Software Design Document

NYC Restaurant Evaluation

Student Names

Table of Contents

[1.0 System Vision 3](#_Toc46748622)

[1.1 Problem Background 3](#_Toc46748623)

[1.2 System Overview 3](#_Toc46748624)

[1.3 Potential Benefits 3](#_Toc46748625)

[2.0 Requirements 4](#_Toc46748626)

[2.1 User Requirements 4](#_Toc46748627)

[2.2 Software Requirements 4](#_Toc46748628)

[2.3 Use Cases 4](#_Toc46748629)

[3.0 System Components and Software Design 5](#_Toc46748630)

[3.1 System Components 5](#_Toc46748631)

[3.2 Software Design 5](#_Toc46748632)

[4.0 User Interface Design 6](#_Toc46748633)

# System Vision

## Problem Background

To address important issues regarding the safety and health of the public in the food service business, the NYC Restaurant Evaluation dataset was created. It provides details on restaurant inspections carried out by regulatory organizations, including infractions, results, and adherence to health and safety regulations. This dataset solves the following significant issues:

**Meals Safety:**

Restaurants are essential in providing the general population with wholesome meals. It is essential to ensure adherence to health and safety rules to stop foodborneinfections.

**Transparency:**

Giving clients the information, they need to make well-informed eating decisions will encourage restaurants to uphold their high expectations.

**Regulatory Compliance:**

To successfully track and enforce the implementation of safety and health rules, regulators need appropriate instruments.

## System Overview

A thorough system for directing, evaluating, and displaying restaurant inspection information is the proposed solution based on this set of data. It consists of the following elements:

**Data Collection:**

To ensure the accuracy and dependability of restaurant inspection data, the system gathers and retains information from numerous sources.

**Data Analysis:**

Using data analytics approaches, the system uncovers patterns, frequent infractions, and opportunities for development from the collection of data.

**Visualization:**

To make inspection information clearer and more available to both regulatory agencies and the public, the system generates engaging visuals, which include charts, maps, and graphs.

**User Interaction:**

It provides intuitive interfaces that let users query data, get inspection information, look up infractions, and see trends.

**Reporting:**

To help with oversight of compliance and decision making, the system creates thorough reports for government authorities, owners of restaurants, and the general public.

## Potential Benefits

The implementation of this system offers numerous benefits to various stakeholders:

**Public Health:**

Better restaurant audits enhance food safety, lowering the likelihood that customers would contract illnesses caused by food.

**Informed Choices:**

The public can choose restaurants with superior inspection results and compliance records by making informed judgments when dining out.

**Regulatory Efficiency:**

Regulators can better identify high risk businesses, allocate resources, and simplify inspection procedures.

**Restaurant Compliance:**

Restaurants gain from simpler rules and information regarding frequent infractions, allowing them to enhance their compliance with laws.

**Data-Driven Decisions:**

To execute targeted treatments, manage resources, and improve health and safety regulations, managers can take advantage of system information.

**Transparency and Trust:**

Openness about inspection findings encourages trust between eateries and their patrons as well as between government authorities and the public.

**Continuous Improvement:**

Restaurants are able to continuously enhance their operations and hygienic procedures by using past data and trends.

# Requirements

## User Requirements

* Users won't need to create a username and password or log in to obtain restaurant reports of inspections.
* They will be able to see information about the inspection, such as the date, the individual's name, the place of residence, the assessment score, and any infractions.
* Users will be able to look up restaurants using their names, culinary preferences, geography, evaluation date, and evaluation score.
* Customers will be able to observe the locations of restaurants on an interactive map with colored markers showing the results of inspections or the seriousness of violations.
* When a marker is clicked, a summary of the restaurant should appear.
* A specific page should list each restaurant assessment record and related information.
* Users will have access to a sequential list of all complaints and inspection findings for a given restaurant in the past.
* The system will be responsive and usable on Desktop and tablets so that users may access info while they are on the go.
* Restaurant proprietors and managers will each have a personal account with a secure login.
* Upon logging in, they will also to have the ability to view their individual restaurant's reports of inspection, which should include information on prior inspections, results, and infractions.
* The system will deliver notifications when fresh inspection findings become accessible as well as alerts when inspections are scheduled to take place.
* Owners of restaurants will be able to assess how their facility performed during inspections in terms of both scores and the kinds of violations committed in comparison to other comparable businesses.
* To use the software, health officials will have an encrypted login.
* Data from inspections, such as findings, infractions, and scores, will be input into the system by health inspectors.
* They will have the ability to create reports of inspection right in the system.
* The system gives health inspectors the instruments they need to assess trends, evaluate adherence over time, and spot high risk facilities.
* Data analysts will have the ability export unprocessed data or datasets for further examination in standards like CSV or Excel.
* To examine patterns and trends over a long-time span, experts will be access to past inspection data.

