Food Waste Prevention - Requirements

10/16/19

**Group 3**

Haelyn Brandt

Martin Edmunds

Neil Gayeta

Brandon Goza

**Requirements Definition**

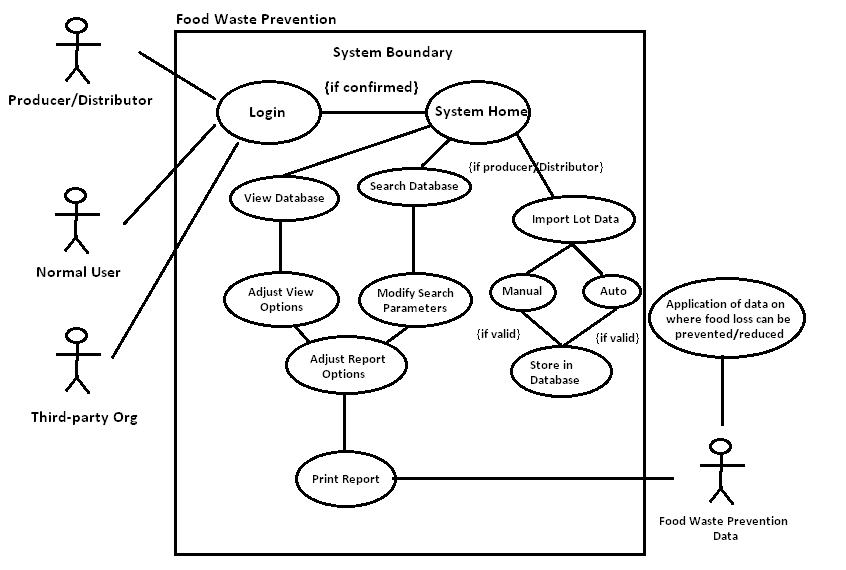
**Functional Requirements**

* The system will authenticate a user and their account type upon login
* The system will allow a user to import lot tracking information into a database
* The system will allow a user to register an account with the system
* The system will allow a user to choose between three different account types
  + Producer/distributor account
  + ‘Viewer’ account
  + Third-party organization account
* The system will allow a user who registered as a producer/distributor to import lot tracking data into the system
* The system will detect food loss events and record the following information:
  + Where a loss occurred
  + Reason for loss
  + How much was loss
  + Food edible state (edible/non-edible)
* The system will allow a viewer account to view shipment data based on the following criteria:
  + Region
  + Food type
  + Quantity of food type
  + Loss events
* The system will allow third-party organization account types to filter the view data and print out food-loss reports
* The system will allow third-party organization account types to have access to contact information of producers/distributors of reported food loss data

**Non-Functional Requirements**

* The system will authenticate user and their account type within 20 seconds
* The system will allow user to import lot tracking information to a database in > 1 minute
* The system will update shipment information within 10 minutes upon upload by the farmers/distributors
* The system will display requested data from the user within 20 seconds
* The system will display the imported in information of produced crop from the farmer/ producer within ±5% accuracy
* The system will display the distributed amount of crop to each region, retailers, and customers within ±10% accuracy
* The system will simulate amount of shipment to be sent to per region based on food loss information
* The system will simulate ways to reduce food loss by testing shipment amounts and ways to recycle edible food
* The system will display the amount of wasted food compared to amount distributed to each region, retailers, and customers within ±10% accuracy
* The system will report any region, retailers, or customers that requested over shipment based on food loss analyzation.
* The system will detect food loss events and record/update required information twice a day
* The system will automatically scan/upload/update any new/changed information every 30 minutes

**Use Cases Diagram**



**Use Cases**

**Use Case 1: Automatic Lot Input**

Actors:

* User with account type: Producer/Distributor

Preconditions:

* User must be authenticated with the system
* User must have account privileges to modify database
* User must have lot-tracking information formatted in JSON or CSV type format

Postconditions:

* Properly formatted data has been inputted into the system’s database
* User receives feedback that data has been inputted successfully, or has failed
* If input has failed, the user is asked to reformat data and try again.

Flow of Events

* Upon having an account of type: Producer/Distributor, the user inputs their creditientals.
* The system confirms user credentials and directs the user to the system’s home page
* The system displays an option for users of the necessary account type: “Import Tracking Data”
* The user will select ‘Import Tracking Data’ and be directed to a page where lot information can be inputted both manually and via JSON or CSV file.
* If the user elects to upload via JSON or CSV, the system will parse the data and import the information to the database component of the system
* After a successful import, the system should display success and the user should be allowed to search for the previous imported lot.

