

# Software Requirements Specification

for

Visulyfe

Version 1.0 approved

Prepared by Adam Dixon, Fernando Serrano Perez, Geovanny Montano, Jacob Hertz,  
Nicholas Trigueros, Ryan Torrez, Steven Partida, Washika Afrozi

Visual Corp.

03/06/2024

# Table of Contents

Table of Contents.....	pg # 2
Revision History.....	pg # 3
1. Introduction.....	pg # 4
1.1. Purpose.....	pg # 4
1.2. Intended Audience and Reading Suggestions.....	pg # 4
1.3. Product Scope.....	pg # 4
1.4. Definitions, Acronyms, and Abbreviations .....	pg # 5
1.5. References.....	pg # 5
2. Overall Description.....	pg # 6
2.1. System Analysis.....	pg # 6
2.2. Product Perspective.....	pg # 7
2.3. Product Functions.....	pg # 8
2.4. User Classes and Characteristics.....	pg # 9
2.5. Operating Environment.....	pg # 10
2.6. Design and Implementation Constraints.....	pg # 10
2.7. User Documentation.....	pg # 11
2.8. Assumptions and Dependencies.....	pg # 11
2.9. Apportioning of Requirements.....	pg # 11
3. External Interface Requirements.....	pg # 12
3.1. User Interfaces.....	pg # 12
3.2. Hardware Interfaces.....	pg # 12
3.3. Software Interfaces.....	pg # 12
3.4. Communications Interfaces.....	pg # 13
4. Requirements Specification.....	pg # 14
4.1. Functional Requirements.....	pg # 14
4.2. Interface Requirements.....	pg # 14
4.3. Logical Database Requirements.....	pg # 17
4.4. Design Constraints.....	pg # 17
5. Other Nonfunctional Requirements.....	pg # 18
5.1. Performance Requirements.....	pg # 18
5.2. Safety Requirements.....	pg # 18
5.3. Security Requirements.....	pg # 18
5.4. Software Quality Attributes.....	pg # 19
5.5. Business Rules.....	pg # 19
6. Legal and Ethical Considerations.....	pg # 20
Appendix A: Glossary.....	pg # 21
Appendix B: Analysis Models.....	pg # 22-25

# Revision History

Name	Date	Reason For Changes	Version
Geovanny Montano	5/2/2024	Software interface additions (3.3)	v0.9.1
Steven Partida	5/3/2024	Product Function Diagram (updated)	v0.9.2
Steven Partida	5/3/2024	User Classes and Characteristics	v0.9.3
Adam Dixon	5/3/2024	Table of contents	v1.0
Adam Dixon	5/7/2024	Edits	v1.0.1

<Add rows as necessary when the document is revised. This document should be consistently updated and maintained throughout your project. If ANY requirements are changed, added, removed, etc., immediately revise your document.>

# 1. Introduction (Washika)

## 1.1 Purpose

- **To a diverse set of users:** For example, the senior management of the organization that is paying for the system to the engineers responsible for developing the software.
- **To find the level of detail:** In a requirements document depends on the type of system that is being developed and the development process used.
- **To safety and security:** Here some critical systems need detailed requirements because safety and security have to be analyzed in detail to find possible requirements errors. When the system is to be developed by a separate company the system specifications need to be detailed and precise.
- **To predict the system evolution:** The standard has been extended to include information about predicted system evolution. This information helps the maintainers of the system and allows designers to include support for future system features

## 1.2 Intended Audience and Reading Suggestions

While the software requirement specification (SRS) document is written for a more general audience, this document is intended for individuals directly involved in the development of software developers, project consultants, and team managers.

The user requirements for a system should describe the functional and nonfunctional requirements so that they are understandable by system users who don't have detailed technical knowledge. They should specify only the external behavior of the system. The requirements document should not include details of the system architecture or design. Consequently, if you are writing user requirements, they should not use software jargon, structured notations, or formal notations. They should write user requirements in natural language, with simple tables, forms, and intuitive diagrams.

## 1.3 Product Scope

- Software Requirements Specification (SRS) can ultimately help we make vital decisions on our product's lifecycle:
- **A structural manner:** Software Requirements Specification (SRS) is a kind of agreement between the customer and the company.
- **To Clarity and Understanding:** The SRS provides a common communication channel between stakeholders, It reduces ambiguity and the risks of misunderstandings among team members in clearly defining requirements.
- **Basis for Project Planning:** The planning for software development projects must be particularly detailed. which allows teams to accurately estimate costs and schedules.
- **Guidance for Development Teams:** The SRS is used to guide developers in their understanding of the functional specifications and how the software will be designed which enable developers to build a system that meets with client expectations.

- Facilitates Testing and Quality Assurance: SRSs are used as references by quality assurance and testing teams. It becomes much more of a systematic procedure and the chance that an important functionality is overlooked are reduced.

## **1.4 Definitions, Acronyms, and Abbreviations**

- Complete: The SRS should be complete.
- Correct: SRS should be correct to the needs of the customer.
- Clear: It should be clearly declared.
- Accurate: SRS should have accuracy. If this is not accurate then the software cannot be developed.
- Verifiable: Software Requirements Specification Should be verifiable. The requirements are verified by experts and testers.
- Modifiable: All the requirements specified in the SRS document should be modifiable.
- Traceable: It should be traceable, that is, each requirement in it.
- Testable: SRS should be testable in any way.

## **1.5 References**

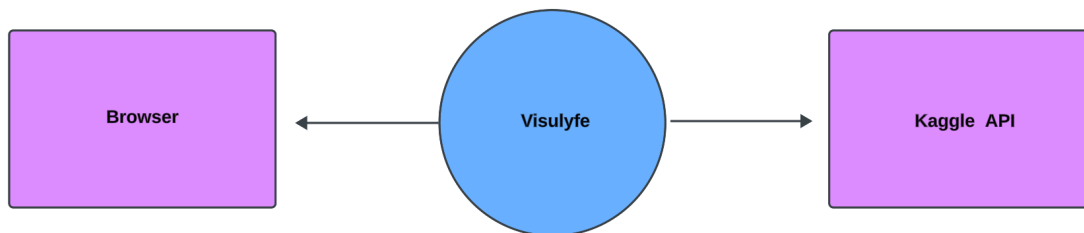
- Test Plan - This will be used to verify if the software is meeting all requirements before the deadline.
- As reference we will also be using DataWrapper visualizer tool which can greatly inspire and help shape the software, Visuallyfe <https://www.datawrapper.de>
- Visulfy Use Case Scenarios document
- Various papers on Data Visualization Techniques