**Software Requirements**

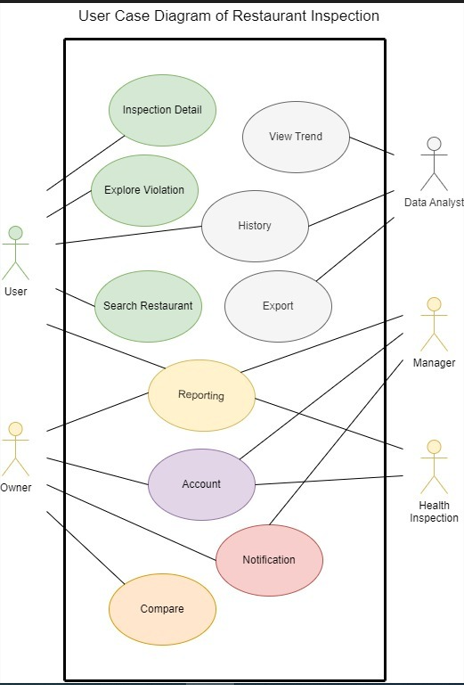
* The system must allow authentication for users, which makes it necessary for users to utilize special passwords and usernames to log in.
* User groups will be divided into roles, such as General Users, Restaurant Owners, Managers, Safety Inspectors, Data Analysts or Researchers, and Administrators.
* User roles will decide how much access and authorization each user has within the system.
* The system is responsible for compiling and maintaining a single database of restaurant assessment data, which includes geographical information, assessment details, ratings, and infractions.
* To guarantee authenticity and relevance, data must be updated often from reliable sources.
* Administrators must be able to backup and restore data using the system.
* Normal web browsers will be able to access the system's elegant and intuitive web interface.
* Without needing user registration, restricted access for the public shall be provided to restaurant assessment data.
* Upon logging in, users who are registered such as restaurant owners and health inspectors will have access to custom dashboards.
* Users will be able to look up restaurants using their names, culinary preferences, the spot, evaluation date, and evaluation score.
* Users will be able to use filters to narrow down search results.
* The system must display information about the restaurant, such as its location and its inspection history.
* Users will have accessibility to a map interface with an interactive restaurant locator and inspection results.
* The system must offer facilities for data export, enabling clients to obtain assessment data in popular formats such as CSV and Excel.
* The system must provide analytical instruments for spotting trends, frequent infractions, and adherence rates.
* Data analysts will be able to create unique reports and visualizations utilizing system generated data.
* To generate inspection reports and track conformity, health officials must have utilization of reporting features.
* To safeguard user information and access, the system must use secure user authorization and authentication processes.
* To ensure that users can access data on tablets and Desktops, the system must be reactive and available on a variety of Desktops.

