

# Software Requirements Specification for Software Engineering: subtitle describing software

Team #11, OKKM Insights

Mathew Petronilho

Oleg Glotov

Kyle McMaster

Kartik Chaudhari

October 7, 2024

# Contents

<b>1</b>	<b>Purpose of the Project</b>	<b>vi</b>
1.1	Goals of the Project . . . . .	vi
1.1.1	High Data Accuracy . . . . .	vi
1.1.2	Ease of use . . . . .	vi
1.1.3	Minimizing Cost to Analyze Images . . . . .	vi
1.1.4	Results Returned Within Appropriate Timeframe . . . . .	vii
1.1.5	High System Reliability and Accessibility . . . . .	vii
<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 . . . . .	vii
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	Implementation Environment of the Current System . . . . .	viii
3.2	Partner or Collaborative Applications . . . . .	viii
3.3	Off-the-Shelf Software . . . . .	ix
3.4	Anticipated Workplace Environment . . . . .	ix
3.5	Schedule Constraints . . . . .	ix
3.6	Budget Constraints . . . . .	ix
<b>4</b>	<b>Naming Conventions and Terminology</b>	<b>x</b>
4.1	Glossary of All Terms, Including Acronyms, Used by Stakeholders involved in the Project . . . . .	x
<b>5</b>	<b>Relevant Facts And Assumptions</b>	<b>x</b>
5.1	Relevant Facts . . . . .	x
5.2	Business Rules . . . . .	x
5.3	Assumptions . . . . .	x

<b>6</b>	<b>The Scope of the Work</b>	<b>x</b>
6.1	The Current Situation . . . . .	x
6.2	The Context of the Work . . . . .	x
6.3	Work Partitioning . . . . .	x
6.4	Specifying a Business Use Case (BUC) . . . . .	x
<b>7</b>	<b>Business Data Model and Data Dictionary</b>	<b>xi</b>
7.1	Business Data Model . . . . .	xi
7.2	Data Dictionary . . . . .	xi
<b>8</b>	<b>The Scope of the Product</b>	<b>xi</b>
8.1	Product Boundary . . . . .	xi
8.2	Product Use Case Table . . . . .	xi
8.3	Individual Product Use Cases (PUC's) . . . . .	xi
<b>9</b>	<b>Functional Requirements</b>	<b>xi</b>
9.1	Functional Requirements . . . . .	xi
<b>10</b>	<b>Look and Feel Requirements</b>	<b>xi</b>
10.1	Appearance Requirements . . . . .	xi
10.2	Style Requirements . . . . .	xii
<b>11</b>	<b>Usability and Humanity Requirements</b>	<b>xii</b>
11.1	Ease of Use Requirements . . . . .	xii
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 . . . . .	xiii
12.3	Precision or Accuracy Requirements . . . . .	xiii
12.4	Robustness or Fault-Tolerance Requirements . . . . .	xiii
12.5	Capacity Requirements . . . . .	xiii
12.6	Scalability or Extensibility Requirements . . . . .	xiv
12.7	Longevity Requirements . . . . .	xiv

