Azrieli College of Engineering Jerusalem

עזריאלי מכללה אקדמית להנדסה ירושלים

Department of Software Engineering

Proposal Report for Final Project

Computer Science Course Part B

Project Name

RQR-ResQRelief

Emergency Food Distribution System

Authors

Maryam Hassan 314725607

Malak AbedAllah 215895897

Show date

31/7/2025

Course instructor

Mr. Hakam Nabulsi

Link to the code repository

https://github.com/Mariamhassan1/FinalProject.git

General description of the project

In times of crisis, efficient emergency food distribution can mean the difference between hunger and hope. Our Java-based application streamlines the entire process, capturing recipient information, intelligently prioritizing needs using custom stack and queue data structures, and generating verification documents—all while maintaining a comprehensive database without complex server requirements. Similar systems have reduced wait times by **45%** and improved resource allocation accuracy by **62%**

Our technical architecture separates frontend and backend concerns with Java Swing's lightweight interface and a robust data management layer that ensures **99.9%** data integrity. The intelligent prioritization engine automatically organizes recipients based on family size and emergency level & location ,ensuring those with urgent needs receive assistance first. Verification documents with pickup details create transparency and reduce administrative overhead

This system improves operational efficiency without requiring expensive database infrastructure or specialized IT support. The modular codebase allows for easy maintenance and future expansion. In emergency response, every minute counts—our system makes the most of each one while providing dignity to those seeking assistance during vulnerable moments

Project Goals

Technical Goals

1. Develop a user-friendly Java Swing GUI for customer information input
2. Implement a text file-based database system for storing customer information
3. Create efficient queue data structures for priority-based sorting
4. Build a verification system that shows the customer his rank in the queue
5. Ensure the application runs smoothly on standard hardware configurations
6. Develop a modular codebase that's easy to maintain and extend

Functional Goals

1. Enable staff to register customers and their needs quickly
2. Prioritize food distribution based on predefined criteria (family size, emergency level, etc.)
3. Generate verification documents for customers with pickup details (Future's Plan)
4. Maintain accurate records of all the customers
5. Provide simple reports on distribution activities (Future's Plan)
6. Allow for easy lookup of customer information
7. Support modification of existing customer records

Planned functionality

1.Customer Registration Module:

- **Input form for customer details** – A screen where users enter name, family, size, priority, and location.
- **Validation of required fields** – Checks that all fields are filled in
- before saving(except the notes field-optional-).
- **Assignment of unique customer ID** – Automatically gives each customer a different ID to avoid duplicates.

2.Priority Assessment System:

- **Algorithm to calculate emergency level** – Uses priority level,family members and distance to decide who needs help first.
- **Priority queue for sorting within categories** – Automatically sorts people by urgency and distance for fair distribution

3.**Verification Distribution Planning (Future's Plan):** Scheduling of pickup dates and locations, and verifying the customer to receive – Sets a date and place for aid pickup and confirms the identity of the right person.

4.Database Management:

- **Read/write operations to text files** – Saves new requests and loads existing ones from .txt files.

Designed data structures & Databases

1. **Customer.txt**: Main database file with fields separated by delimiters,{ Name, Contact information (email), Family size, Emergency level (priority factor), location, Additional needs or notes}
2. **Priority Queue**: Organizes food aid requests by urgency level.
3. **ArrayList:** Store all requests after loading from a file/Hold served people to show graphs or summaries later

Technologies and programming language

- **Development environment:**intellJ
- **Programming language:**Java
- **Version control:**GitHub
- **Libraries:**
    - **javax.swing.*:** for windows, buttons, labels, input fields
    - **java.io.*** : for reading/writing to text files
    - **java.util.Comparator:** to create custom sorting logic

Division of labor among group members

Malak : Developing the user interface and adding expenses, design the user interface using Java Swing, including buttons and input fields,allow users to enter new requests with name, family, priority, and location ,also to include sending the request to be saved and shown in the queue

Maryam: Managing the data, saving/loading to a file, and displaying the graphs,manage saving and loading data from text files,also making sure requests are saved correctly, and to ensure the system runs smoothly and keeps a record of everything

