# Software Design Document

# High Performance Deep Learning for disaster detection

Harshita .B
Sonal Dharmik

# Contents

# Chapter 1

# Introduction

This document discusses about the design and architecture used for the system.Also ,it provides information of components

## 1.1 Design Overview

This application will identify the damaged areas that have been affected by disaster using CNNs. The user has to feed the satellite image of the area as input for the application for damage detection. The application will process the image and extract the necessary features. After feature extraction, the application will detect the damaged areas and return the image having highlighted damaged areas as the output. The application uses component-based architecture. The application can be broken down into the following individual components - Image pre-processing, image segmentation, feature extraction, classification and damage detection.

## 1.2 Requirements Traceability Matrix

| Requirements/Components | User Interface | Application Logic | Result Display |
|---|---|---|---|
| Input Image | X | | |
| Process Image | X | X | |
| Classification | | | X |

# Chapter 2

# System Architectural Design

## 2.1 Chosen System Architecture

The component-based architecture describes a software engineering approach to system design and development. We focus on the decomposition of the design into individual functional or logical components that expose well-defined communication interfaces containing methods, events and properties. This also provides a higher of abstraction compared to object oriented design.

## 2.2 Discussion of Alternative Designs

The alternative architectural model that can be used are Layered architecture and Data-flow architecture. Layered architecture focuses on grouping of related functionality within an application into distinct layers that are stacked vertically on top of each other. They are mainly suited for 3-tier/n-tier applications. Data-flow architecture is much more suited for web application based projects. It is suitable for applications that involve a well defined series of independent data transformations or computations on orderly defined input and output such as compilers and business data processing applications.

# Chapter 3

# System Interface Description

## 3.1 Detailed Description of Components

### 3.1.1 Component-1 : Load Image

| | |
|---|---|
| Responsibilities | Provides an interface to user to enter a near - real time image into the system. |
| Composition | Front - end: Python |
| Interactions | User enters an image into the system. |

### 3.1.2 Component-2: Segmented Image

| | |
|---|---|
| Responsibilities | Provides an interface where the segmented image is displayed. |
| Composition | Front - end: Python |
| Interactions | User can view the segmented image. |

### 3.1.3 Component-3: Classification

| | |
|---|---|
| Responsibilities | Provides an interface to view the damaged areas in the image after the image is processed. |
| Composition | Front - end: Python |
| Interactions | User can view the classified image. |

# Chapter 4

# System Architecture

## 4.1 Use Case Specification using template 1

| Use case ID: | 1 | | |
|---|---|---|---|
| Use case name | Upload Image | | |
| Created By: | Sonal Dharmik | Last updated by | Sonal Dharmik |
| Date created: | | Date last updated: | |

| Primary Actors | User |
|---|---|
| Secondary Actors | |
| Description | The user can use the application by uploading the image to be processed |
| Trigger | The user requests to start the application |
| Preconditions | |
| Post-conditions | The user can upload the image for preprocessing |
| Normal flow | 1.User opens the application 2.User clicks on the load image button. 3.User uploads the image |
| Alternative flow | |
| Exceptions | |
| Includes | Upload image |
| Priority | 1 |
| Frequency of use | Number of times |
| Business Rules | |
| Special requirements | Image has to be valid and uploaded for further processing |
| Open issues | |
| Assumptions | All details should be valid |
| Notes and issues | |

## 4.2 Use Case Specification using template 2

| Use case ID: | 2 | | |
|---|---|---|---|
| Use case name | Image segmentation | | |
| Created By: | Harshita Bayeti | Last updated by | Harshita Bayeti |
| Date created: | | Date last updated: | |

| Primary Actors | User |
|---|---|
| Secondary Actors | |
| Description | The user can view the result of the segmented image |
| Trigger | The user clicks on the detect button. |
| Preconditions | The user needs to upload the image. |
| Post-conditions | |
| Normal flow | 1.User clicks on detect button. 2.User can view the results of the segmented image. |
| Alternative flow | |
| Exceptions | |
| Includes | |
| Priority | 2 |
| Frequency of use | |
| Business Rules | |
| Special requirements | |
| Open issues | |
| Assumptions | |
| Notes and issues | |

## 4.3 Use Case Specification using template 3

| Use case ID: | 1 | | |
|---|---|---|---|
| Use case name | Analysis | | |
| Created By: | Sonal Dharmik | Last updated by | Sonal Dharmik |
| Date created: | | Date last updated: | |

| | |
|---|---|
| Primary Actors | User |
| Secondary Actors | |
| Description | The user can view the result of the uploaded image. |
| Trigger | The user clicks in the detect button |
| Preconditions | The image that enters this module has to be segmented. |
| Post-conditions | |
| Normal flow | 1.User clicks on detect button. |
| Alternative flow | |
| Exceptions | |
| Includes | Image |
| Priority | 3 |
| Frequency of use | |
| Business Rules | |
| Special requirements | |
| Open issues | |
| Assumptions | |
| Notes and issues | |

## 4.4 Use Case Specification using template 4

| | | | |
|---|---|---|---|
| Use case ID: | 1 | | |
| Use case name | Classification | | |
| Created By: | Sonal Dharmik | Last updated by | Sonal Dharmik |
| Date created: | | Date last updated: | |

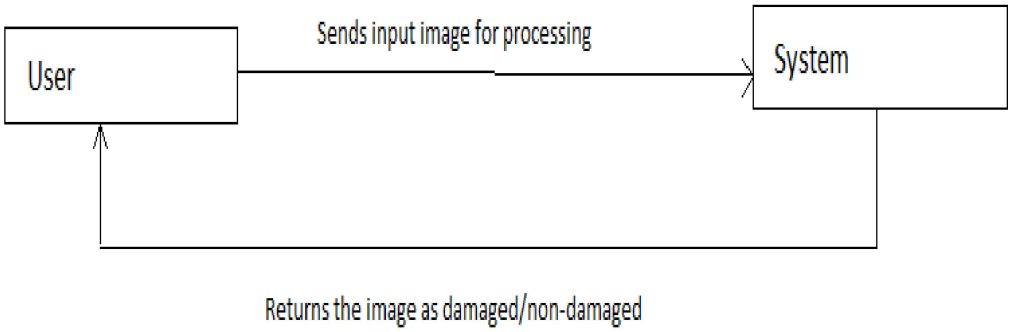
| | |
|---|---|
| Primary Actors | User |
| Secondary Actors | |
| Description | The user can view the classification of the damaged regions present in it |
| Trigger | The user clicks on the detect button |
| Preconditions | The image needs to go through the specific steps of feature extraction and then classification based on the trained model. |
| Post-conditions | |
| n Normal flow | 1.User clicks on classification button |
| Alternative flow | |
| Exceptions | |
| Includes | Image |
| Priority | 4 |
| Frequency of use | |
| Business Rules | |
| Special requirements | |
| Open issues | |
| Assumptions | |
| Notes and issues | |

# Chapter 5

# Data FLow Diagrams

## 5.1 Level 0 DFD

**Work flow diagram:** Below given is the workflow of the software to be made.



## 5.2 Level 1 DFD

**Work flow diagram:**
Below given is the workflow of the software to be made.