## 2. Overall Description

### 2.1 System Analysis

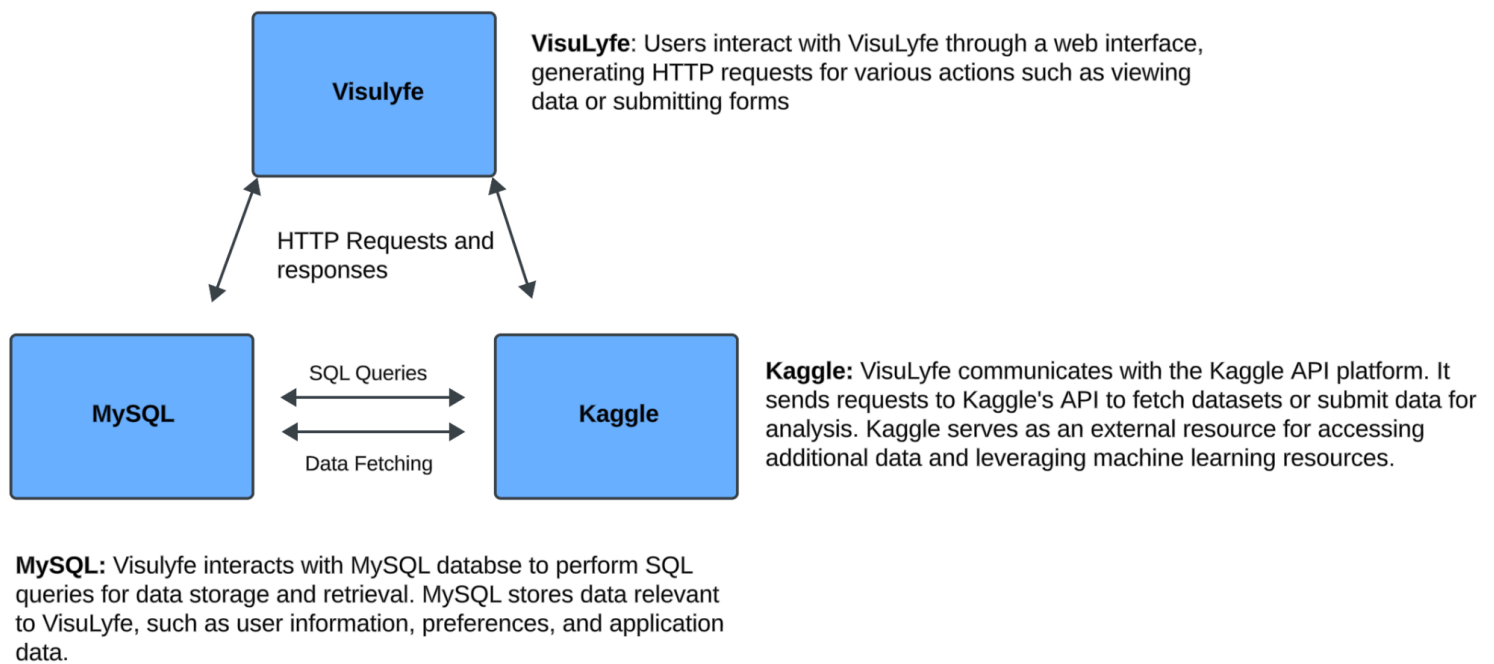
- **Project Goals:**
  - Streamline the process of transforming raw datasets into visually appealing representations on a user-friendly platform.
  - Automate the visualization process to minimize manual effort and time required for visualizing.
  - Simplify the data analysis process for users by providing an intuitive and user-friendly platform.
  - Offer customization for a better user experience
- **Major Technical Challenges and Solutions:**
  - **Data Processing**
    - **Solution:** Implement algorithms and techniques to handle various inconsistencies and ensure data quality before visualization.
  - **Automated Visualization Algorithms**
    - **Solution:** Conduct extensive research and experimentation to design adaptable algorithms that analyze dataset attributes efficiently.
  - **User Interface Design**
    - **Solution:** Employ user-centered design methods to iteratively refine the user interface while focusing on simplicity, clarity, and interactivity .
  - **Performance Optimization**
    - **Solution:** Utilize efficient algorithms and data structures, implement parallel processing techniques, and employ hardware acceleration whenever possible to improve computational efficiency.

### 2.2 Product Perspective



Kaggle is an online platform for data science and machine learning enthusiasts, offering datasets, competitions, and collaborative tools for data analysis and model development.