<b>13 Operational and Environmental Requirements</b>	<b>xiv</b>
13.1 Expected Physical Environment . . . . .	xiv
13.2 Wider Environment Requirements . . . . .	xiv
13.3 Requirements for Interfacing with Adjacent Systems . . . . .	xiv
13.4 Productization Requirements . . . . .	xiv
13.5 Release Requirements . . . . .	xiv
<b>14 Maintainability and Support Requirements</b>	<b>xiv</b>
14.1 Maintenance Requirements . . . . .	xiv
14.2 Supportability Requirements . . . . .	xv
14.3 Adaptability Requirements . . . . .	xv
<b>15 Security Requirements</b>	<b>xv</b>
15.1 Access Requirements . . . . .	xv
15.2 Integrity Requirements . . . . .	xvi
15.3 Privacy Requirements . . . . .	xvii
15.4 Audit Requirements . . . . .	xvii
15.5 Immunity Requirements . . . . .	xviii
<b>16 Cultural Requirements</b>	<b>xviii</b>
16.1 Cultural Requirements . . . . .	xviii
<b>17 Compliance Requirements</b>	<b>xviii</b>
17.1 Legal Requirements . . . . .	xviii
17.2 Standards Compliance Requirements . . . . .	xviii
<b>18 Open Issues</b>	<b>xviii</b>
<b>19 Off-the-Shelf Solutions</b>	<b>xix</b>
19.1 Ready-Made Products . . . . .	xix
19.2 Reusable Components . . . . .	xix
19.3 Products That Can Be Copied . . . . .	xix
<b>20 New Problems</b>	<b>xix</b>
20.1 Effects on the Current Environment . . . . .	xix
20.2 Effects on the Installed Systems . . . . .	xix
20.3 Potential User Problems . . . . .	xix
20.4 Limitations in the Anticipated Implementation Environment That May Inhibit the New Product . . . . .	xx

20.5 Follow-Up Problems . . . . .	xx
<b>21 Tasks</b>	<b>xx</b>
21.1 Project Planning . . . . .	xx
21.2 Planning of the Development Phases . . . . .	xx
<b>22 Costs</b>	<b>xx</b>
<b>23 User Documentation and Training</b>	<b>xx</b>
23.1 User Documentation Requirements . . . . .	xx
23.2 Training Requirements . . . . .	xx
<b>24 Waiting Room</b>	<b>xxi</b>
<b>25 Ideas for Solution</b>	<b>xxi</b>

## Revision History

Date	Version	Notes
10/1/2024	Mathew Petronilho	Added Purpose Of Project
Date 2	1.1	Notes

# 1 Purpose of the Project

There is currently a lack of high-quality, labeled satellite imagery datasets tailored for specific use cases. Many industries require specialized data for tasks like disaster response, environmental monitoring, urban planning, or defense, but building these datasets manually is time-consuming, costly, inefficient and may require expert data analysis. This hinders the development and deployment of accurate computer vision models for critical use cases across these various industries.

The purpose of this project is to create an online platform that accelerates this process and brings simplicity to satellite imagery data analysis.

## 1.1 Goals of the Project

### 1.1.1 High Data Accuracy

The system should have high classification accuracy for objects reported in the images. The core problem this system must solve is extracting useful information from the provided images. One key metric to determine the utility of the information found, is the classification accuracy of objects identified in the images. If the system is not able to determine what is contained in an image, it will not be useful to stakeholders.

### 1.1.2 Ease of use

The system should be very easy for stakeholders to use. There should be very low friction for users to classify images and objects found within images, with minimal training. It should also be simple for users to upload images to be analyzed. To maximize the information gained from users who are contributing to classification efforts, the system must ensure it is simple for users to get started with, and continue using the system. This is necessary to build a large enough user base, which will make it more likely to get insights in an acceptable amount of time.

### 1.1.3 Minimizing Cost to Analyze Images

The system should minimize the cost for users request insights from images. This could be implemented through intelligent algorithms for task delegation. Users of the system who upload images are interested in getting an

appropriate return for their investment. If the cost to analyze is too high, the platform will not retain a sufficiently large user base of purchasers.

#### **1.1.4 Results Returned Within Appropriate Timeframe**

The system should ensure the time it takes to obtain information from images is within a specified limit, as determined by users who upload images. Purchasers will have some time limit they require the system to process images within. To ensure timing needs are met, the system should provide realistic timelines and stick to them.

#### **1.1.5 High System Reliability and Accessibility**

The system should be useable remotely for purchasers and labellers, and have minimal downtime. The system should allow purchasers to upload images without being physically located where the system is hosted to ensure flexibility of use. The same should also be true for labellers, as they should be able to perform their tasks remotely. In both cases, the system should have low down time as to not introduce additional friction into the completion of tasks.

## **2 Stakeholders**

### **2.1 Client**

*Insert your content here.*

### **2.2 Customer**

*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

- The solution must be fully compatible with the latest stable releases of Google Chrome, Firefox, Microsoft Edge, and Safari browsers.

