Requirements

15.1 Access Requirements

Insert your content here.

15.2 Integrity Requirements

Insert your content here.

15.3 Privacy Requirements

Insert your content here.

15.4 Audit Requirements

Insert your content here.

15.5 Immunity Requirements

Insert your content here.

16 Cultural Requirements

16.1 Cultural Requirements

Insert your content here.

17 Compliance Requirements

17.1 Legal Requirements

LG-1: The system shall store and handle all sensitive personal data in compliance with PIPEDA regulations.

17.2 Standards Compliance Requirements

Insert your content here.

18 Open Issues

Insert your content here.

19 Off-the-Shelf Solutions

19.1 Ready-Made Products

There are some products that exists on the market which solve certain components of this product but no product exists that combines these components into one. For event management there are various tools such as EventBrite and Stubhub which can allow for the selling of tickets for events. The problem with these products are that they don't integrate well with current MES Systems and don't provide all the features as required such as collecting waivers and ensuring McMaster students attend university specific events. For general data collection Google Forms is the most used tool and it allows for the creation of complex forms and stores data in spreadsheets. For further data analysis it requires effort on the part of the form creator to actually go and analyze the data to gain actual insights. Also the forms can become hard to handle as they become complex and can become confusing for respondents. As mentioned before there is no tool that has a combination of the two components.

19.2 Reusable Components

There are various components that we can use to help develop our application:

- **Next.JS**: Provide us with a framework for developing web apps in **React** and provides built in routing behaviour. It also handles many built in libraries and packages that we can use for front-end styling such as **tailwind-css** and **shadcn/ui**. We can also integrate it with other libraries for authentication and for database management

- **SurveyJS**: This is a potential tool we can use to develop the form builder as it provides us with some pre-built components we can use to develop and customize our own forms.

- **Recharts**: This could be potential charting tool we can use to develop our admin dashboard. Though there are other charting tools the benefit of Recharts is that it provides customization on top of the pre built charts.

- **Drizzle**: We can use Drizzle as an ORM to map from our database to our web-app. Drizzle is useful since it is lightweight and is code first. Drizzle's simplicity allows for faster development and is perfect for our use case.

These components will allow us to develop our application faster and more efficiently as many of these products are already optimized. All the above products are free to use so do not incur any extra costs or require the handling of licenses.

19.3 Products That Can Be Copied

There don't seem to be any products that can be copied which contain all the required components within a single product. Moreover the MES wants a product that doesn't require an external license and wants to be fully built in house so they have control over future development and usage.

20 New Problems

20.1 Effects on the Current Environment

Insert your content here.

20.2 Effects on the Installed Systems

Insert your content here.

20.3 Potential User Problems

Insert your content here.

20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

Insert your content here.

20.5 Follow-Up Problems

Insert your content here.

21 Tasks

21.1 Project Planning

Insert your content here.

21.2 Planning of the Development Phases

Insert your content here.

22 Migration to the New Product

22.1 Requirements for Migration to the New Product

Insert your content here.

22.2 Data That Has to be Modified or Translated for the New System

Insert your content here.

23 Costs

We don't expect any costs to arise during the development of the project. We will be using free tools and libraries for development and for hosting we will look to host databases and the back-end locally. In production the hosting will be done by the MES so it is out of the scope of this capstone.

24 User Documentation and Training

24.1 User Documentation Requirements

Insert your content here.

24.2 Training Requirements

Insert your content here.

25 Waiting Room

Insert your content here.

26 Ideas for Solution

Insert your content here.

Appendix — Reflection

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?

2. What pain points did you experience during this deliverable, and how did you resolve them?

3. How many of your requirements were inspired by speaking to your client(s) or their proxies (e.g. your peers, stakeholders, potential users)?

4. Which of the courses you have taken, or are currently taking, will help your team to be successful with your capstone project.

5. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

6. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?