<2.2 Context Diagram>

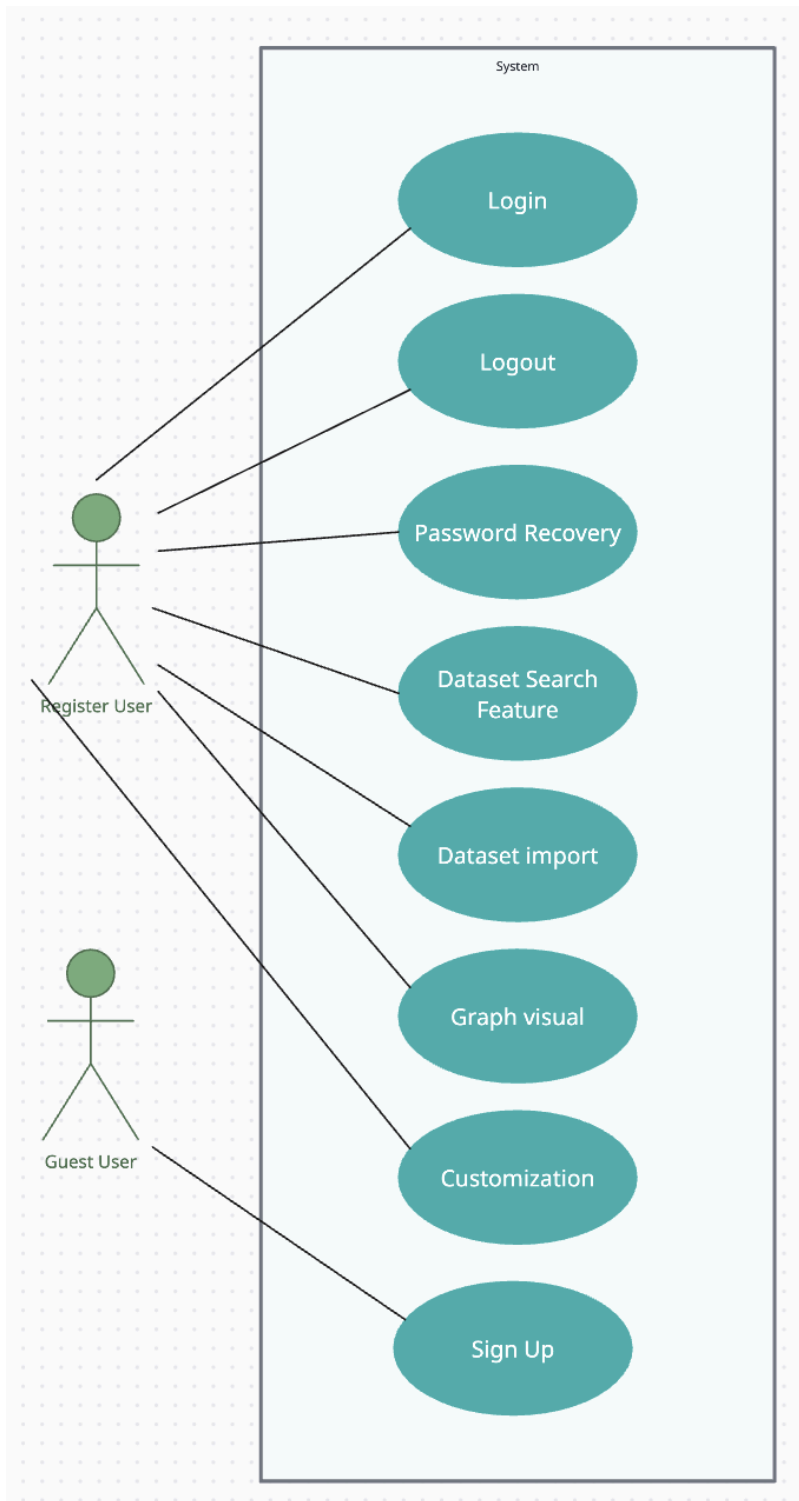


## <2.2 DFD-0 data flow diagram>

**Technical Nature:** Visulyfe is a standalone software designed to provide users with automated data visualization capabilities. It is not dependent on any other products or systems.

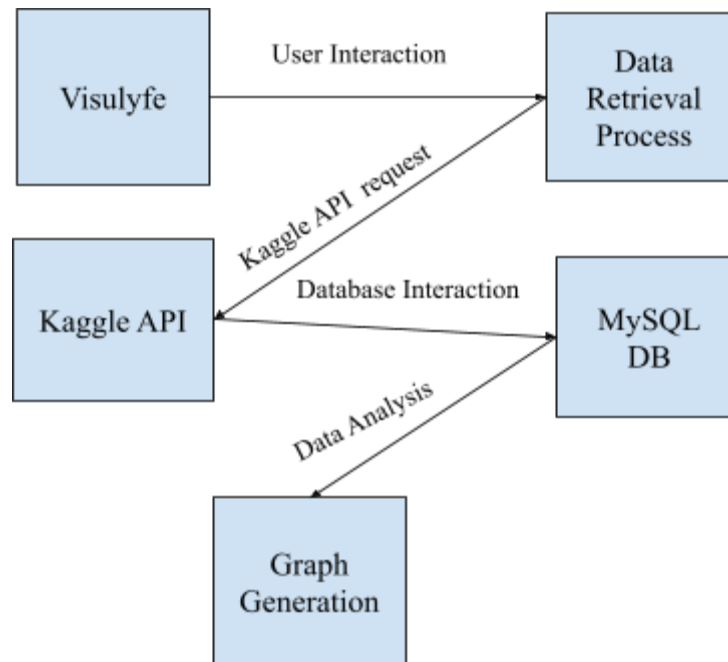
**Other Software Relationships:** While Visulyfe is designed as a standalone tool, it can integrate with larger systems that require data analysis and visualization capabilities. For example, it can be integrated into business intelligence platforms, data analytics software, or dashboarding solutions. Visulyfe serves as a component within these larger systems, providing users with advanced data visualization functionalities.

## 2.3 Product Functions (Steven)



**<2.3 Use Case Diagram>**





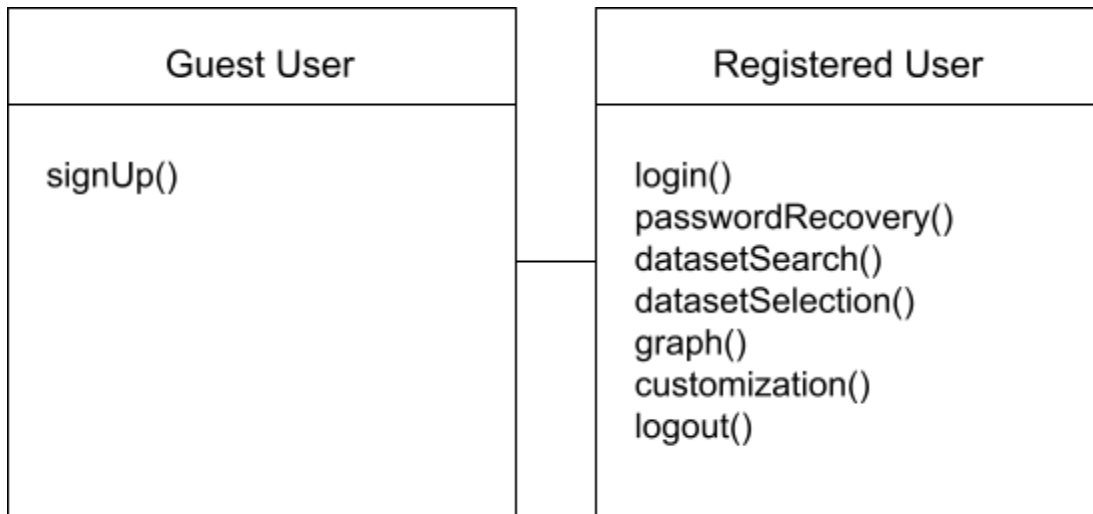
<2.3 DFD-1 Data Flow Diagram>

- **User Authentication and Authorization**
  - Allow users to register and log in securely
  - Manage user permissions
- **Data Import and Management**
  - Provide a feature to import data sheets
  - Allow users to manage and organize imported data sheets
- **Data Visualization**
  - Generate data visualizations of data sheets
  - Customization options for visualizations such as color, layout, etc
- **Security and Data Privacy**
  - Implement measures to ensure the security and privacy of user data
- **Performance Optimization**
  - Optimize the web app for performance to ensure fast loading times and smooth interactions

## 2.4 User Classes and Characteristics (Steven)

- **Guest User**
  - **Frequency of Use:** Moderate to frequent use
  - **Technical Expertise:** Basic to intermediate knowledge of data visualization concepts
  - **Privilege Levels:** Restricted access. Needs to create an account
  - **Experience:** Varied level of experience with data analysis
- **Registered User**
  - **Frequency of Use:** Moderate to frequent use

- **Technical Expertise:** Basic to intermediate knowledge of data visualization concepts
- **Privilege Levels:** Standard access to import data sets and analysis tools
- **Experience:** Varied level of experience with data analysis



<2.4 Class Diagram>

## 2.5 Operating Environment (Fernando)

The goal of Visualyfe is to design it in such a way that it can be used in a variety of environments such as schools, workplaces, or for research purposes. Since we plan on making it a web application, the software should be accessible as long as the user has an internet connection, access to google sheet or Excel spreadsheets, and a device to connect to the internet.

## 2.6 Design and Implementation Constraints (Fernando)

Several factors that can influence the implementation of the software hardware, compatibility, reliability and safety. The hardware may affect the software performance by being limited to the hardware's capabilities. Also, the software needs to be compatible with a variety of work environments to ensure that the user has a great user experience. Also, because we are working with data, we have to incorporate measures to ensure that data of the user is protected. Because of this we plan on being diligent with our research and development.