**Use Case 2: View/Analyze/Report Food Waste Data and Trends**

Actors:

* User with account type: Normal User
* Any interested individuals
  + Academic researchers
  + Industry analysts
  + Local food distribution coordinators

Preconditions:

* User must be authenticated with the system
* User must be able to retrieve data from the system

Postconditions:

* Properly formatted data and reports have been viewed/downloaded by the user.
* Availability of information has increased awareness of food waste.
* Analysis of reports/data has led to identification of hunger and food waste situations and trends.
* User feedback prompts actions taken to address food waste inefficiencies.

Flow of Events

* Upon having an account of type: Normal User, the user inputs their creditientals.
* The system confirms user credentials and directs the user to the system’s home page.
* The system displays options for viewing interpreted data via maps and reports as well as retrieval of raw data in JSON or CSV format or via a web API.
* The system will provide custom datasets/reports/maps based on parameters specified by the user.
* The analysis of this information leads to increased awareness of and/or insight into the food waste/hunger situation.
* Feedback from the user can lead to action being taken by producers, distributors, and interested organizations to better serve the needs of the hungry.

**Use Case 3: Utilize Food Waste**

Actors:

* Users with the account type: Third-Party Organization
* Regional/Local Non-Profit Organizations

Preconditions:

* User has a device to access webpage
* User has the training and/or experience to analyze visuals that provide a mapping of regional lot/distribution and corresponding food loss data
* User is part of or affiliated with an organization that provides services to people that need food resources
* User has the resources to perform potential pick-up and distribution of otherwise wasted food

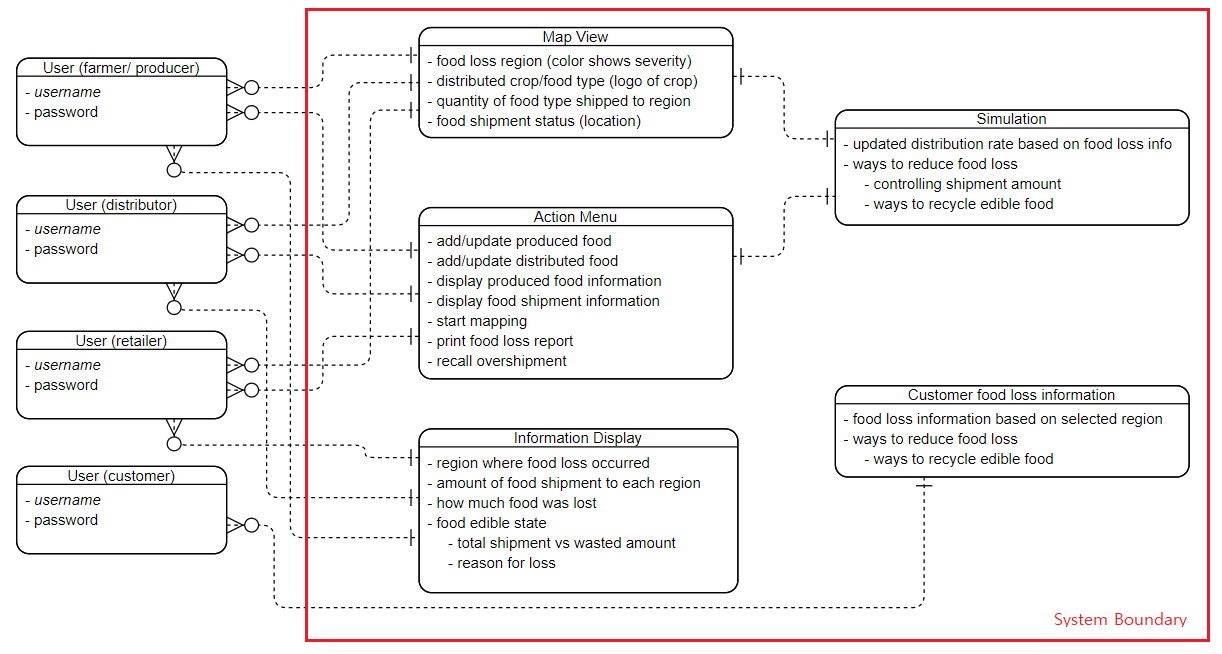
Postconditions:

