Project Scope

Note: This document is intentionally incomplete and is in non-standard format as part of the course this document will be re-written.

Project Scope: (Double Iterative Hybrid Agile process)

Saving a life is a noble cause. But natural disaster events like COVID-19, floods, earthquakes, tornados, and hurricanes occur too often grounding local disaster management. There is a need for a global Virtual Disaster Assistance capability. In this model, **Assistance is delivered virtually** but directly from **Donors** to the **Recipient.** **Zero Inventory capability** is foundational. For example, a Donor wishing to supply blankets for a particular request should be able to ship it directly from retail stores to the Recipient. **Optimized use of available resources** is a critical success factor. It is much easier for Iowa City to accept help from North Liberty than from Southern Florida. Admins create events and support donation items. Recipients request items and Donors respond to such requests. A proactive pledge of donation may also be made. One or more of the responses may fulfill a request. The business goal is to match request-response automatically and optimally to deliver automated relief supplies with zero storage capacity, with total confidentiality and friendly notifications.

Request for Help (Recipients)

People who are exposed to nature's fury need help. Help is required in the form of human rescuers, money, and materials. Various categories of items like food, clothing shelter, medicines, medical attention, and common day-to-day consumables are required. The need for volunteers who can be on the field is very critical. Transport on land, water, and air is required. Such events needing help may occur simultaneously at multiple locations. **Optimized use of available resources is a critical success factor.** Anyone can **request help.** A request for Help may have one or more components (Human resources, money, material) in it. The requestor or the relief center person can **expire** individual request items or the entire request. If a request has been fulfilled it is so marked. Both fulfilled and expired items **can't** be delivered. There are many categories of items and categories of help that may be required, and it does become a challenge to collect, process, deliver, and manage such requests.

Response to Help (Donors)

Fortunately, people are generous and keen to help **but they may not know what is required and where such help needs to be delivered.** Both coordination and automation are required for the success of the relief and rescue efforts. Two types of Responses are

- **Respond to Requests:** Anyone can search the request database and respond to it. The response may be towards one or more items in a request. A response will be towards a request.
- **Pledge:** A response to help may be proactively made (even before a disaster has occurred or a request is created) but delivered against a real event. **This is known as a pledge.** We certainly need a virtual agent or admin human that connects Donors, Recipients, volunteers, and government agencies.

Project Scope

Match Requests with Responses (There are two types)

- **Auto Match:** Our goal is to have the request-response matched automatically as much as possible using match criteria, to have automated relief management. Admin users should be able to override this match using a manual match.
- **Manual Match:** Request-response matched manually using the proximity using a match criterion.
- **Match Criterion:** Accept match criteria

Admin users should be able to query and match the request with a response and notify the donor/recipient for direct delivery. There is **no** storage capacity with this system.

Exclusions: The actual mechanism of fulfillment (shipment and delivery) is external to this tool. Once the shipment is made the donor will update the response/pledge with shipping tracking info.

The data model should be designed in such a way that users can search for the requirements at the relief center near their location. It is considered most efficient if people near the location of disaster provide help. The proposed web portal must be highly extensible and support concurrency. The goal of this phase of the project is to create a robust data model and design the database using the principles of our course. And to develop a web application interface with as many functions as possible.

Disaster Assistance Management System (DAMS)

Functional requirement sketch Call center users can connect those who need help, those who wish to help, volunteers, and government agencies. Call center operator and admin users.

a. Disaster events with geographic location details, contact center details, duration of occurrence of event, and the categories of required items or help can be created by admins.
b. Once an event is created, individual items required at each location can be entered by any user from the event center or administrator
c. The categories of items will differ from event to event.
d. Assume common attributes of each item to be listed.
e. Confidentiality of volunteers must be maintained.
f. Context-sensitive help is required, and the help text should be changeable without changing software
g. Email Notifications
h. Roles: Donor, Admin, Recipient
i. Search - From-To (open to public- Unauthenticated)

An urgent requirement is to build a state-of-the-art **web** application for such a tool. An iterative approach is probably ideal. Recommend using **Double iterative Hybrid Agile process.**

Project Scope

While we have a good idea of the requirements, we would like help from your group to draft the requirements and develop this application within the next 3 months. Anticipate lots of changes. The broad scope is

1) **Donor functions: Respond, Pledge to respond, etc.**
2) **Disaster Event Center representative (Recipient) functions include registering for help, receiving help, expiring help, etc.**
3) **Administrator functions must be provided.**

The key is to come up with a good architecture that can support both Mobile and Web browser environments. Code needs to be layered and completely reused between the two. The initial version can support just the web browser environment.