## 2.7 User Documentation (Fernando)

Along with the software, the team plans on adding a user manual with images and tutorials to show the user how to use the software. This in depth tutorial will be shown to users in the home page and be the first thing they see when using the application.

## 2.8 Assumptions and Dependencies (Adam)

The web application will be accessible on all operating systems as it operates through an internet connection. One third-party component that has not been previously mentioned but will be included here is the databases sourced from Kaggle, which will be utilized for demonstration purposes. These databases consist of publicly available datasets intended for user instruction. Additionally, other

third-party software utilized for Visulyfe will include open-source libraries such as Matplotlib, Plotly, and Seaborn, among others.

## **2.9 Apportioning of Requirements (Fernando)**

The project requirements are apportioned across four key components: backend development, data visualization, data acquisition, and data storage. Backend development involves setting up the Flask framework, creating endpoints, and managing server-side logic. Data visualization utilizes Plotly to generate interactive graphs based on user input and stored datasets. Data acquisition entails connecting to the Kaggle API, retrieving datasets, and processing them for visualization and storage. Data storage requires configuring an SQL database, creating necessary tables, and ensuring data integrity and security. Each component is assigned to specific teams or individuals with relevant expertise, and the project management strategy includes regular coordination and integration checks to ensure seamless collaboration and alignment with project goals.

## **3. External Interface Requirements(Geo)**

### **3.1 User Interfaces**

The primary interface for this application is a web-based platform accessible through standard browsers like Chrome, Firefox, and Safari. It is designed with responsive features, ensuring a seamless experience whether accessed from a desktop, laptop, tablet, or smartphone. The graphical user interface (GUI) allows users to select a topic and choose the type of graph they would like to generate. Users can also interact with these graphs, with options to zoom in or out and inspect specific data points.

Customization is a key feature, giving users the ability to tailor the appearance of graphs. They can set titles, labels, colors, and even choose the chart type that best suits their needs. To ensure a smooth experience, clear and responsive error messages are displayed if any issue arises during graph generation.

To aid users in navigating the application, a comprehensive Help section is included, offering guidance on how to use the various features. Additionally, a feedback mechanism with pop-up messages keeps users informed about the latest updates and bug fixes. Security is also a top priority, with users required to create an account with a unique username and password for secure access. This combination of features provides a robust and user-friendly experience.

### **3.2 Hardware Interfaces**

Users can interact with the application using various data input devices, including keyboards and mice, to navigate and select data within the interface. These devices offer intuitive control, allowing users to efficiently choose options, enter information, and interact with graphical elements.

The application supports multiple display devices, ensuring flexibility in how users view their work. It is designed to be displayed on screen monitors, providing a clear and detailed view of data and graphs. Additionally, it can be projected, allowing for presentations and group discussions in larger settings. This range of input and display options enhances user engagement and adaptability in different environments.

### 3.3 Software Interfaces

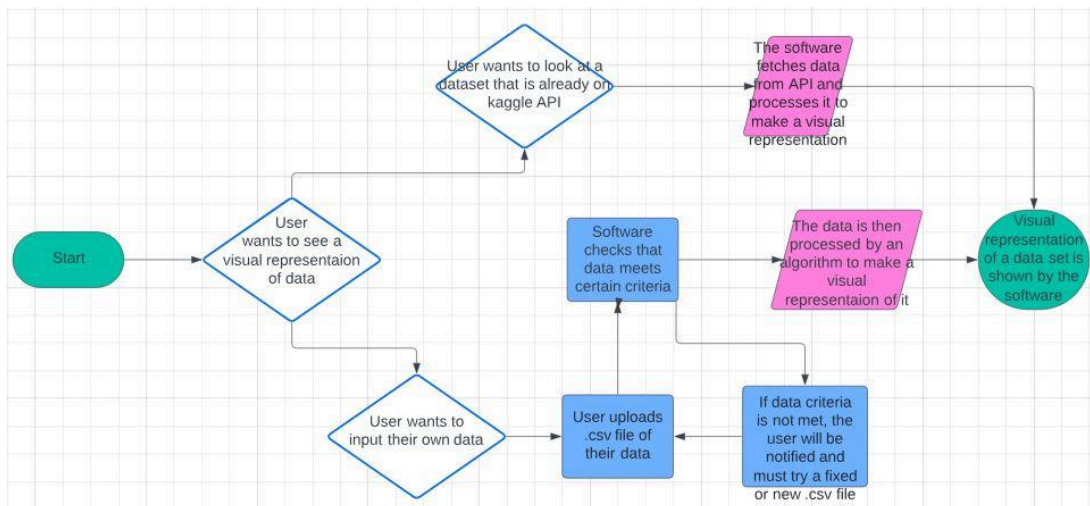
The user would retrieve datasets that are available through the endpoint and use that information for the creation of a customized graph.

### 3.4 Communications Interfaces

Users will input data into the application using electronic forms. These forms are designed to be user-friendly, allowing for efficient data entry. Once filled out, the forms can be submitted to the server, where the data is processed according to the application's logic. This seamless process facilitates user interaction with the system and supports various operations within the application.

The application also features email notifications to keep users informed about important updates or issues. When a significant change occurs or a user encounters a problem, an automated email is sent to notify them. This feature ensures users stay updated on the latest developments and provides a convenient way to receive information without having to constantly check the application.

## Requirements Specification



<4.1 DFD-2 Data Flow Diagram>

## 4.1 Functional Requirements (Fernando)

- Validity checks on the inputs.
  - The system shall perform validity checks on incoming data to ensure that the data meets specific criteria which will determine if the criteria is suitable to be processed.
- Exact sequence of operation
  - The system shall follow a predetermined sequence of operations to either fetch data from the API or process the inputs and generate a visual output.
- Responses to abnormal situation, such as
  - In an event such as overflow, the system shall be able to detect and notify the user of an error and give the user options to correct this system fault
  - The system shall handle errors in a way where it does not affect the user experience. In events of errors, the system should be able to identify and log the error. The system can then let the user try again or let the user know that there was an error.
  - The system shall accurately represent the data that was inputted/fetched , depending on if the user wants to input the data in or fetch data.