* Wasted food based on lot/distribution map has been identified that can be of use for people in need
* Identified wasted food has been picked up and distributed by organization
* The distribution of this food has lowered the overall food waste and contributed to the slowing down of rising food costs
* Distribution of wasted food contributed more data that will aid in the process of identifying people that may need food in subsequent cases

Flow of Events:

* User authentication occurs, the user is directed to input their credentials under the account type: Third-Party Organization
* Confirmation of user credentials and directs user to the homepage
* The system provides the option for the user to view a mapping of lot/distribution and their corresponding food loss data
* The system allows the user to filter the view of the lot/distribution data
* The user has the option to print or download the specified view of data they would like to consume
* The user may contact producer/distributor who provided the input via provided phone number or email within the map view to potentially pursue action in retrieving edible wasted food for potential pick-up and distribution

**UML Class Diagram**



**Requirements Specification**

**Functional Requirements**

* The system with authenticate a user with a database lookup query
* The system will import lot tracking information via parsing a JSON or CSV formatted file
* The system will store user accounts in a database module of the system. Each account will have an associated ‘type’ in which access can be controlled
* Any user with a ‘producer/distributor’ account will be able to perform the JSON, CSV input of lot information
* Lot information being stored will include: weight of the lot before and after it leaves/enters any point in the shipping process. The system will detect loss events by periodically checking the difference between them
* Loss events will be specific entities contained in the database module. These events will be tracked and can be queried directly
* A web page will be devoted to allowing an account to change various settings when viewing lot information from the database
* The system will allow accounts with ‘third-party organization’ access to query the underlying database to find significant loss events, find where losses occur and when they occur.
* Accounts with ‘third-party organization’ will have access to information regarding supplies/distributors, which will also be tracked as entities in the database.