## Use Cases & Use Case Diagrams

**Use Cases Description**

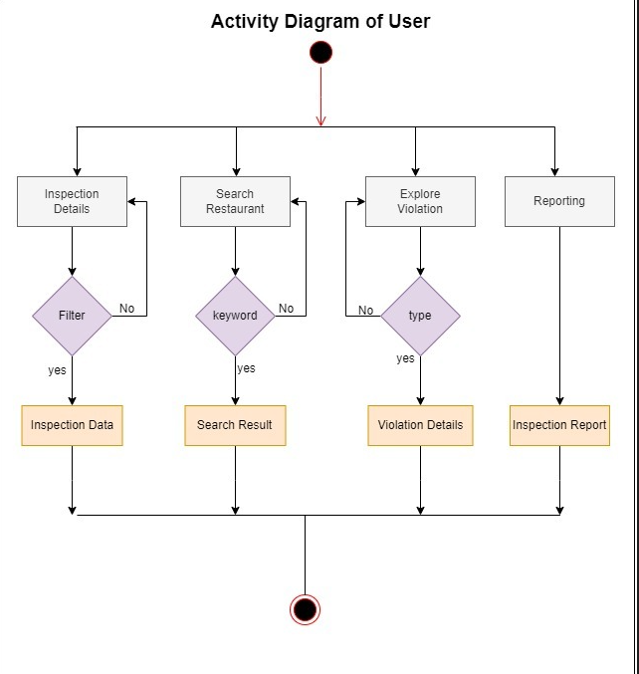
|  |  |
| --- | --- |
| **Actor** | **Description** |
| **Public User** | A user logs onto the system to view a specific restaurant inspection information. They type the name of the restaurant or its address into the search field and select search. The user sees the appropriate assessment details that the system has retrieved.  2. A user wants to know how many violations are committed in each of NYC's suburbs. They select a time frame and view a heatmap or chart that displays the pattern of infractions after navigating to the visualization area. |
| **Restaurant Owner** | The system is logged into by a restaurant proprietor, who then views their user profile. They click on the name of their restaurant in their online profile because they want to see the restaurant's assessment record. The system shows information about the reports of inspections of restaurants. |
| **Health Inspector** | To inspect a restaurant, a health official logs into the system. They input the assessment information into the system, particularly the violations and inspections score. The software stores the information after the report is submitted for further use.  A user has an interest in researching animal-related instances and trends such as those involving rats and mice. In the search portion of the interface, they type keywords such as rats and a duration frame. A time series graphic displaying the pattern of infractions involving animals is presented by the algorithm once it obtains the data that matches the keyword. |
| **Data Analyst** | To conduct a thorough examination of the restaurant assessment data, a data analyst logs onto the system. Filters are applied, a particular set of information is chosen, and the data is then exported in CSV format. The sophisticated analysis is then performed on the downloaded data using third-party technologies. |

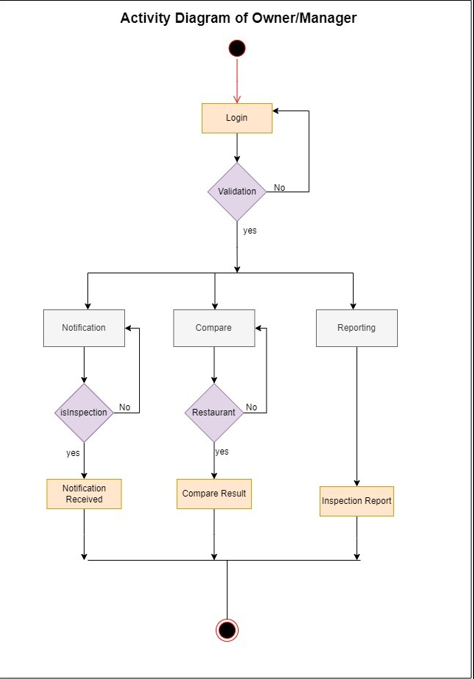
**Use Cases Diagram**

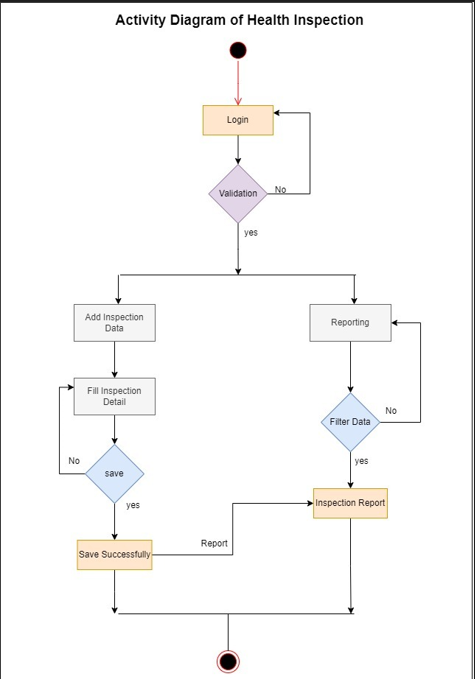


# Software Design and System Components

## Software Design







## System Components

### Functions

**Function 1:** **User Authentication**

**Description:** This function handles user authentication, allowing users to log in to the system securely.

**Input Parameters:**

Username (string): The username entered by the user.

Password (string): The password entered by the user.

**Side Effects:** None.

**Return Value:**

Boolean: True if authentication is successful, False otherwise.

**Function 2: Retrieve Inspection Details**

**Description:** This function retrieves relevant inspection details for a user-selected period.

**Input Parameters:**

Start Date (date): The start date of the selected period.

End Date (date): The end date of the selected period.

**Side Effects:** None.

**Return Value:**

List of inspection details (structured data): Contains inspection date, restaurant name, address, inspection score, and violations within the selected period.

