Software Development Group Project

(2nd Year)

PROJECT PROPOSAL

Disaster Management System for Sri Lanka

Group Name:

Group Members:

Full Name of the Students	Student Registration Numbers
Keshari Hasithangi	UOG/BSC/IS/2021/1/00007
Vidushan Prabash	HD/2020/NOV/0149

Higher Diploma in Software Engineering / BSc. in Computing
Department of Computing
LNBTI

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
UI	User Interface
GUI	Graphical User Interface
HTML	Hypertext Markup Language
CSS	Cascading Style sheet

List of Appendices

Appendix	Description	Page
Appendix - A	Plan of Action	11
Appendix - B	Gantt Chart	12

1. Introduction

A disaster is a serious problem that occurs over a short period or long period which will cause widespread human, resources, economic and even environmental loss that may be caused by natural or man-made even in Sri Lanka. The damage caused by those disasters may depend on climate, geographical location, human negligence, and errors occurring in a man-made system. Even though there are systems, still a disaster is crucial.

With the new advances in computer technology, information management has become easier than ever. As a result, people have started to adopt information technologies. Accordingly, people used to develop systems to provide a service. The existing methods of disaster management software focus on detecting disasters beforehand to give early warnings to the people in order to reduce the risk by empowering the local authorities.

As the existing system is deployed for the government, the general public in Sri Lanka is not much familiar with a system which helps them to reduce the risk of death on their own. Therefore the "Disaster management system for Sri Lanka" gives pop-up messages about safe and unsafe locations, will find safe and nearest places and will provide a service for donations. Any individual or company who is willing can provide donations to the people who are impacted by the disaster.

Also, this application will show information about affected parties and their need which will also be useful for the donators.

1.1 Background & Motivation

A disaster which is natural or man-made is a serious problem which will affect people by causing loss of human lives and property all over the world. So in the case of a disaster, it is important to have an organized and effective response effort to take preventive measures in a tragedy. When considering peoples' perspective, in an unexpected situation they are willing to get notified beforehand as well as they prefer to get help immediately when in a disaster.

when we imagine a disastrous situation people are more likely to feel hopelessness, empathy and despair. When we came across we felt to develop a system which will help people in such a hopeless moment.

Not only that the inspiration for this system was also taken from an existing system named "Sahan Eden" which is a disaster management software made for organizations [1]. So a similar software will be made in this software which is for the general public which is portable.

1.2 Problem in Brief

The main cause for the deployment of this system is that people can not be able to know relevant location is safe or not, even after a disaster they might not know a safe and nearest temporary place to shelter themselves easily by themselves using an app. So donators' point of view, might not know about the victims, and their needs.

This application will support by checking and will show if any location is safe or not on the map and will help people with donations, and safe places and will provide instructions which include a simple architecture which is more understandable.