## 4.2 External Interface Requirements (Ryan)

### 4.2.1 User Interface Requirements

- Web Interface
  - Purpose: Visulyfe will be accessible via web browser. Thus, applications such as Safari, Chrome, OperaGX, or Edge will be necessary to test and study any changes made to Visulyfe.
  - Relationships: Primary stage for Visulyfe, thus everything is connected to the Web Interface.
- Responsive Design
  - Purpose: The user should be able to respond to a different shape environment, such as a screen from a desktop or laptop usage. Thus, the windows of Visulyfe must be dynamic and appropriately sized no matter the scale.
  - Relationships: Web Interface, GUI, Graph Display, Display Devices.
- Graphical User Interface (GUI)
  - Purpose: The GUI serves as the main way for a user to interact with Visulyfe. The graphs they see, the buttons they push, and other ways for them to interact with the site are all due to the GUI.
  - Output Destination: User Screen, Web Browser
  - Relationships: Web Interface, Responsive Design, GUI, Customization Options, Graph Display, Feedback Mechanism, Security
- Graph Display

- Purpose: The graphs generated by Visulyfe must be accurate and detail-oriented. Generated graphs should be able to zoom in and out while also being able to check for specific data.
- Input Source: User Provided
- Output Destination: User Screen, Web Browser
- Relationships: Web Interface, Responsive Design, GUI, Customization Options
- Customization Options
  - Purpose: Users should have the ability to customize the appearance of graphs, including titles, labels, colors, and chart types.
  - Input Source: User
  - Output Destination: User Screen, Web Browser
  - Relationships: Web Interface, Responsive Design, GUI, Graph Display
- Error Handling
  - Purpose: Should a user experience an error, they will be met with a clear and responsive error message when graphs are generated. Ideally, the system should be able to recognize what error has occurred and output an error code for reference.
  - Input Source: Error Handling
  - Output Destination: User Screen, Web Browser
  - Relationships: Web Interface, Responsive Design, GUI, Graph Display
- Help and Documentation
  - Purpose: Visulyfe should be able to provide a “Help” option in order for the user to be able to go through the web-application with ease. A reference manual must be made available.
  - Output Destination: User Screen, Web Browser
  - Relationships: Web Interface, Responsive Design, GUI, Customization Options, Graph Display, Feedback Mechanism, Security
- Feedback Mechanism
  - Purpose: Visulyfe should be able to give pop-messages to the user for informational purposes like up to date information on how bugs have been fixed and the updates that have been implemented.
  - Output Destination: User Screen, Web Browser
  - Relationships: Web Interface, Responsive Design, GUI
- Security
  - Purpose: The user will have an account for Visulyfe, which requires a username and password to log in to their account and use their work. Since this data is sensitive, encryption should be used while storing this data.
  - Input Source: User
  - Relationships: Web Interface, GUI

#### 4.2.2 Hardware Interface Requirements

- Data Input Devices
  - The user should be able to use devices such as keyboards and the mouse to navigate the app with ease, and select the data that they desire.

- Relationships: Web Interface, Responsive Design, GUI, Customization Options, Graph Display, Feedback Mechanism, Security
- Display Devices
  - If the user would like to have his work displayed on a screen, monitor, or a projector they should be able to do so.
  - Output Destination: User Screen, File Download(pdf, jpg, png, etc...)

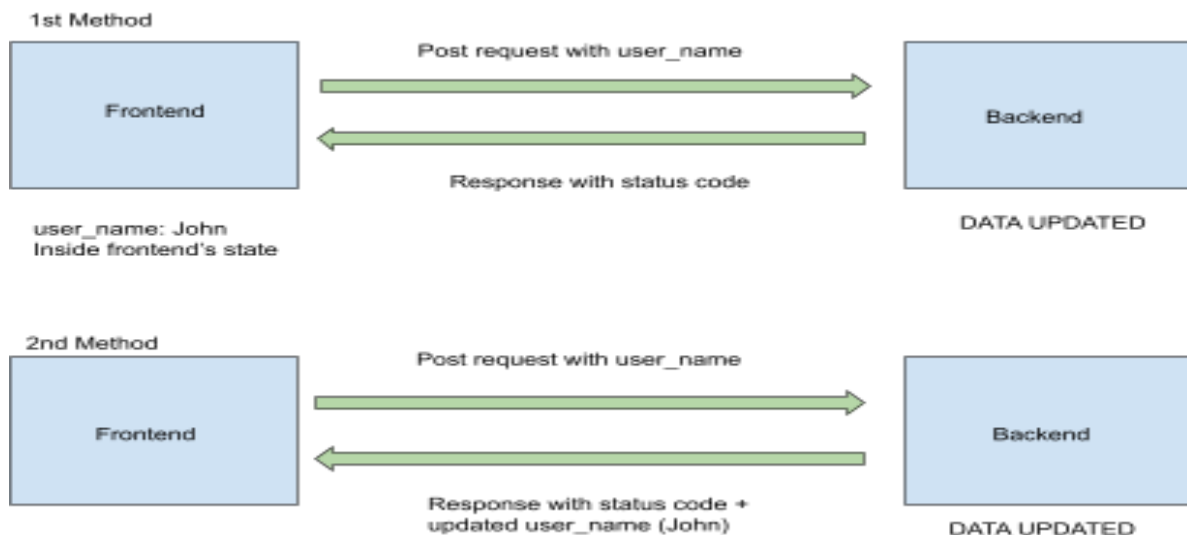
### 4.2.3 Software Interface Requirements

- Data Management System
  - MySQL
  - Version 8.0.36
  - Source: Microsoft
- Programming Language and Framework
  - Python
  - Version: 3.11
  - Source: Official Python website (<https://www.python.org/>)
- Web Framework
  - Flask
  - Version: 3.0.2
  - Source: Flask official website (<https://flask.palletsprojects.com/>)
- Kaggle API
  - Source: Kaggle API documentation (<https://www.kaggle.com/docs/api>)
- Plotly
  - Source : <https://github.com/plotly/plotly.py>

## **4.3 Logical Database Requirements (Ryan)**

The only constraint to a user would be that they would have to upload a .csv file to add any personal data sets to our database.

- Frequent use possible
- Access limited to personal account
- User data cannot be tampered with by outside users. User Login maintains integrity.
- Data retention of the user will be secured and maintained until the user's next session.



### <4.3 Static Class Diagram>

## 4.4 Design Constraints (Adam)

Visulyfe is a program that solely works with an internet connection and with browsers such as Safari, Chrome, OperaGX, and Edge. The web application does not have any hardware limitations that would be restrictive to specific operating systems. The addition of extra datasets from users can be created from external spreadsheet creation applications such as Excel spreadsheets and google sheets. The user would download the .csv file and then upload that file to be visualized.