**Rationale:** Users will interact with the web application through various modern web browsers, so ensuring cross-browser compatibility is essential for providing a consistent user experience.

**Fit Criteria:** The web application must display consistently, maintain full functionality, and support core features across all specified browsers without major visual or functional discrepancies. Testing should be conducted on each browser to validate compatibility.

## 3.1 Implementation Environment of the Current System

There is no current environment in which our application must be implemented.

## 3.2 Partner or Collaborative Applications

There are no constraints regarding external applications that must be used alongside our product.

### 3.3 Off-the-Shelf Software

There is no required off-the-shelf software that must be used for our application.

### 3.4 Anticipated Workplace Environment

There is no particular location where users are required to work and use the product. As a web application, it can be accessed from most computers with an internet connection. We do not anticipate that the users' environment will physically constrain their ability to use the app in any way.

### 3.5 Schedule Constraints

- The proof of concept for this project must be ready to demonstrate by **November 11, 2024**. Not meeting this deadline will result in uncertainty about overcoming major risks associated with the project.
- The first project demonstration must be ready by **February 3, 2025**. Missing this deadline will reduce the time available to make refinements based on feedback and findings.
- The final demonstration must be ready by **March 24, 2025**. Missing this milestone would prevent the project from being presented and result in a significant loss of marks.

To see other documentation deadlines related to this project, refer to our [Development Plan](#).

### 3.6 Budget Constraints

- The project budget must not exceed \$750. All funds will be sourced from the team itself.

## **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

**Requirement LF1:**

- **Description:** The application shall adapt to various screen sizes, ensuring legibility and an uncluttered layout.

- **Rationale:** Users will have computers with varying screen sizes, so a consistent experience across all these sizes is ideal.
- **Fit Criterion:** Visual elements must not exceed the boundaries of a screen with a size between the range 1024×768 pixels to 1920×1080 pixels.

#### **Requirement LF2:**

- **Description:** Interactive elements such as buttons shall provide visual feedback to the user.
- **Rationale:** This will allow users a better understanding of when their actions have been processed by the application.
- **Fit Criterion:** Every interactive element changes colour or displays additional visual cues, such as animations or shadows, to indicate interaction.

## **10.2 Style Requirements**

#### **Requirement LF3:**

- **Description:** The application should maintain a unified visual design across all components.
- **Rationale:** A consistent appearance enhances the application's cohesiveness and conveys a professional aesthetic.
- **Fit Criterion:** Font type, sizing, and colour, along with background tones are all consistent throughout the application.

## **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**

*Insert your content here.*

## **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**

*Insert your content here.*

## **12.5 Capacity Requirements**

*Insert your content here.*

## **12.6 Scalability or Extensibility Requirements**

*Insert your content here.*

## **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**

*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

### Requirement SR1:

- **Description:** The application shall only allow users with labeling access, including labellers, customers, and admins, to view active projects and label images.
- **Rationale:** We do not want random users with no stake in the process to effect the results.
- **Fit Criterion:** Users who have not logged in to the application have no way of viewing projects or labeling images. Users logged in as labellers, customers, or admins have access to these features.

### Requirement SR2:

- **Description:** The application shall only allow users with customer access and above to create new image analysis projects.
- **Rationale:** Unidentified users creating projects would be impossible to facilitate. Also, labellers have no need to access project creation.
- **Fit Criterion:** Users who have not logged in to the application have no way of creating an image analysis project. Users logged in as customers or admins have access to these features.