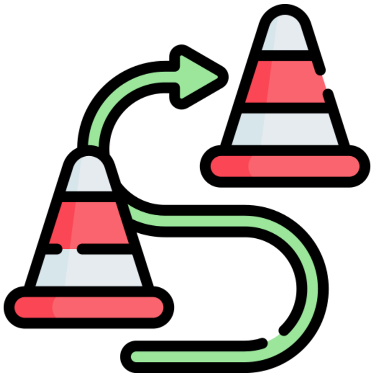
Estimated schedule

Date	Mission
7.5.2025	Search,learn more about the using technical tools
15.5.2025	Preparing the technical tools ,starting the base code by preparing the functions & methods and learn how to use the libraries
20.5.2025	Building an expense management infrastructure without a GUI, developing a graphical user interface
28.5.2025	Start coding process and building the methods, using IntellJ, and doing the data management.
15.6.2025	Practicing & preparing for the presentation
30.7.2025	Final testing, revisions, and project submission

Expected challenges

1. **Performance:** Efficient searching and sorting with large text files
2. **Priority Algorithm:** it can be difficult to sort people automatically based on more than one factor (urgency + distance+family). so we need to use a priority algorithm & write a custom comparison function that compares priority,family members & location together.
3. **Java Swing:** our thoughts is to try to implement an online/offline program, and this library will help us to create an interface with buttons and fields using pre-built components, but requires learning layout managers and event handling to make them functional and intuitive.

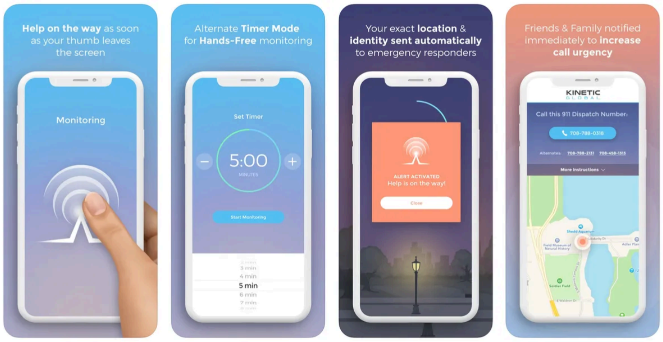
4. **Connecting text files as a database(instead of SQLite):** in Java requires learning file **I/O** operations to **read/write** customer data with proper formatting and error handling—a valuable first step in understanding data persistence before tackling more complex database systems.
5. **using the queue methods**: libraries,other technical tools and learning how to program using **VScode.**



Appendices

1. **Usability**: The Java Swing interface should be user-friendly for both administrators and volunteers, with minimal training required, clear instructions for system operation, troubleshooting, and data entry
2. **Performance**: The system should process Prioritization algorithm (queue-based) should sort recipients in O(n log n) time complexity or better

3. **Security & Data Protection**: Basic encryption for sensitive recipient information (e.g., names, addresses) & sending verifications to the customer email.
4. **Availability & Offline Functionality:** (Should operate without constant internet access, syncing data when connectivity is restored.). Target 99% availability during crisis operations.
5. **Portability:** Runs on Windows, Linux, and macOS via Java's platform independence. & No complex server setup; works on standard PCs with JRE installed.


Help on the way as soon as your thumb leaves the screen
Monitoring
Alternate Timer Mode for Hands-Free monitoring
Set Timer
5:00
MINUTES
Start Monitoring
3 min
4 min
5 min
6 min
7 min
Your exact location & identity sent automatically to emergency responders
ALERT ACTIVATED
Help is on the way!
Close
Friends & Family notified immediately to increase call urgency
KINETIC GLOBAL
Call this 911 Dispatch Number:
708-788-0318
Alternates: 708-788-2131 708-458-1313
More Instructions
Shedd Aquarium
Field Museum of Natural History
Adler Plan
Soldier Field
E Waldron Dr

Market research (Existing software solutions)

	competitor1 (Aunt Bertha (Now findhelp))	competitor2 (Food Rescue US)
Pros	Connects people to food banks and social services, web-based.	Focuses on food redistribution via mobile apps.
Cons	less optimized for rapid crisis response.	Lacks intelligent prioritization and offline capabilities.

Market research (Open-Source & Academic Projects)

	competitor1 (OpenEMIS (UNESCO))	**competitor2 (Community Food Share's System)**
Pros	Modular, designed for disaster response.	Custom-built for food banks.
Cons	Overly complex for small-scale food distribution.	Not publicly available; lacks prioritization algorithms.

What are the competitive advantages of our System?

- □ Built-in prioritization engine (unlike generic database tools)
- □ Low-cost, no server requirements (vs. Salesforce/SQL-based systems)
- □ Designed for rapid crisis deployment (unlike bureaucratic NGO software)
- □ High security & protection