## 5. Other Nonfunctional Requirements (Jacob)

### 5.1 Performance Requirements

- ~300-1500 visitors per day are expected to be the average volume of traffic on the app
  - especially when the app first launches
- If an account feature is completed, then the app will contain the users' username and password, and possibly email
- Otherwise, the data handled within the app will be inputted by the user, or gathered from other APIs to be converted into a graph

### 5.2 Safety Requirements

- Certifications are to be obtained to ensure the safety of the app and the users
- Domain Validated (DV) Certificate
  - an automated process to prove that we do, in fact, own the app/website
- Secure Sockets Layer (SSL)
  - encrypted connection between web server and browser



- Transport Layer Security (TLS)
  - eliminates security vulnerabilities in SSL
- Hypertext Transfer Protocol Secure (HTTPS)
  - uses SSL and TLS to encrypt data

### 5.3 Security Requirements

- Users will be able to make an account with username and password
  - Once created, we can optionally provide a two-factor authentication
- We can work towards getting a SSL or TLS certificate to use HTTPS with our domain and help encrypt user data
- General Data Protection Regulations (GDPR) guidelines state 7 principles to abide by:
  - Lawfulness, fairness, and transparency
    - data must be processed lawfully
  - Purpose limitation
    - data must only be collected for specific, legitimate purposes
  - Data minimisation
    - collected data must be adequate and limited to what is relevant
  - Accuracy
    - collected data must be necessary and up-to-date
  - Storage limitation
    - any unnecessary data can be deleted to make room
  - Integrity and confidentiality
    - data must be processed with security and confidentiality
  - Accountability
    - data collection must be compliant with the other principles

### 5.4 Software Quality Attributes

- Visulyfe will be a free app/software, so it will be available to all
- Visulyfe can be accessed from the app or web browser
- Maintenance will be performed at regular intervals, unless a bigger problem were to occur
  - users could also submit their reviews or concerns, which could help the dev team prioritize certain aspects to update first
- Only one graph can be made at a time, based on data inputted, or selected, by the user
  - a feature could be implemented for users who make an account with the app, so that they could either save graphs made previously, or make multiple graphs at once
- Data provided in the graphs will be provided by the user, or collected from other APIs
  - accuracy should not be a problem either way

### 5.5 Business Rules

- Dev team will make sure to fix any bugs that are present, even after launch of the app
- Users will access the app to make graphs with an assortment of data
- Users will possibly be able to have extra benefits when making an account with the app

## 6. Legal and Ethical Considerations (Ryan)

The Legal and Ethical concerns of Visulyfe are as follows:

- **Data Privacy and Protection:** Compliance with data privacy laws and regulations is essential to protect sensitive and personal information. Data visualization often involves accessing and analyzing large datasets, some of which may contain personally identifiable information (PII). Ignoring these laws can lead to data breaches, identity theft, reputational damage, and legal consequences. By adhering to them we will demonstrate our commitment to safeguarding data and respecting individual privacy rights.
- **Legal and Regulatory Obligations:** Different regions and industries have specific legal and regulatory requirements governing data handling and protection. For instance, the California Consumer Privacy Act (CCPA) outlines individuals' rights to control their personal information in California. Compliance with these regulations is not only a legal obligation but also a means to build trust with our customers.
- **Trust and Customer Confidence:** Data breaches can erode customer trust and confidence. Individuals are increasingly concerned about how their data is collected, stored, and used. Adhering to privacy regulations assures customers that their data is being handled responsibly and transparently. By demonstrating a commitment to transparency, we can establish and maintain strong relationships with our customers.
- **Risk Mitigation and Reputation Management:** Doing anything but following the regulations laid out by these laws will pose a significant risk to us, including financial, legal, and reputational risks. Regulatory authorities can impose hefty fines and penalties for violations, leading to financial losses. Moreover, data breaches and privacy scandals can tarnish an organization's reputation, resulting in decreased customer loyalty. By prioritizing compliance, we will mitigate these risks and ensure that we operate within legal boundaries, reducing the likelihood of legal and reputational damage.

# Appendix A: Glossary (Adam Dixon)

**Data Input Devices:** Devices like keyboards and mice used to input data into the application.

**Display Devices:** Hardware devices like monitors or projectors used to visualize the output of the application.

**Data Management System (MySQL):** A relational database management system (RDBMS) used to store and manage data efficiently.

**Form Submission:** The process where users input data into electronic forms, which is then sent to the server for processing.

**Email Notification:** Automatic emails sent to users to provide updates or inform them about issues encountered while using the web application.

**API:** stands for Application Programming Interface. It is a set of rules and protocols that allows different software applications to communicate with each other.

**SSL:** stands for Secure Socket Layer. It is a standard security protocol used to establish encrypted links between a web server and a browser in online communication.

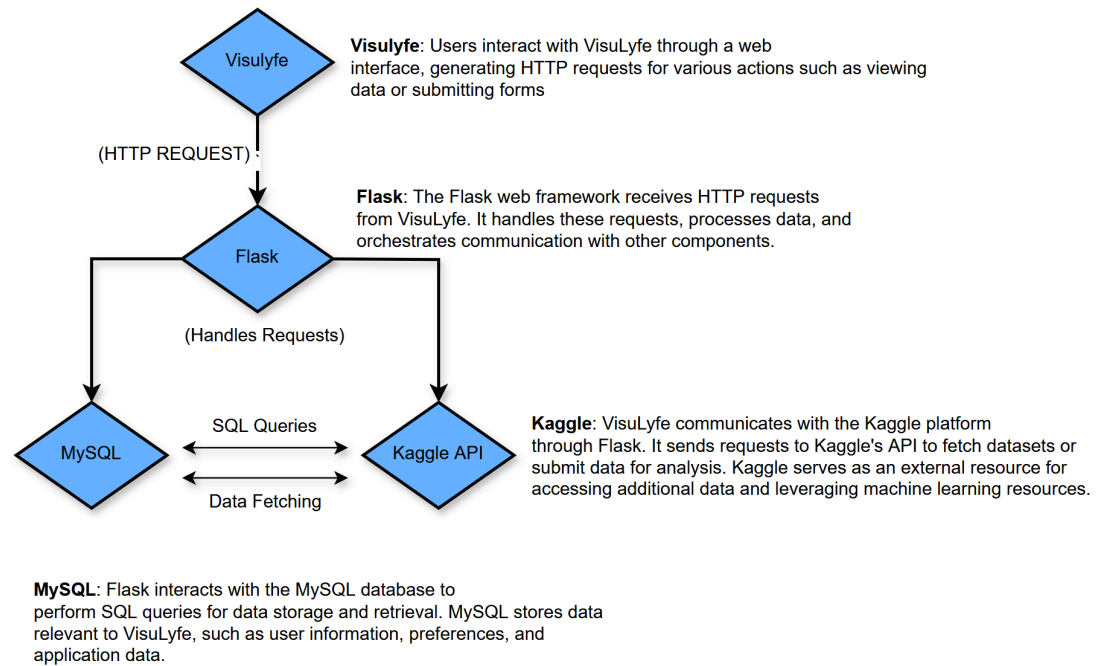
**TLS:** stands for Transport Layer Security. It is a cryptographic protocol that ensures privacy and data integrity between applications communicating over a computer network, commonly the internet.

**HTTPS:** stands for Hypertext Transfer Protocol Secure. It is a secure version of the Hypertext Transfer Protocol (HTTP), which is the protocol used for transmitting data over the World Wide Web.

