

# Software Requirements Specification for Software Engineering: subtitle describing software

Team #18, Gouda Engineers

Aidan Goodyer

Jeremy Orr

Leo Vugert

Nathan Perry

Tim Pokanai

September 28, 2025

# Contents

<b>1</b>	<b>Purpose of the Project</b>	<b>vi</b>
1.1	User Business . . . . .	vi
1.2	Goals of the Project . . . . .	vi
<b>2</b>	<b>Stakeholders</b>	<b>vii</b>
2.1	Client . . . . .	vii
2.2	Customer . . . . .	vii
2.3	Other Stakeholders . . . . .	vii
2.4	Hands-On Users of the Project . . . . .	viii
2.5	Personas . . . . .	viii
2.6	Priorities Assigned to Users . . . . .	viii
2.7	User Participation . . . . .	viii
2.8	Maintenance Users and Service Technicians . . . . .	viii
<b>3</b>	<b>Mandated Constraints</b>	<b>viii</b>
3.1	Solution Constraints . . . . .	viii
3.2	Implementation Environment of the Current System . . . . .	viii
3.3	Partner or Collaborative Applications . . . . .	viii
3.4	Off-the-Shelf Software . . . . .	ix
3.5	Anticipated Workplace Environment . . . . .	ix
3.6	Schedule Constraints . . . . .	ix
3.7	Budget Constraints . . . . .	x
3.8	Enterprise Constraints . . . . .	x
<b>4</b>	<b>Naming Conventions and Terminology</b>	<b>xi</b>
4.1	Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project . . . . .	xi
<b>5</b>	<b>Relevant Facts And Assumptions</b>	<b>xi</b>
5.1	Relevant Facts . . . . .	xi
5.2	Business Rules . . . . .	xi
5.3	Assumptions . . . . .	xi
<b>6</b>	<b>The Scope of the Work</b>	<b>xi</b>
6.1	The Current Situation . . . . .	xi
6.2	The Context of the Work . . . . .	xi
6.3	Work Partitioning . . . . .	xi

6.4	Specifying a Business Use Case (BUC)	xi
<b>7</b>	<b>Business Data Model and Data Dictionary</b>	<b>xii</b>
7.1	Business Data Model	xii
7.2	Data Dictionary	xii
<b>8</b>	<b>The Scope of the Product</b>	<b>xii</b>
8.1	Product Boundary	xii
8.2	Product Use Case Table	xii
8.3	Individual Product Use Cases (PUC's)	xii
<b>9</b>	<b>Functional Requirements</b>	<b>xii</b>
9.1	Functional Requirements	xii
<b>10</b>	<b>Look and Feel Requirements</b>	<b>xii</b>
10.1	Appearance Requirements	xii
10.2	Style Requirements	xiii
<b>11</b>	<b>Usability and Humanity Requirements</b>	<b>xiii</b>
11.1	Ease of Use Requirements	xiii
11.2	Personalization and Internationalization Requirements	xiii
11.3	Learning Requirements	xiii
11.4	Understandability and Politeness Requirements	xiii
11.5	Accessibility Requirements	xiii
<b>12</b>	<b>Performance Requirements</b>	<b>xiii</b>
12.1	Speed and Latency Requirements	xiii
12.2	Safety-Critical Requirements	xiv
12.3	Precision or Accuracy Requirements	xiv
12.4	Robustness or Fault-Tolerance Requirements	xiv
12.5	Capacity Requirements	xiv
12.6	Scalability or Extensibility Requirements	xiv
12.7	Longevity Requirements	xv
<b>13</b>	<b>Operational and Environmental Requirements</b>	<b>xv</b>
13.1	Expected Physical Environment	xv
13.2	Wider Environment Requirements	xv
13.3	Requirements for Interfacing with Adjacent Systems	xv
13.4	Productization Requirements	xv