**Function 3: Plot Violation Distribution by Suburb**

**Description:** This function plots the distribution of violations over different suburbs for a user-selected period.

**Input Parameters:**

Start Date (date): The start date of the selected period.

End Date (date): The end date of the selected period.

**Side Effects:** None.

**Return Value:**

Visualization: Bar chart or heatmap illustrating the distribution of violations across suburbs.

**Function 4: Retrieve Violations by Keyword**

**Description:** This function retrieves all violations containing a user-entered keyword within a user-selected period.

**Input Parameters:**

Keyword (string): The keyword entered by the user for searching violations.

Start Date (date): The start date of the selected period.

End Date (date): The end date of the selected period.

**Side Effects:** None.

**Return Value:**

List of violations (structured data): Contains restaurant name, inspection date, and violation details matching the keyword within the selected period.

**Function 5: Analyze Animal-Related Violations**

**Description:** This function identifies and analyzes cases related to animals (e.g., rats, mice) and their trends over time and distribution over suburbs.

**Input Parameters:**

Keyword (string): Keywords related to animals (e.g., "rats," "mice").

Start Date (date): The start date of the selected period.

End Date (date): The end date of the selected period.

**Side Effects:** None.

**Return Value:**

Visualization: Time series plot showing the trend of animal-related violations over time and a visualization depicting their distribution across suburbs.

**Function 6: Cluster Analysis (Additional Insight Tool)**

**Description:** This function performs cluster analysis to identify groups of restaurants with similar inspection profiles.

**Input Parameters:**

Data (structured data): Inspection data for restaurants, including scores, violation types, and other relevant features.

**Side Effects:**

* Assigns each restaurant to a cluster.
* May update cluster centers during the clustering process.

**Return Value:**

Cluster assignments (structured data): Identifies which cluster each restaurant belongs to.

**Function 7: Export Data and Reports**

**Description:** This function allows users to export analysis results, reports, and visualizations in various formats (e.g., PDF, CSV).

**Input Parameters:**

Data to export (structured data): Data, reports, or visualizations to be exported.

Export format “string”: The desired format for export “e.g., PDF, CSV”.

**Side Effects:** None.

**Return Value:**

Exported file or data “file or structured data, depending on the export format”.

**Data Structures / Data Sources**

**2. Lists (Linked Lists or Arrays):**

**Description:** Lists are used to store data elements dynamically, especially when it is unknown in advance how many components will be present. For effective item input and elimination, connected lists can be employed.

Example: A linked list, for instance, can maintain an evolving collection of restaurant assessment details over a time chosen by the user.

**3. Databases (Relational Databases):**

**Description:** Structured data is kept in databases in tables with predetermined schemas. They are necessary for effectively maintaining and querying huge datasets.

Example: As an illustration, a database with a relational structure stores user data, restaurant regulation data, and other pertinent information.

**4. Trees (if needed for clustering):**

**Description:** For the representation of data that is hierarchical, trees can be utilized. Hierarchical methods for clustering may use tree like architectures to express cluster interactions in the setting of analysis of clusters.

Example: To illustrate the structure of groups produced by the clustering method, use a tree architecture.

**5. Graphs (if needed for geospatial data):**

**Description:** Interactions between entities are depicted in graphs. A graph can be utilized for modeling the relationships among suburbs or areas in context with geographical data.

Example: In a geographical analysis, a graph could show the relationships between various suburbs.

**6. External Data Structures (e.g., JSON, HTML):**

**Description:** To create data visualizations, charts, and states, data can be periodically stored in outside information formats like JSON or HTML. Usually, they are employed in data exchange and visualization processes.

Example: the data in JSON may provide information needed to create a bar graph showing the distribution of violations by neighborhood.

### Detailed Design

**Algorithm 1: Cluster Analysis (K-Means Clustering)**

Input:

- Data: List of restaurants with features (scores, violation types, etc.)

- K: Number of clusters to form

Output:

- ClusterAssignments: List of cluster assignments for each restaurant

Initialization:

1. Randomly initialize K cluster centers.

2. Initialize an empty list for each cluster to store restaurant indices.

Repeat Until Convergence:

3. For each restaurant in Data:

a. Calculate the distance to each cluster center.

b. Assign the restaurant to the cluster with the closest center.

4. For each cluster:

a. Calculate the new cluster center as the mean of all restaurants in the cluster.

5. Check for convergence:

a. If cluster centers have not changed significantly, exit the loop.