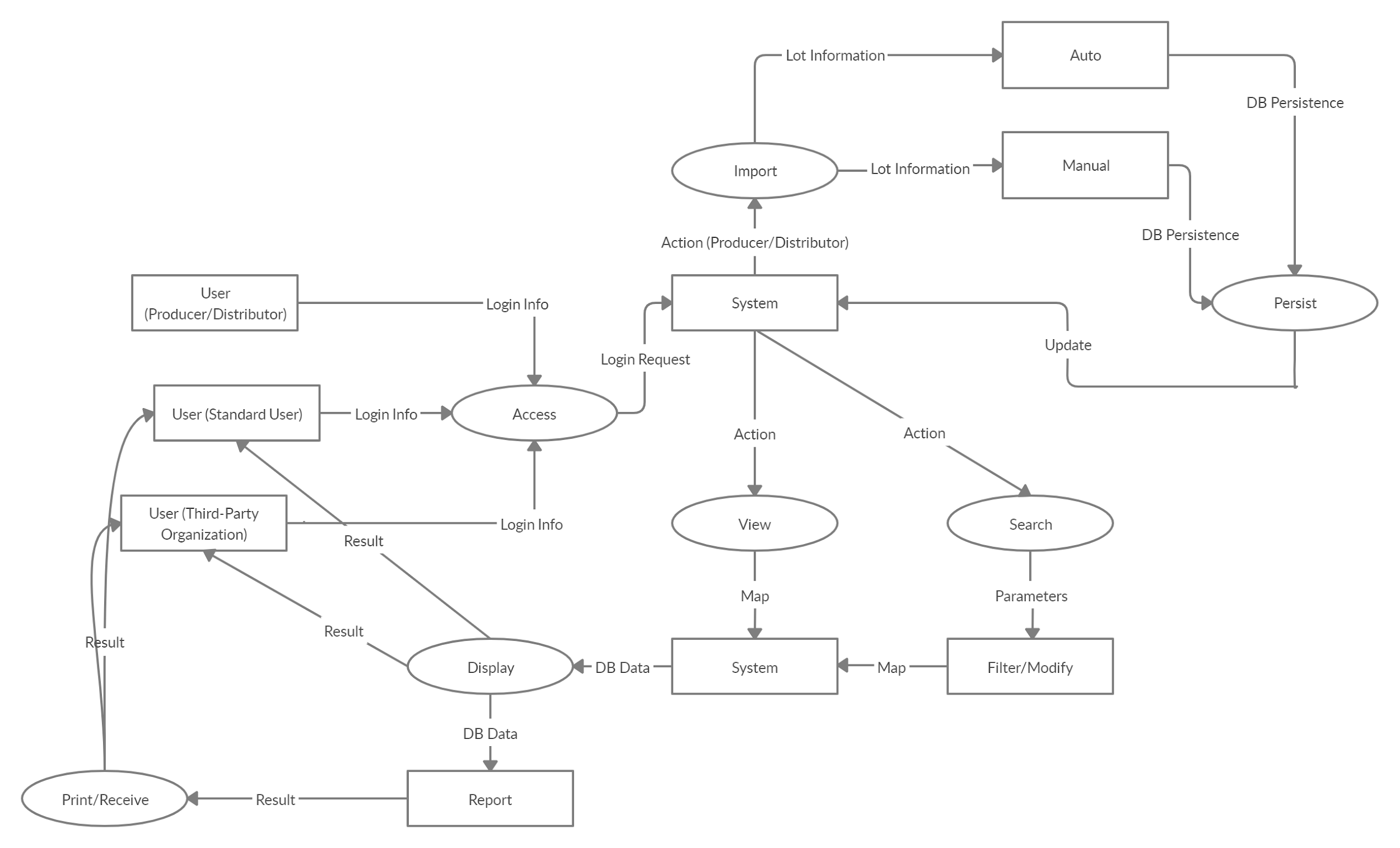
**Non-Functional Requirements**

* The system will query the database and return a response of user authentication and their account type within 20 seconds
* The system will query the database and allow user to import tracking information to database within 1 minute of user interaction
* Upon information upload by the farmers/distributors, the system will query the database and automatically update shipment information in less than 10 minutes
* The system will query the database and return a response of requested data by the user within 20 seconds
* The system will query the database and return a response/report of overall produced crop in system entered with 95% accuracy
* The system will query the database and return a response/report of distributed amount of crop to each region within 90% accuracy
* The system will query the database and simulate amount of shipment to be sent to per region based on food loss to produce best shipment rate, producing reports of reduced food loss up to 10% per region.
* The system will query the database and analyze the best way to recycle edible food in each region 1 week after every shipment
* The system will query the database and return a response of wasted food compared to shipment amount based on region, retailers, and customers producing a report within 90% accuracy
* The system will automatically query the database and report any over shipment based on analyzed food loss percentage based on each region within 10 minutes and recall or change shipment location
* The system will automatically query the database and detect food loss events and update information twice a day (beginning and end)

**Data-Flow Diagram**

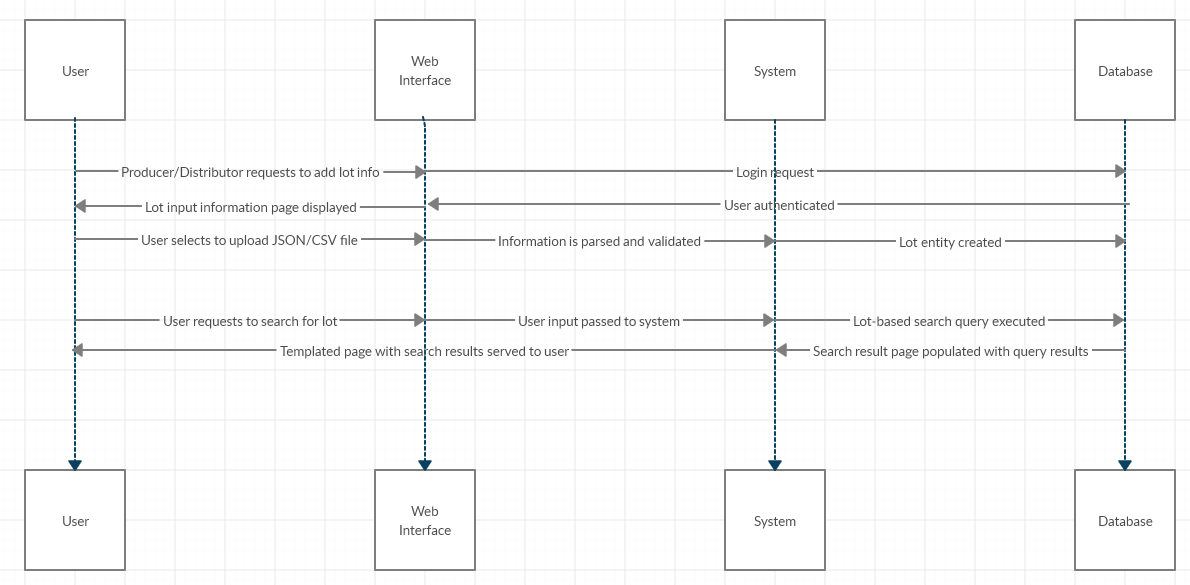
Showing how the system relates to the entities in the environment.