13.5 Release Requirements . . . . .	xv
<b>14 Maintainability and Support Requirements</b>	<b>xv</b>
14.1 Maintenance Requirements . . . . .	xv
14.2 Supportability Requirements . . . . .	xvi
14.3 Adaptability Requirements . . . . .	xvi
<b>15 Security Requirements</b>	<b>xvi</b>
15.1 Access Requirements . . . . .	xvi
15.2 Integrity Requirements . . . . .	xvi
15.3 Privacy Requirements . . . . .	xvi
15.4 Audit Requirements . . . . .	xvi
15.5 Immunity Requirements . . . . .	xvi
<b>16 Cultural Requirements</b>	<b>xvi</b>
16.1 Cultural Requirements . . . . .	xvi
<b>17 Compliance Requirements</b>	<b>xvii</b>
17.1 Legal Requirements . . . . .	xvii
17.2 Standards Compliance Requirements . . . . .	xvii
<b>18 Open Issues</b>	<b>xvii</b>
<b>19 Off-the-Shelf Solutions</b>	<b>xvii</b>
19.1 Ready-Made Products . . . . .	xvii
19.2 Reusable Components . . . . .	xvii
19.3 Products That Can Be Copied . . . . .	xvii
<b>20 New Problems</b>	<b>xvii</b>
20.1 Effects on the Current Environment . . . . .	xvii
20.2 Effects on the Installed Systems . . . . .	xvii
20.3 Potential User Problems . . . . .	xviii
20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product . . . . .	xviii
20.5 Follow-Up Problems . . . . .	xviii
<b>21 Tasks</b>	<b>xviii</b>
21.1 Project Planning . . . . .	xviii
21.2 Planning of the Development Phases . . . . .	xviii

<b>22 Migration to the New Product</b>	<b>xviii</b>
22.1 Requirements for Migration to the New Product . . . . .	xviii
22.2 Data That Has to be Modified or Translated for the New System	xviii
<b>23 Costs</b>	<b>xix</b>
<b>24 User Documentation and Training</b>	<b>xix</b>
24.1 User Documentation Requirements . . . . .	xix
24.2 Training Requirements . . . . .	xix
<b>25 Waiting Room</b>	<b>xix</b>
<b>26 Ideas for Solution</b>	<b>xix</b>

## Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

# 1 Purpose of the Project

## 1.1 User Business

The business of the user is to use the system to aid in the creation and analysis of experiments relating to a large but fragmented data set of rat trials that investigate compulsive behaviour of rats and how they relate to various factors such as drug injections or brain lesions. This data set is comprised of videos of rats moving on a flat surface, files of their (x,y) coordinates and plots that show the trajectories of these coordinates. Users may want to extract a concise set of pre-existing data that can be applicable to their hypothesis to streamline their workflow and avoid the necessity of carrying out trials. Users may also want to use the system to provide analysis on the data set as a means of further supplementing their academic work. This can include drawing behavioural metrics of the rats based on various parameters (i.e. rate of compulsion based on injection type) or data visualizations based similarly on compulsive behaviours as they relate to attributes of the rats.

*Insert your content here.*

## 1.2 Goals of the Project

The high-level goal of the project is to provide users in the field of behavioural sciences, a unified and non-technical way of drawing from the vast amount of available data on OCD rat trials to aid in their academic work and experiments. This will be accomplished via the following sub-goals:

- Create a DBMS that provides query access to the entirety of the data set of rat trials as well as all metadata associated with the data.
- Provide a UI that makes querying this data approachable to non-technical users by incorporating familiar and intuitive filtering and searching techniques (Attribute filters, Natural Language search bar). Further the UI must provide users with pre-generated query options that are likely to be useful for users that may not know where to start. This could include selecting all rat trials of a particular drug injection along with all saline control data.
- Provide an algorithm that can identify compulsive behaviour by looking at each rat trial and provide options for behavioural metrics and data

visualizations based on various attributes that the user can extract for their purposes.

*Insert your content here.*

## **2 Stakeholders**

### **2.1 Client**

The clients for this project are Dr. Henry Szechtman and Dr. Anna Dvorkin-Gheva, two professors at McMaster University who are experts in the fields of Psychiatry and Behavioural Neurosciences and Bioinformatics, respectively. They have worked extensively with these OCD rat trials and were responsible for generating and depositing this data in the FRDR repository where it currently is stored. Dr. Szechtman and Dr. Dvorkin-Gheva want to drastically improve the availability of the vast data set they have created so that the trials they conducted can become useful to other students and academics in related fields. *Insert your content here.*

### **2.2 Customer**

Since our application is intended to be open-source and accessible to the public for free, our customers will simply be a group of end-users of our application. Our end users will primarily involve behavioural neuroscience researchers like Dr. Henry Szechtman and Dr. Anna Dvorkin-Gheva, as well as graduate students and lab members, who will all benefit from the user-friendly and accessible functionality of the platform. Additionally we will have data scientists as customers, who will benefit from our application architecture and offered functionality for their purposes. Lastly, collaborating institutions and organizations from the open research community will be users of our application, since the application is intended to be free-use and accessible to all. *Insert your content here.*