1.3 Aim and Objectives

1.3.1 Aim

This project aims to develop a system which helps the general public by alerting them about disasters and providing services for their needs when in a disaster.
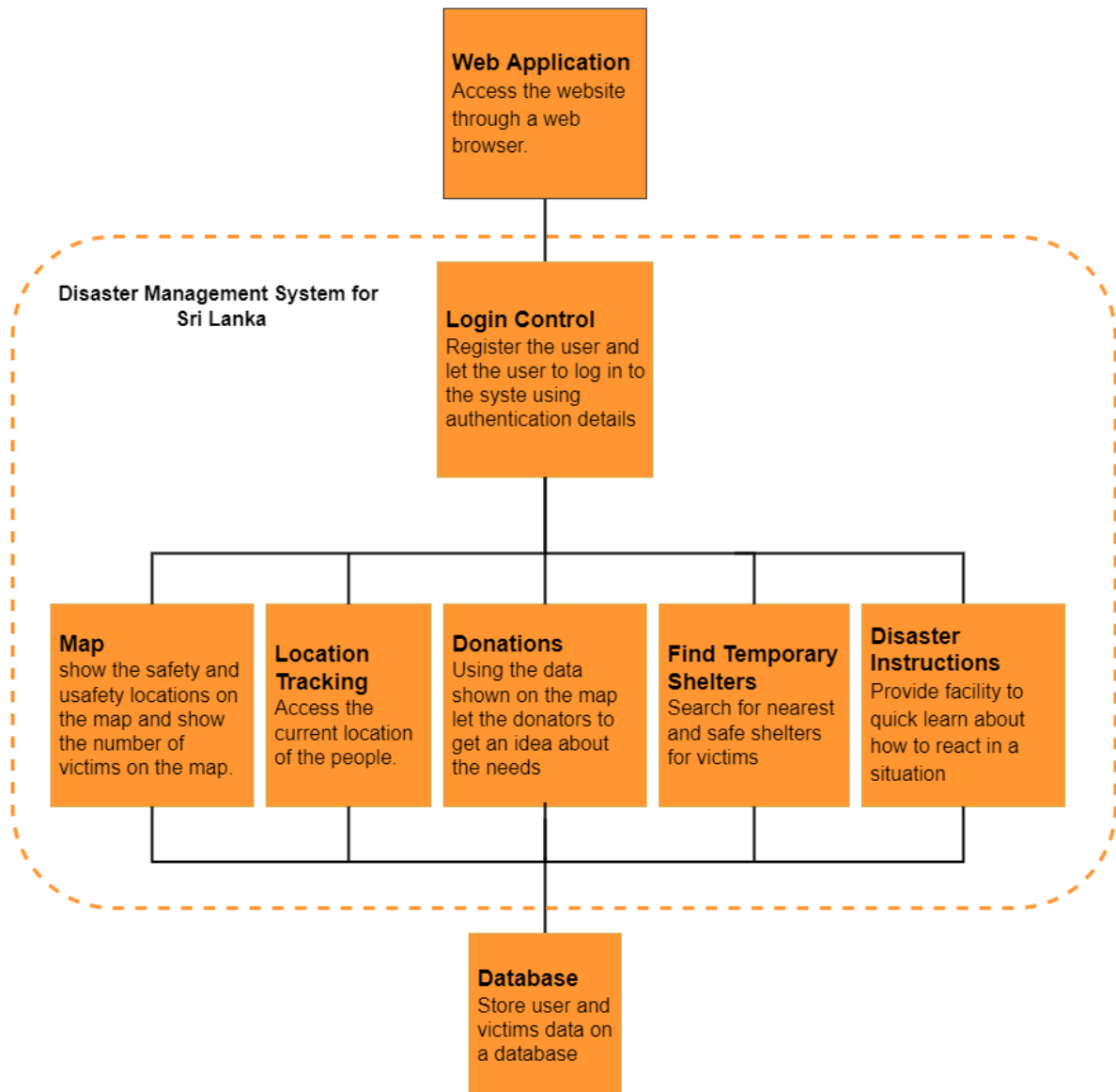
1.3.2 Objective

1. Identify the hazards
2. Search temporary shelters for victims
3. Analyse the details of the victims
4. Provide details of the victims for donors
5. Giving warning before disaster thus people can be prepared and safe from that.
6. Design and develop a clear user interface for donation

1.4 Summary

Preventing people from a disastrous moment is not a very easy task to do. If anyone can view whether a location is safe or unsafe using the application easily, it will be able to minimize the impact that occurs due to a disaster. Even they will be able to get an alert when in a disaster and also how they should act in a disastrous situation will be also given to them as instructions. Considering the distance safe and nearest temporary shelters will be located on the map to provide service to the victims. Via a user-friendly User interface (UI), donators will be able to get a clear idea about the victims and their needs. In order to achieve this, the following functionalities and technologies will be used in the system.

2. Proposed system

2.1 Abstract Architecture of the System

Web Application
Access the website through a web browser.

Disaster Management System for Sri Lanka

Login Control
Register the user and let the user to log in to the syste using authentication details

Map
show the safety and usafety locations on the map and show the number of victims on the map.

Location Tracking
Access the current location of the people.

Donations
Using the data shown on the map let the donators to get an idea about the needs

Find Temporary Shelters
Search for nearest and safe shelters for victims

Disaster Instructions
Provide facility to quick learn about how to react in a situation

Database
Store user and victims data on a database

2.1.1 Functional Requirements

1. Users should be able to check whether there is a disaster in a relevant place using maps.

    Using the map, users can point to any location and will give a pop-up message about whether the relevant location is safe or not.

2. Users be able to find safety and nearest places using GPS

    The system will show the nearest and safe locations by considering the distance.

3. the system should support and provide a better interface for the donators.

    The system will indicate the information about victims like the number of victims, their current location

4. The user should be able to log in to the system.

    The user should be able to register into the system by giving authentication details and logging in to the system to use the application.