**Data Flow Diagram**

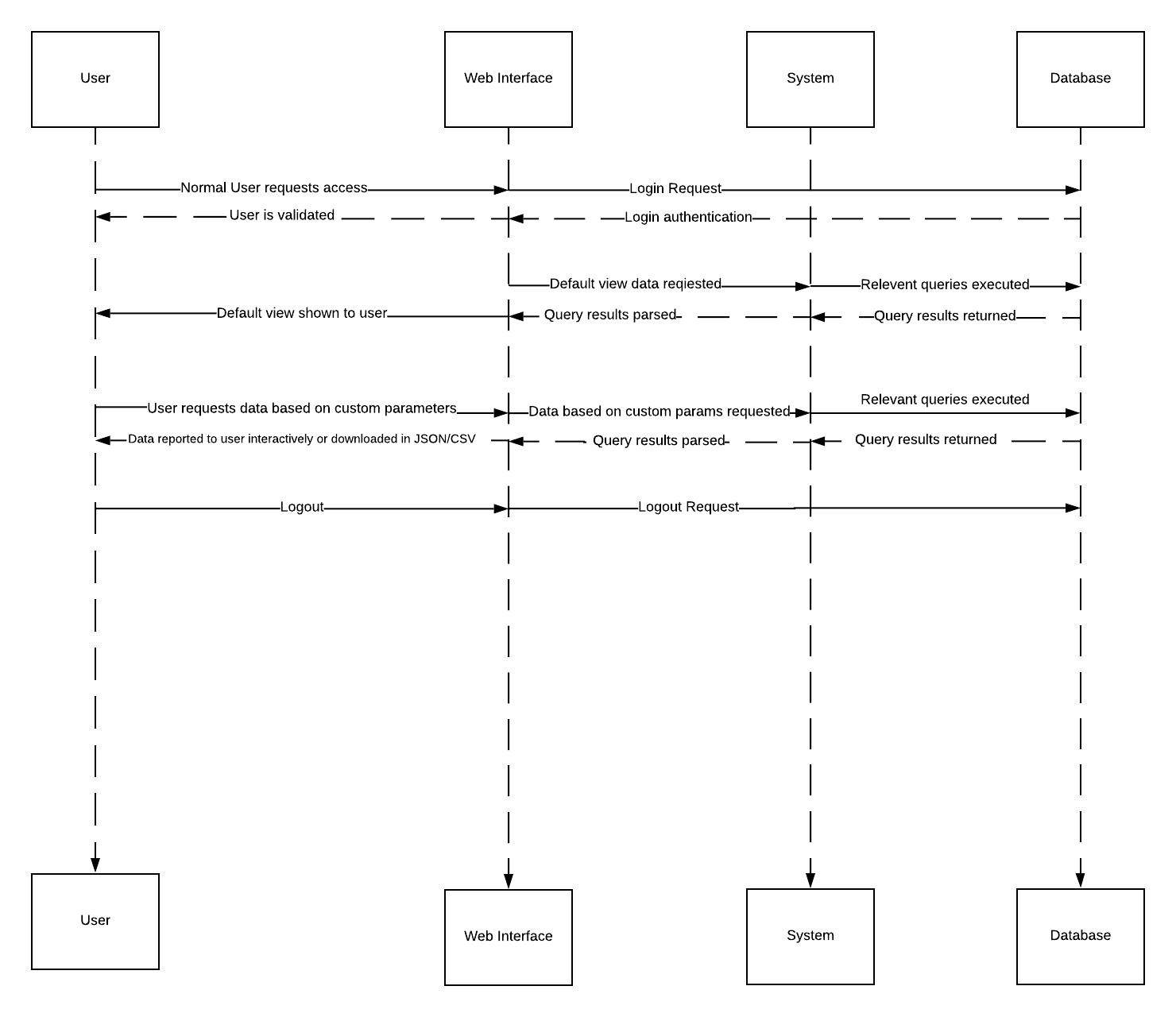
****

**Message Sequence Charts**

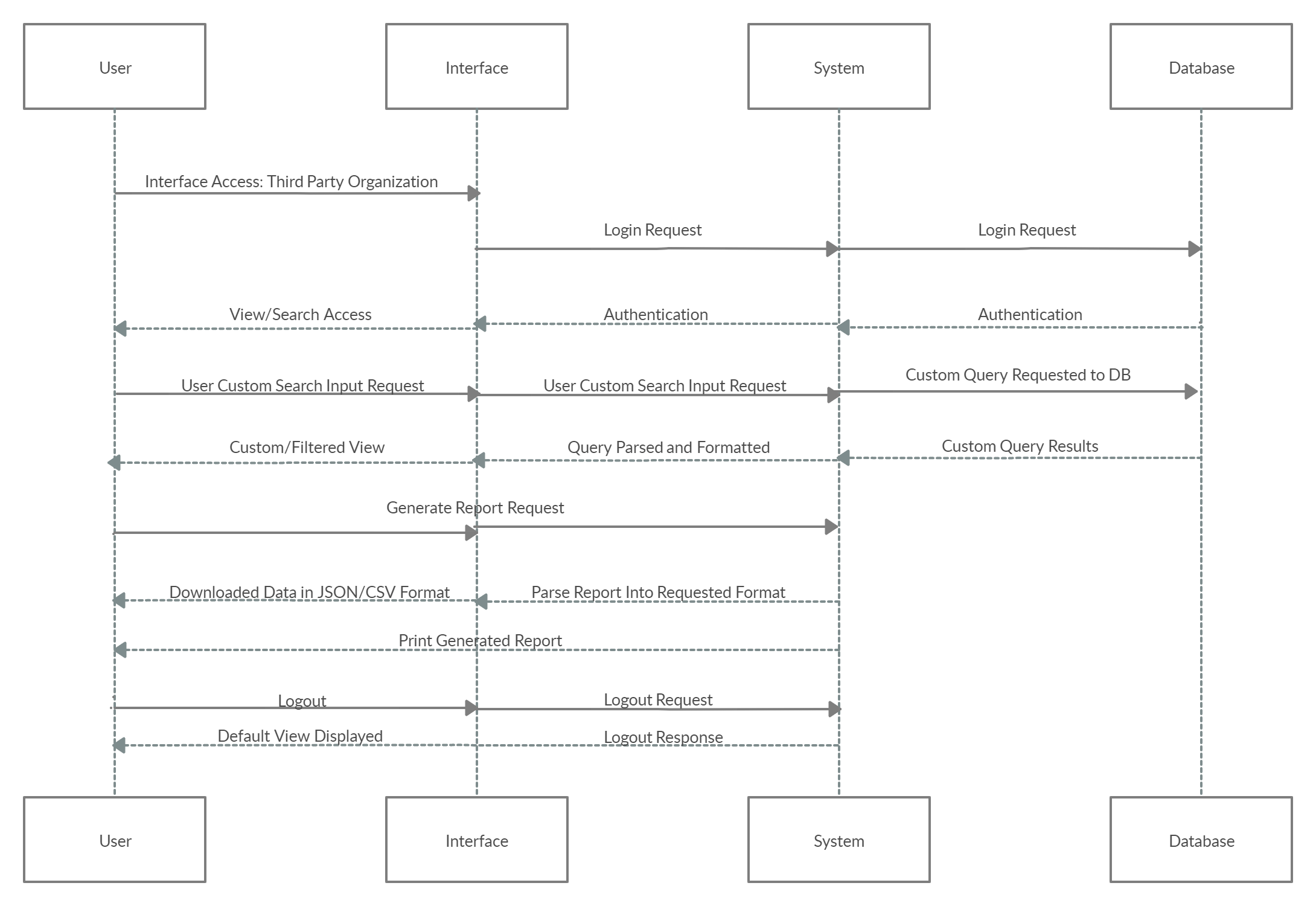
**Use Case 1**

****

**Use Case 2**

****

**Use Case 3**

****

**Customer Meeting Summary**

Our Customer (Michael Czapary) was able to meet with our group via Slack on Wednesday night. He was able to answer all of our initial questions about the project during the meeting and also notified us that he will be available when necessary to answer any further questions.

**Team Member Contributions**

The following bulleted-list summarizes team member contributions starting from customer-meeting up till HW1 deadline:

* All - continuous team meeting and communication via Slack throughout the week.
* Haelyn Brandt - Non Functional Requirements Definition, Non Functional Requirements Specification, UML class diagram
* Martin Edmunds - Functional Requirements Definition, Functional Requirements Specification, Use Case 1, Use Case 1 Message Sequence Chart, Home/Main page, Account Creation page, Login page
* Neil Gayeta - Use Case 3, Use Case 3 Message Sequence Chart, Data-Flow Diagram
* Brandon Goza - Use Case 2, Use Case 2 Message Sequence Chart