# Chapter-1

Introduction

Road Repair System is the web application to solves all kind of cracks damages in the roads. The purpose of this software is to manage the roads in the city. We develop this software to maintain proper and suitable roads. This project is developed using HTML, CSS, JAVASCRIPT and JQUERY as front end and NODEJS as back end. Here we are trying to develop such type of system which report a pothole or road defect. Admin repairs portholes and other road defects to ensure safe access for all road users, and maintain the structure of the road or footway in accordance with our highway safety inspection manual. In auto road repair system there are few steps:-

1. User can request with some details; which area roads are defected or cracks by capturing photos and submitted on the application.
2. Allocated area supervisor taking all information of defected roads.
3. Supervisor allots a right contactor is a hand applied product that eliminates the contamination of adjacent areas, which can be problematic with other patch repair products its benefits include

* Zero waste generated with no excavation.
* Mixes prepared site specifically and on site eliminating waste.
* speed of application.
* Reduced traffic management costs.
* Reduced traffic disruption.
* Aesthetically pleasing finish.

## Rationale

Roads make a crucial contribution to economic development and growth and bring important social benefits. They are vital importance in order to make a nation grow and develop. In addition, providing access to employment, social health and education services makes a road network crucial in fighting against poverty. roads open up more areas and stimulate economic and social development. for those reasons, road repair request is most important system is the most important of all public assets.

## Goal

The goal of the project is to put in place new solutions for maintenance of roads as per the end user expectations; so that the <accuracy, speedup, digitalization> of the system will be enhanced. To do this:

1. Provide an efficient system for requesting repairs of roads
2. Provide effective solutions for road maintenance
3. Provide simpler system for user admin communication

## Objective

The objective of the works is to propose options for requesting and repairing roads To do this it requires to:

1. Review and study different apps and road repair systems
2. Find out difference
3. Propose a faster solution for road repairs

Checkpoints:

1. Request for road repair

2. Allot Engineer

3. Request Quotation

4. Allot Constructor

5. Perform Repairs

6. Change request status

## Role

Role of individual member may in each phase of software development or anything else.

1. Chandra Pratap Singh Mandloi 0832IT151009 Database

2. Murtaza Mehmudji 0832IT151019 Backend Logics

3. Nilesh Prajapat 0832IT151022 Documentation

4. Rajnish Pratap Singh 0832IT151033 Front End

## Contribution of Project

### Market Potential

The project has great market potential as the government will be have a great option with this system.

### Innovativeness

### Usefulness

This project is useful as it will contribute a lot towards the development of the cities.

# Chapter-2

Requirement Engineering

<Some introduction of requirement engineering, focused and brief>

## 2.1 Requirement Collection

### 2.1.1 <Collection Type 1>

### 2.1.2 <Collection Type N>

## 2.2 Requirements

BR 1:

BR N:

TR 1:

TR N:

# Chapter-3

Analysis & Design

## 

## Use-case Diagrams

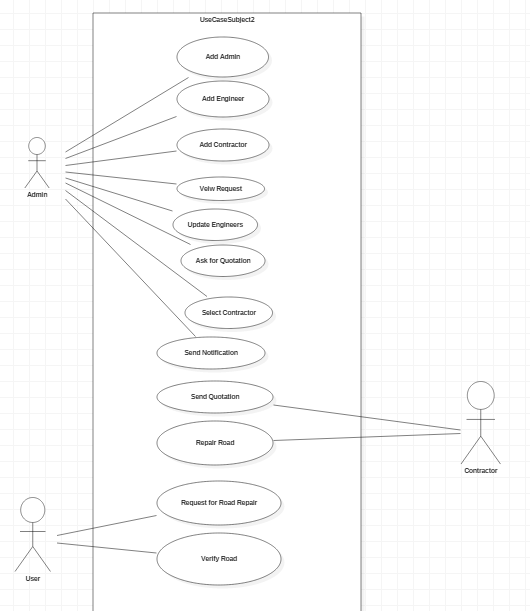


Figure 3.1: Use-case Diagram of <<>>

## Activity Diagrams

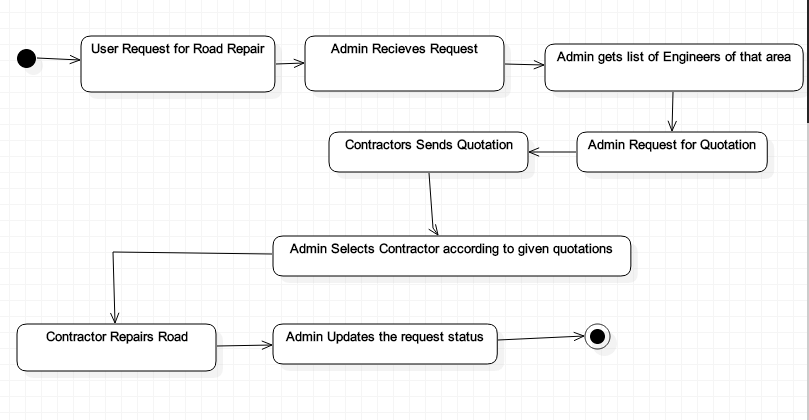


Figure 3.2: Activity Diagram of the <<Road Repair Request System>>

## 3.3 Sequence Diagrams

## 3.4 Class Diagrams

## 3.5 Data Design

### 3.5.1 Schema Definitions

Table 3.1: Schema for <<>>

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

### 3.5.2 Integrity Constraints

## 3.6 System Architecture

# Chapter-4

Construction

## 4.1 Implementation

### 4.1.1 Implementation Details

#### 4.1.1.1 Software Details

* NodeJS
* HTML5
* CSS3
* Node Modules
* StarUML
* Bootstrap
* MongoDB

#### 4.1.1.2 Hardware Details

* Server:-
  + 4GB RAM
  + Quad Core Processor(Preferred)
  + 1GB Storage
* Client:-
  + 1GB RAM
  + Dual Core Processor
  + 1GB Storage

## 4.2 Testing

### 4.2.1 White Box Testing

Test Case: 1

Table 4.1: Test Case for <<>>

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Test Case: N

### 4.2.2 Black Box Testing

Test Case: N+1

Test Case: M

# Chapter-5

Conclusion & Future Works

Appendix A