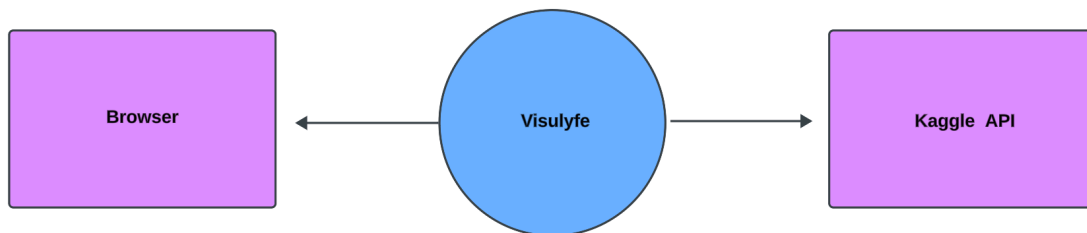
**Encrypt:** to convert information or data into a coded or scrambled format using a specific algorithm and a cryptographic key. The purpose of encryption is to enhance the security and confidentiality of the data.

**Visualization:** the presentation of information or data in a graphical or visual format, often to make complex information more understandable, accessible, and interpretable.

## Appendix B: Analysis Models

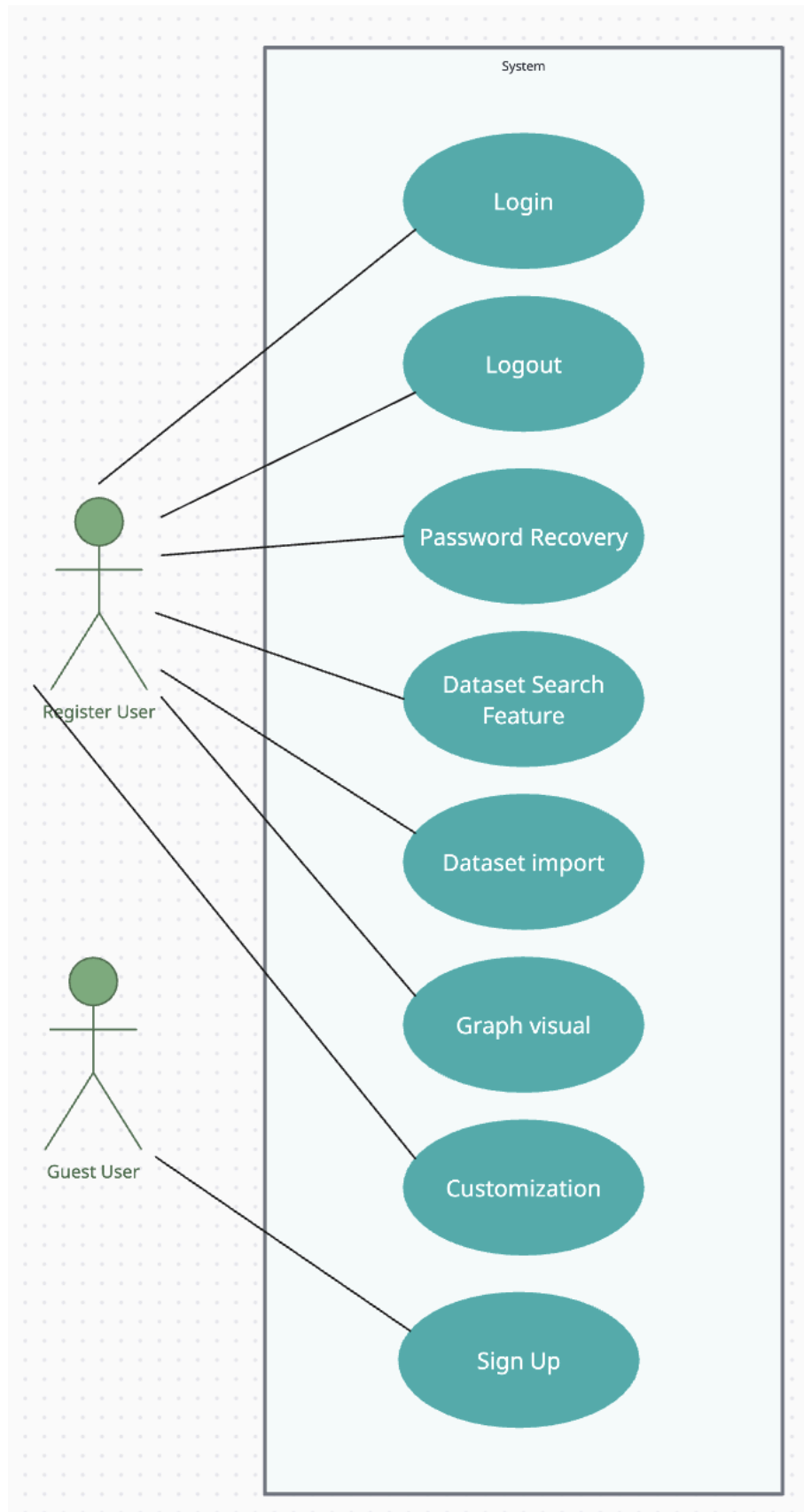


<2.2 DFD-0 data flow diagram page: 7>

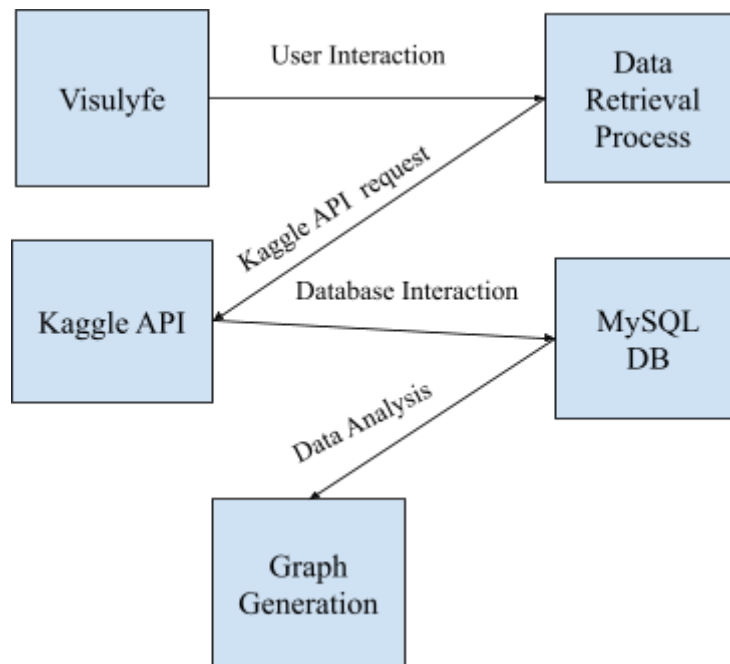


Kaggle is an online platform for data science and machine learning enthusiasts, offering datasets, competitions, and collaborative tools for data analysis and model development.