5. The system should send an alert to the users (general public, self-defence teams) before and after the disaster

    The system will send an alert to the general public and self-defence teams using the application and SMS in order to give a trigger before a disaster and after a disaster, self-defence teams can provide service to the people.

6. The system should get contact details of the family, friends and others to notify them by sending the locations

    When registering to the system, the system shall ask for the emergency contact details of anyone that will send the locations of the victims.

7. Users should be able to check the safety of their family or friends

8. Users should be able to get directions on what they should do before, during and after the disaster.

    The system will provide instructions for the general public and pieces of advice which they should follow before, during and after a disaster.

9. Analyze deaths and property damages that occurred during the disaster

2.1.2 Non-Functional Requirements

1. Security of the user information

   The user will provide the user name and password and will allow location tracking when logging in to the system. So the system should provide security to the details from unauthorized access as the user with valid user credentials will only be able to log in to the system.

2. Validity of the contact details and emergency numbers

   There should be proper contact details of the emergency services as well as victims' relations to get them notified in a disastrous moment.

3. User Interfaces (UI) with user-friendliness

   Users of different ages can be able to use it as it is user-friendly for any age group.

4. Better Performance

   The system downtime should be less.

5. Up to date

   The system should provide updated information about the disasters, victims, etc.

6. Maintainability

   The issues that arise with the system will be repaired and restored within a short time.

2.2 Software Development Methods

The main software development method to be followed in this project will be "Iterative and Incremental methodology". This application will work by segmenting the product into individual sections which are referred to as an "Iteration" and building them separately [2]. After finishing an individual section, combine it with the previously designed one in order to make a whole. As this methodology works as a progression, it is adapted to changes that occur during the processes like adding new features or adding unexpected technologies. By this model we can,

7. reduce the risk as can test the integrated increment at least once per sprint.
8. Changes are acceptable even when in an unexpected situation.
9. Deliver working software.
10. Provide important or prioritized functionalities easily [3].
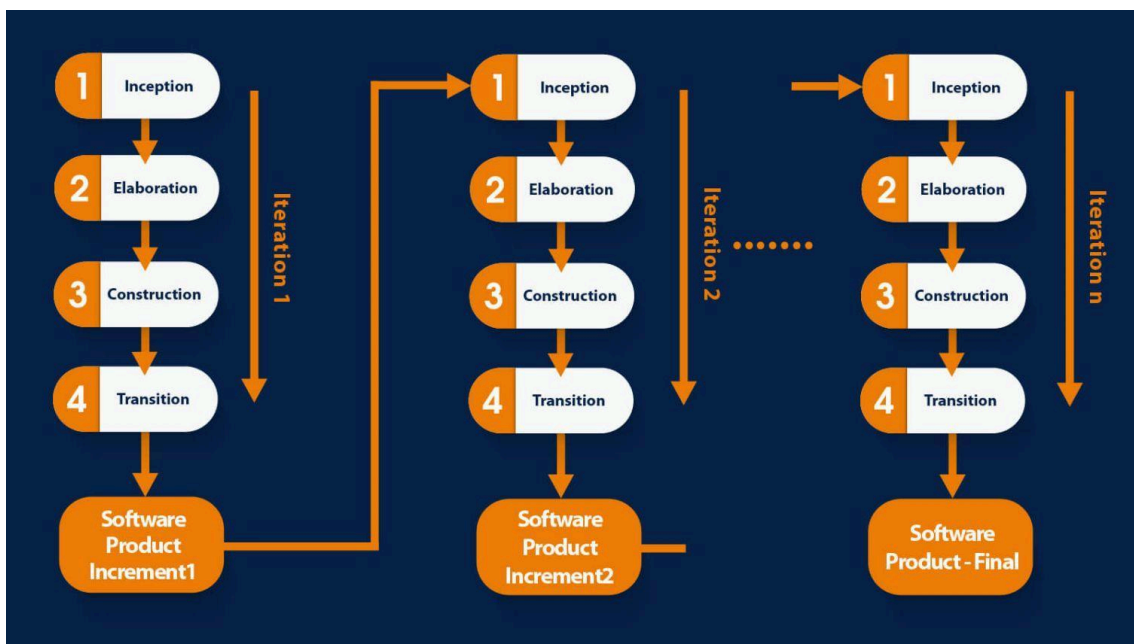
2.3 Technologies to be Adopted

Table 2.3.1 Technologies to be Adopted

Hardware	Desktop - allows users to do operations using the Graphical user interfaces (GUI).
	Mobile - allows the user to easily use the web application and the interfaces.