### **2.3 Other Stakeholders**

*Insert your content here.*



## **2.4 Hands-On Users of the Project**

*Insert your content here.*

## **2.5 Personas**

*Insert your content here.*

## **2.6 Priorities Assigned to Users**

*Insert your content here.*

## **2.7 User Participation**

*Insert your content here.*

## **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.*

Collaborative System	System Overview
FRDR Repository	The Federated Research Data Repository is a 'bilingual bilingual publishing platform for sharing and preserving Canadian research data. It is a curated, general-purpose repository, custom built for large datasets.' This is where the data set of the rat trials is physically located and is dispersed across 29 independent datasets. Our system will provide a unified database schema but will pull data from this repository.
ratbat.mcmaster.ca	While not a directly collaborative system, this is a system made by a previous years' capstone team to address the same problem that our system seeks to. It will be a collaborative system in the sense that it provides a reference of potential ways to approach our solution, ideas that work well and can be carried forward and providing visibility to shortcomings of the system will help us to avoid repeating mistakes.

### 3.4 Off-the-Shelf Software

*Insert your content here.*

### 3.5 Anticipated Workplace Environment

*Insert your content here.*

### 3.6 Schedule Constraints

Due to the nature of the capstone course, the scheduling constraints map directly to dates in which major deliverable related to the project are due in the capstone course. They are laid out below:

Project Milestone	Scheduling Constraint
Software Requirements Specification	Oct 6 2025
Verification and Validation Plan	Oct 27 2025
Design Document Revision -1	Nov 10 2025
Proof of Concept Demonstration	Nov 17-28 2025
Design Documentation Revision 0	Jan 19 2025
Revision 0 Design Demonstration	Feb 2-13 2025
Verification and Validation Report	Mar 9 2025
Extras (Performance Report + User Manual)	Mar 9 2025
Final System Demonstration	Mar 23-29 2025
Final Documentation	April 6 2025

### 3.7 Budget Constraints

There is very limited budget available for this project. The department of Computing and Software at McMaster University will provide \$125 CAD for approved expenses. Outside of that funding, we are asked not to exceed spending of \$500 CAD culmulatively as a team. Thus, the total budget constraints for this project are \$500 CAD in total and \$375 of our team's personal funding.

### 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**

*Insert your content here.*

## **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**

### **6.1 The Current Situation**

*Insert your content here.*

### **6.2 The Context of the Work**

*Insert your content here.*

### **6.3 Work Partitioning**

*Insert your content here.*

### **6.4 Specifying a Business Use Case (BUC)**

*Insert your content here.*

## **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**

*Insert your content here.*

## **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**

*Insert your content here.*

### **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**

- The system shall return all query results within 2 seconds for result sets of up to 5,000 records.
- All network requests shall have latency below 100 ms under normal operating conditions.

## **12.2 Safety-Critical Requirements**

- There are no safety-critical operations for this system, as it handles only user interface and data processing with no direct impact on human safety.

## **12.3 Precision or Accuracy Requirements**

- All query results shall be accurate and filtered according to user specifications.
- There shall be no inconsistencies between the source database and query results.

## **12.4 Robustness or Fault-Tolerance Requirements**

- The system shall log and report all input errors without crashing.
- The backend shall handle errors gracefully, storing detailed logs for debugging and monitoring purposes.

## **12.5 Capacity Requirements**

- The system shall support up to 250 concurrent users and handle 5,000 transactions per hour.
- The database and associated controllers shall be able to manage up to 11 TB of data.

## **12.6 Scalability or Extensibility Requirements**

- The system shall allow additional modules to be integrated without requiring major changes to existing features.
- Under heavy user loads, the system shall maintain at least 80% of its performance efficiency.

## **12.7 Longevity Requirements**

- The system shall be designed to operate reliably for at least 5 years, with multiple teams and developers able to maintain and extend it.
- The system shall be compatible with any operating system when running offline.

## **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**

*Insert your content here.*



## **14.2 Supportability Requirements**

*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**

*Insert your content here.*

### **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**

*Insert your content here.*

### **19.2 Reusable Components**

*Insert your content here.*

### **19.3 Products That Can Be Copied**

*Insert your content here.*

## **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**

*Insert your content here.*

## **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?