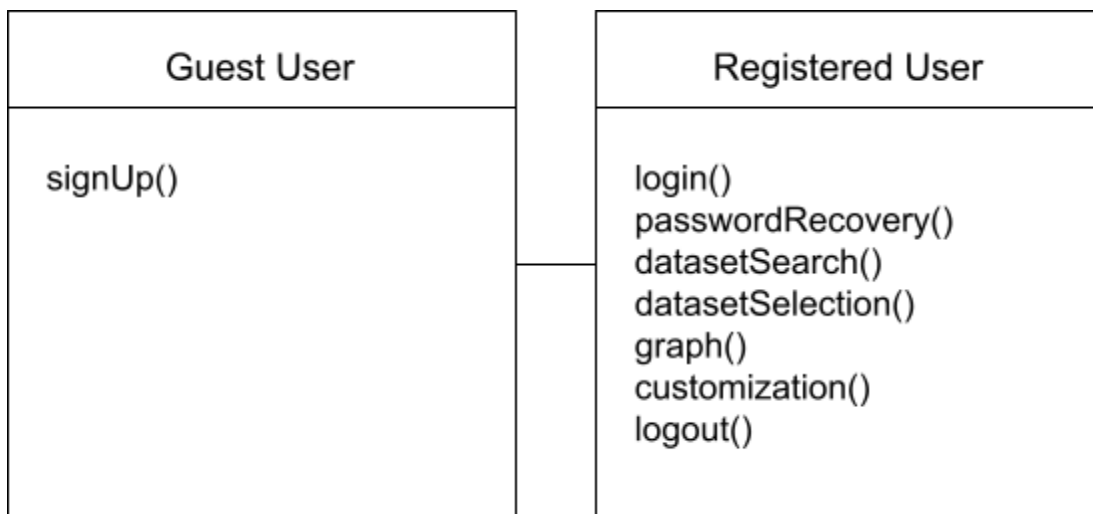
<2.2 context diagram page: 7>



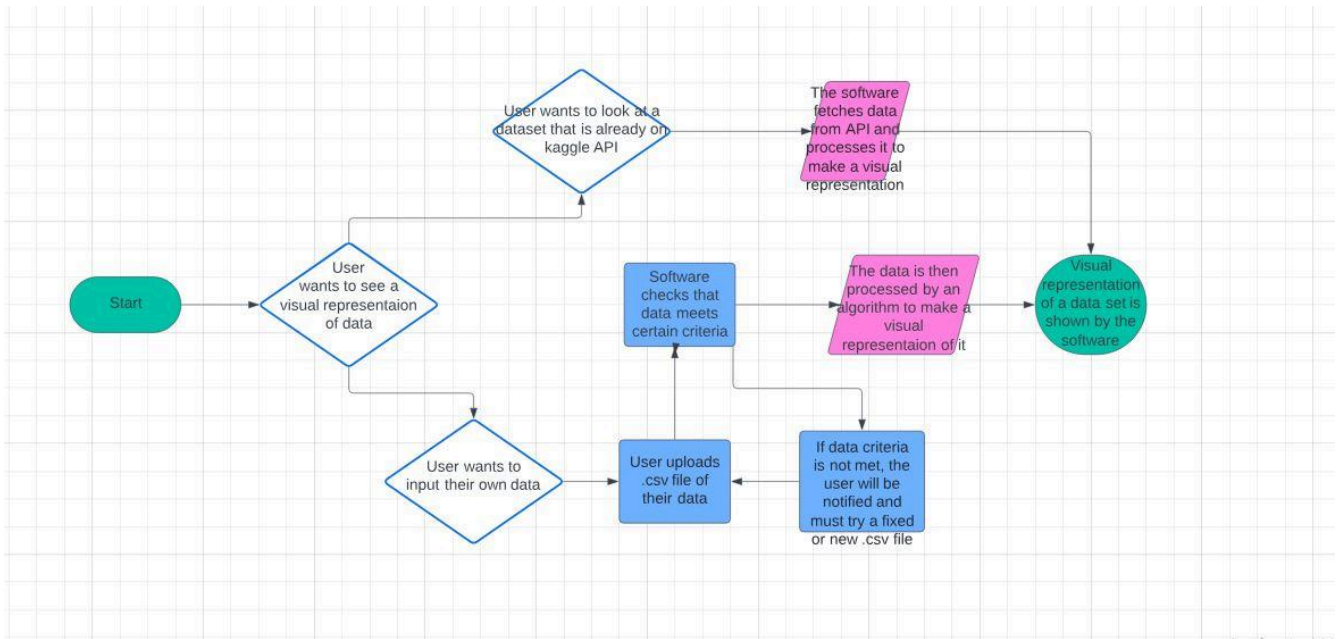
<2.3 Use case diagram page: 8>



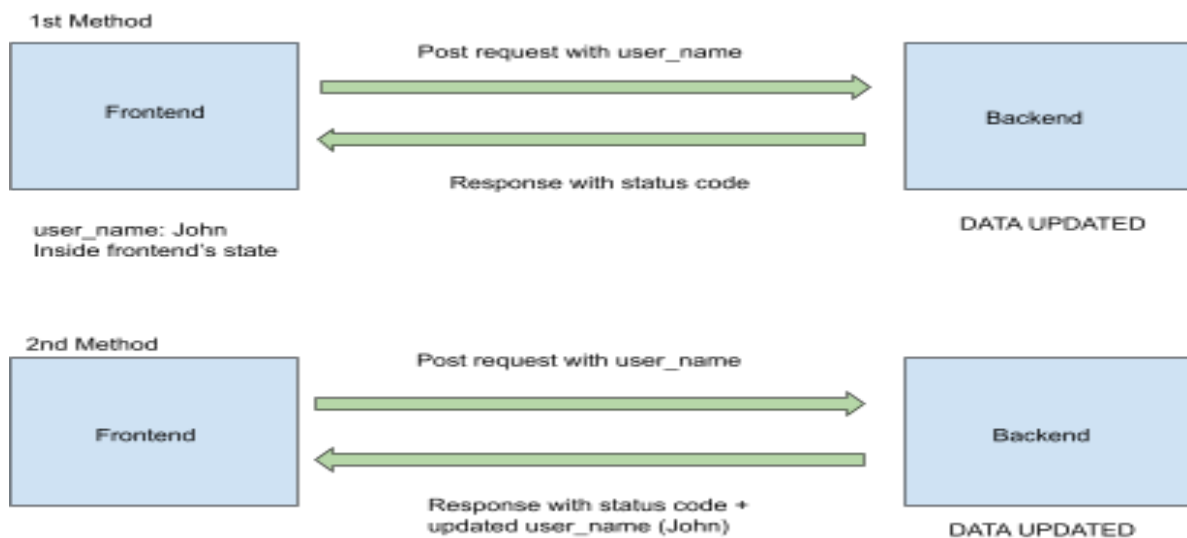
<2.3 DFD-1 data flow diagram page: 8>



<2.4 class diagram page: 10>



<4.1 DFD-2 data flow diagram page:13>



<4.3 static class diagram page:16>