Languages	**Python** – use to develop the back end of a web application.
	Java Script - use to make web pages more dynamic.
	HTML – use to structure the web page and its content.
	CSS – use to create different layouts on web pages.
Platforms	**Django** – is a framework used to develop the backend of a web application.
Software	**SQL Server Management Studio 2018** – is an environment used to manage SQL infrastructure.

3. References

[1] "ICT and DRR Gateway," United Nations Escape, [Online]. Available:
https://drrgateway.net/e-resilience/tool/sahana-disaster-management-software. [Accessed 17 03 2023].

[2] "Indeed," [Online]. Available:
    https://www.indeed.com/career-advice/career-development/iterative-vs-incremental. [Accessed
    20 03 2023].

[3] "Nimble Humanize work," [Online]. Available:
    https://www.nimblework.com/agile/iterative-and-incremental-development/. [Accessed 20 03
    2023].

4. Appendix A – Plan of Action

Description of Work	Start-end days

Project Planning	• Conducting a literature review regarding the broader topic area "Disaster Management"	1st March – 10th March
	• Identify and gather requirements on finalized project topic, "Disaster Management System for Sri Lanka"	1st March– 10th March
	• Doing a thorough investigation on technologies adapted in social networks	10th March – 20th March
	• Conducting a stakeholder analysis	10th March – 21st March
	• Finalizing the project proposal	22nd March – 23rd March
Project Proposal Evaluation		**23rd March**
Design	• Design UML diagrams with the identified requirements	24th March – 27th March
	• Sketch and wireframe the interfaces	28th March – 4th April
	• Prototype and Visual UX/UI designs	5th April – 20th April
Project Proposal Evaluation		20th April
Coding	• Create database (tables and relationships)	21st April – 26th April
	• Create sign up, login pages with user profiles	27th April – 2nd May
Interim Evaluation (Report + Presentation)		**2nd May**
	• Web application development	2nd May– 12th May
	• Mobile application development	13th May – 22nd May
	• Unit/ module testing	23rd May – 25th May
	• Coding for the search process	26th May – 31st May
Prototype Demonstration		**31st May**
	• Additional coding and finalizing	1st June – 9th June
Testing	• Design test cases and validate requirements	10th June – 16th June

	● Testing documentation	17th June – 26th June
Final Report Submission		**26th June**
Final Project Evaluation		**30th June**

Appendix B – Gantt Chart

	ACTIVITIES	ASSIGNEE	EH	START	DUE	%
	Planning:		-	01/Mar	23/Mar	0%
1	Conducting a literature review		-	01/Mar	10/Mar	0%
2	Identify and gather requirements		-	01/Mar	10/Mar	0%
3	Doing a thorough investigation on tec…		-	10/Mar	20/Mar	0%
4	Conducting a stakeholder analysis		-	10/Mar	21/Mar	0%
5	Finalizing the project proposal		-	22/Mar	23/Mar	0%
6	Project Proposal Evaluation		-	23/Mar	23/Mar	0%
	Design:		-	24/Mar	20/Apr	0%
8	Design UML diagrams with the identi…		-	24/Mar	27/Mar	0%
9	SRS design submission		-	27/Mar	27/Mar	0%
10	Sketch and wireframe the interfaces		-	28/Mar	04/Apr	0%
11	Prototype and Visual UX/UI designs		-	05/Apr	20/Apr	0%
	Coding:		-	21/Apr	09/Jun	0%
13	Create database (tables and relations…		-	21/Apr	26/Apr	0%
14	Create sign up, login pages with user …		-	27/Apr	02/May	0%
15	Interim Evaluation (Report + Presenta…		-	02/May	02/May	0%
16	Web application development		-	02/May	12/May	0%
17	Mobile application development		-	13/May	22/May	0%
18	Unit/module testing		-	23/May	25/May	0%
19	Coding for the search process		-	26/May	31/May	0%
20	Prototype Demonstration		-	31/May	31/May	0%
21	Additional coding and finalizing		-	01/Jun	09/Jun	0%
	Testing:		-	10/Jun	30/Jun	0%
23	Design test cases and validate requir…		-	10/Jun	16/Jun	0%
24	Testing documentation		-	17/Jun	26/Jun	0%
25	Final Report Submission		-	26/Jun	26/Jun	0%
26	Final Project Evaluation		-	30/Jun	30/Jun	0%