#### Requirement SR3:

- **Description:** The application shall validate the email format the user provides when creating an account.
- **Rationale:** We do not want users using invalid emails to sign up.
- **Fit Criterion:** Let  $E$  represent the set of all email addresses, and let  $V$  represent the set of all valid email addresses. A valid email address conforms to the general pattern:

$$V = (\forall \textit{email} \in E \mid \textit{email matches the pattern } [a-zA-Z0-9+_{-}]+@[a-zA-Z0-9_{-}]+[a-zA-Z])$$

#### Requirement SR4:

- **Description:** The application shall validate the password format the user provides when creating an account.
- **Rationale:** We do not want users using weak passwords to sign up.
- **Fit Criterion:** Let  $P$  represent the set of all passwords, and let  $V$  represent the set of all valid passwords. A valid password has a at least one lowercase, uppercase, number and special character and is a minimum of 8 characters in length:

$$V = (\forall \textit{password} \in P \mid \textit{password matches the pattern } (?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*[!$%\&])[a-zA-Z0-9!$%\&]{8,})$$

## 15.2 Integrity Requirements

#### Requirement SR5:

- **Description:** The application shall prevent incorrect data from being introduced.
- **Rationale:** The database of information should always reflect correct and up to date information.

- **Fit Criterion:** The system must validate user inputs for data accuracy and format before they are saved. Any invalid data must trigger error messages, preventing it from being entered into the database. Users must be required to correct errors before proceeding.

## 15.3 Privacy Requirements

### Requirement SR6:

- **Description:** User data will be securely encrypted to protect user's privacy.
- **Rationale:** This will help to avoid user's being compromised if a data leak occurs.
- **Fit Criterion:** An encryption algorithm is used on sensitive user data such as passwords.

### Requirement SR7:

- **Description:** The application shall ensure that all payment transactions are processed securely using encryption and comply with relevant security standards, such as PCI-DSS, which helps to protect payment account data (PCI Security Standards Council, 2024).
- **Rationale:** Protecting users' financial information is critical to maintaining trust. Failing to secure payments can lead to data breaches, financial loss, and legal liabilities.
- **Fit Criterion:** All payment transactions must use industry-standard encryption to protect sensitive data. Payment information, such as credit card details, must not be stored locally on the application and must be processed via a secure, PCI-DSS-compliant third-party payment gateway.

## 15.4 Audit Requirements

These requirements are not applicable as we are not an organization that is currently subject to audits.

## 15.5 Immunity Requirements

### Requirement SR8:

- **Description:** The application shall use parameterized queries or prepared statements for all database interactions.
- **Rationale:** We want to prevent SQL injection attacks which can lead to unauthorized data access or manipulation.
- **Fit Criterion:** All database queries must be implemented using parameterized queries or prepared statements. Dynamic SQL strings that concatenate user input must not be used in the codebase.

## 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

**Task Assignment Algorithm:** To ensure labelers are engaged as they complete tasks and to obtain the highest quality of information possible, the system must implement an intelligent task allocation system. This system has not yet been determined.

**Label Consensus Algorithm:** Similarly, the algorithm for combining multiple user labels into one accurate label has not yet been determined.

**Labeling Services Offered:** The system has determined several potential labeling services to be offered by the system, but has not confirmed with certainty what will be included. This will be determined after more research has been completed on the task assignment algorithm.

## 19 Off-the-Shelf Solutions

### 19.1 Ready-Made Products

**Amazon Mechanical Turk:** A web-based crowdsourcing platform. Instead of building a novel front end, the system could obtain labels through this platform instead.

### 19.2 Reusable Components

**Label Studio:** A React library which contains components for building a web-based data annotation platform.

### 19.3 Products That Can Be Copied

**Tolka AI:** A general purpose image label crowdsourcing site. Supports image segmentation, bounding box drawing, and more computer vision labeling tasks.

## 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 Costs**

*Insert your content here.*

# **23 User Documentation and Training**

## **23.1 User Documentation Requirements**

*Insert your content here.*

## **23.2 Training Requirements**

*Insert your content here.*

## **24    Waiting Room**

*Insert your content here.*

## **25    Ideas for Solution**

*Insert your content here.*

## References

1. PCI Security Standards Council. (2024, May 13). *PCI Security Standards Council – Protect Payment Data with Industry-driven Security Standards, Training, and Programs*. <https://www.pcisecuritystandards.org/standards/pci-dss/>

## Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. 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.
2. 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?