**Programming Data Structures and Algorithms-1**

**Higher National Diploma in Software Engineering 24.2F**

**Smart Disaster Evacuation System - Course Work**

COHNDSE242F - 070 - R.A. RAVIKUMAR

COHNDSE242F – 052 – W.S.A.T. SEMAL FERNANDO

COHNDSE242F – 044 – ABDUR FAREED

COHNDSE242F – 045 – UMAR FAREED



**School of Computing and Engineering**

**National Institute of Business Management**

**Colombo-7**

­

Contents

[Chapter 01: Introduction 2](#_Toc206662893)

[1.1 About the Proposed System 2](#_Toc206662894)

[1.2 System Inputs 3](#_Toc206662895)

[1.3 System Processes 3](#_Toc206662896)

[1.3 System Outputs 3](#_Toc206662897)

[1.4 Data Structures 3](#_Toc206662898)

[1.5 Algorithms 4](#_Toc206662899)

# Chapter 01: Introduction

## 1.1 About the Proposed System

Natural disasters such as flooding, landslides, earthquakes and even forest fires have been common in Sri Lanka and countries around the world. When these disasters strike it is important that people are able navigate by means of evacuation routes to their nearest shelter. However these routes can become unstable due to dynamic conditions such as blocked/flooded roads and shelters becoming increasingly overcrowded. Traditional systems are not able to address this issue due to their reliance on static maps and routes, which as expected may become unsafe due to the conditions mentioned above.

By combining the use of various data structures, algorithms and real-time data (maybe simulated for this project) we aim to provide a solution to the above mentioned problem and thus significantly improve how the public can response to a disaster effectively.

Therefore the need for a system in this aspect is well established.

## 1.2 System Inputs

User Inputs:

* Current location coordinates.
* Enter user’s RFID (simulated) when entering shelter.
  + Tags give data such as family count, number of children and elderly.

Sensor Inputs (simulated)

* Hazard sensor example: fire, temperature, flood level, vibration, etc.
* Road block sensors data example: An ultrasonic can detect obstacles blocking the path.

## 1.3 System Processes

* Update the user’s current position in the graph.
* Update route based on shelter capacity.
* Update nodes as unsafe based on sensor data.
* Notify rescue personal if user is unable to evacuate.
* Update a user’s rescue priority using merge sort.
* Send user SMS or notification to evacuate.
* Add user’s RFID to queue when they enter the shelter.

## 1.3 System Outputs

* Display users with the safest evacuation route.
* Display users with the nearest shelter with capacity.
* Display users with alerts if the current route becomes unstable.
* Display list of users that need to be rescued based on priority.

## 1.4 Data Structures

* Graphs
* Queue
* Arrays

## 1.5 Algorithms

* Dijkstra algorithm to find the shortest-path.
* Dynamic route calculation based on adjusted weights when a disaster blocks a path. (Re-runs Dijkstra algorithm)
* Merge sort is used to identify several factors regarding shelters such as distance, capacity and also user based factors such as children, elderly people
* Queue-based FIFO (first-in-first-out) algorithm to add people to a shelter in an orderly way that ensures fairness and prevents overcrowding.