Such software is expected to attract worldwide users. It must be simple software, easy to use, maintain, and offer very high performance. While we have a good idea of the requirements, we would like help from your group to draft specific requirements and develop this application within the next 3 months. Customer contacts will provide additional details of requirements if requested by the Project team. Any of the users should be able to search only the data they are authorized to see. A donor can see only their responses. Only Admin can see all.

The requirement is to build a state-of-the-art application for the above-integrated software application. The first step is to identify the module functions you wish to develop and gather more detailed requirements catering to excellent performance in a secure environment.

The data model should be designed in such a way that extensions are easy to make. The proposed tool must be highly extensible including the use of a different architecture. Such software is expected to attract worldwide users. It must be simple software, easy to use, maintain, and offer very high performance. Customer contacts will provide additional details of requirements if requested by the Project team.

Customer contacts: Raman, TA

General Requirements:

1. Must use the principles of Software Engineering.
2. Good use of tools is recommended. This environment setup should be the zeroth iteration of the project. The tools may be used as a service or installed in your local. Note ITS and CSG offer many options. Here is our 5-tool recommendation.
    a. Source library control (GitHub, GitLab, etc.)
    b. Automated tests (jUnit, Cucumber, Selenium, etc.)
    c. Automated Continual code inspection (sonar is provided by ITS) https://www.sonarqube.org/
    d. Continual integration (CI) (Cruise control, Teamcity, Gitlab) http://cruisecontrol.sourceforge.net/
    e. Defect tracking (Bugzilla etc.) https://www.bugzilla.org/

2. The tool should be user-friendly and require minimal data entry. The Software development team and the Business analysts are required to suggest specific user-friendly features while drafting the User Requirement Specifications. Some examples of user-friendly features may be
    a. minimum data entry
    b. data transfer from context-sensitive help screens
3. The system must support user ID / Password-based authentication and must offer the best security features. The software development team is required to identify (in URS) the specific security features to be incorporated in VSS. Some examples of such security features may be
    a. Data encryption techniques (128-bit)
    b. Invisible password
    c. Structure of Password
    d. Password change policy
    e. User ID / Password retrieval policy
4. The System must provide four categories of users with role-based access control (RBAC). Should be able to add more.

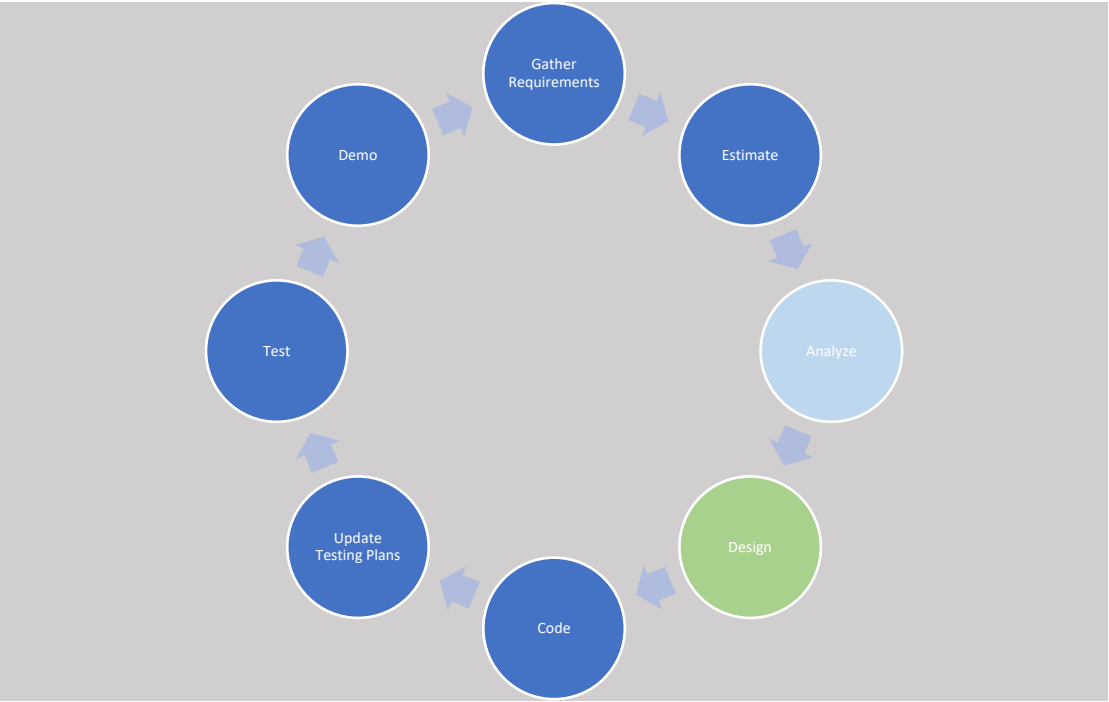
Non-functional, Operational requirements from the project team:

- MySQL or Oracle or SQL Server database only.
- **Web Application**
- **Code:** Java/.Net/Python/Node/Angular/React/JavaScript
- The use of good architecture is a must.
- Teamwork is very important
- Team members must have a minimum of 1 e-meeting a week detail of which must be recorded. e-meetings are preferred, and in-person meetings (except after class) should be rare or minimal.
- Meetings should be well planned and should preferably be less than 30 minutes in each meeting.
- It is highly recommended that teams meet once right after class.
- Distributed teams are a matter of fact; technology should be used to meet remotely if it is convenient.
- The team must submit a more detailed schedule and get it approved.
- The team may choose programming tools but must get them approved.
- Automated testing is mandatory.
- All members of the team must assume specific responsibilities for each activity of software development
- The customer must be informed of the Roles and Responsibilities
- The tools required for the project will NOT be provided by the customer
- Due to an acute shortage of resources, the customer can meet with project teams only on a pre-arranged basis.
- All deliverables must be delivered using an ICON drop box.
- **Get your tools approved.**

Project Scope

Sprint Process - Define-Analyze-Design-Develop-Test-Implement

#	Deliverable Description	Due on or before
Plan	Documentation, planning, professionalism 5%	For each iteration
Define	User Requirement Spec (URS) – Use cases 10%	For each iteration
Test	Test Plans, Automated Testing 10%	For each iteration
Analyze	Software Requirements Specification (SRS)10%	For each iteration
Design	High Level Design (screen, database, arch) 10%	For each iteration
Develop, Test, Implement	Setup Environment, Training 10%	**Jan End**
	Sprint 1 (4-week sprint) 10%	**Feb End**
	Sprint 2 (4-week sprint) 10%	**March end**
	Sprint 3 (4-week sprint) 15%	**April end**
Present	Project Completion 5%	Semester end
Reports	Weekly Progress Reports 5%	Every Sunday



Possible Use-cases: (Backlog)

Usecase	Roles	Sprint	Process
Logon	All	**S1**	**Gather Requirements, Estimate, Code, Test, Demo**
Donor/Recipient Account Creation	Admin/Actor	**S1**	
Public access – Home page	All	**S1**	
Donation Item Maintenance	Admin	**S1**	
Create Disaster Event	Admin	**S2**	**Gather Requirements, Estimate, Analyze, Code, Test, Demo**
Create Request for Donation	Recipient	**S2**	
Make a Pledge 2 donate items	Donor	**S2**	
Create Response to a Request	Donor	**S2**	
Item Shipped Status Update	Donor	**S3**	**Gather Requirements, Estimate, Analyze, Design, Code, Test, Demo**
Donor – Disaster association	Donor	**S3**	
Match Request-Response- Manual	Admin	**S3**	
Match Request-Response- Auto	Admin	**S3**	
Add Modify Donation items	Recipient	**S3**	
Password Change, Retrieval	All	**S3**	
Notification	Admin	**S3**	

Revision Log

Date	Version	Description of change
1/28/2023	1.0	Initial version

Distribution List {Students}