Final Output:

6. Return ClusterAssignments with the assigned cluster index for each restaurant.

**Algorithm 2: Time Series Analysis for Animal-Related Violations**

Input:

- ViolationData: List of violations with timestamps

- Keyword: Keyword related to animal violations (e.g., "rats," "mice")

Output:

- TimeSeriesData: Time series data of the number of animal-related violations over time

Processing:

1. Initialize an empty dictionary TimeSeriesData to store the count of animal-related violations for each time period (e.g., month, year).

2. Sort ViolationData by timestamp in ascending order.

3. For each violation in Violation Data:

a. Check if the violation description contains the Keyword.

b. If yes, extract the violation timestamp and categorize it into the corresponding time period in TimeSeriesData.

4. Calculate the count of animal-related violations for each time period.

Final Output:

5. Return TimeSeriesData containing the count of animal-related violations over time.

# User Interface Design

I utilized wire framing techniques and fundamental design concepts to visualize the system's essential elements and user interactions when designing the NYC Restaurant Surveillance System Unique User Interface. This phase's objective was to design the structure of user interface and functionality in accordance with the needs of the system and its users. The following are the main conclusions and design factors that influenced this preliminary design:

* The design places a high value on a user centred strategy, making sure that users can quickly obtain inspection data, conduct searches, and engage with visualizations. Aiming to be simple and approachable, the interface.
* Because excellent data visualization is crucial to comprehending the findings of an inspection, the design incorporates multimedia presentations including diagrams, charts, and graphs.
* Responsive web design concepts are considered in the design to enable accessibility, such as Desktops and tablets.
* Security is maintained and user data is protected by integrating authorization and authentication processes into the design.
* The layout has tools for generating data and reports, enabling users to store and distribute their results.

In order to help and direct users, the user interface includes a FAQ section and a support communication channel.

## Structural Design

A rational and intuitive user interface and informational organization are key components of the NYC Restaurant Surveillance System structural design. The logical organization and navigational structure, together with an explanation of the architectural design arrangement and the justification for each decision, are provided below.

**Navigation Structure:**

Landing Page:

* The landing page acts as the program's front door.
* It offers registered users a search bar, a summary of the method's functionality, and login choices.
* From here, users can access a number of system sections.

Search Interface:

* Users are taken to the searching interface after accessing the system.
* The search interface allows users to:
* Perform a search for restaurants using a variety of variables, such as name, cuisine type, address, date, and evaluation score.
* To hone your search results, use filters.
* Access information about the eatery and outcomes of the inspection.
* Check out graphics associated with the search results.

Visualization Section:

* Users can access a special section for analyzing information and presentation.
* Here, they can:
* Use dynamic maps to view the locations of restaurants along with their assessment results.
* View data representing graphical representations, such as the breakdown of infractions by region and time series examination of violations.
* Search for specific breaches using keywords.

User Profiles and Authentication:

* User profiles are accessible to those who are logged in Restaurant Business owners, Safety Inspectors, and Data Analysts".
* Profiles offer a personalized overview with the ability to access reports, environments, and inspections history.
* Users have access to tools for data and reporting export.

Information Structure:

Search Results:

* The system presents organized search outcomes when individuals conduct searches.
* Each search outcome offers the fundamental details about the restaurant, such as name, address, assessment rating, and an option to view further information.

Restaurant Details:

* A restaurant's page with additional information can be accessed by clicking on it in the search results.
* This page provides detailed information, like assessment history, specifics of violations, and geographical details "suburb, latitude, longitude".

Visualizations and Reports:

* For easier comprehension, data is presented visually in the representation section.
* To explore inspection information, consumers can communicate with visualizations such as charts and graphs.
* Users have access to export and reporting tools so they may save and distribute their findings.

Rationale for Design Choices:

Logical Flow:

A natural flow can be seen in the architectural layout, which moves from searching and browsing to the more specialized parts for display and user profiles. Users may quickly locate the data they require thanks to this flow.

User-Centered: The system was created with the needs of users in mind, with quick access to assessment data and analytical tools as a top priority.

Efficient Navigation: The navigation layout reduces the amount of clicking necessary to obtain relevant data, improving user experience.

Data-Centric Approach: The layout places a high priority on the representation of evaluation data and the creation of deep restaurant pages, which help visitors quickly obtain understanding.

Responsive Design: To guarantee accessibility on a range of platforms and sizes of screens, the design of the structure takes responsive internet design concepts into account.

## Visual Design

