SOFTWARE PROJECT MANAGEMENT PLAN

High Performance Deep Learning for disaster detection

Harshita .B
Sonal Dharmik

Contents

Chapter 1

Introduction

This document discusses about the overall project overview and the responsibilities and tasks that have to be performed in the plan.The roles of each and every individual in the project is also mentioned clearly in the document.

1.1 Project Overview

Last two decades witnessed the increasing use of remote sensing for understanding the geophysical phenomena underlying Natural hazards.Earthquake disaster detection requires high speed computation as the changes need to be detected in real time. Thus our approach is to do this analysis on GPU with the help of deep learning libraries. The software will have an option to input the image for damage detection. The output will be an image highlighting the damaged area and the percentage of area that is damaged.

1.2 Project Deliverable

The table shows the documents to be submitted during the development of the software and the delivery dates of the document.

Deliverables	Description	DeliveryDate
Submission of Project details	This includes scope of the project and people involved and guidelines to be followed right through the project.	15/07/2018
Project presentation - 1	The project will be presented to the panel and any modifications suggested will be taken into consideration and implemented.	10/08/18
Software Project Management Plan(SPMP)	This includes the outline of the project and people involved an detailed process and guidelines to be followed right through the project.	15/10/18
Software Requirement Specification (SRS)	This includes details about ht efunctional and non-functional requirements.	15/10/18
Software Design Document(SDD)	Contains in-depth design of the software including uml diagrams if applicable and tasks to be done and risk tables	25/10/18
Software Test Document(STD)	Details about the types of tests to be carried on the software to ensure that the system meets the requirements and maintains the integrity.	25/10/18
Project Presentation and Demonstration -II	Project presentation of whatever has been implemented and understood so far.	31/10/18
Implementation of modules	Here the modules are implemented one by one and approved by the guide.	30/11/18
Demonstration	Here the final project demonstration will be done and presented to the panel.	15/02/19
Report Submission	The final project with the final report and demonstration will be submitted.	15/03/19

Chapter 2

Project Organization

This chapter gives information about the process model followed and the roles and responsibilities of the project members.Also,it gives information about the tools and techniques of the software.

2.1 Software Process model

Waterfall model

In "The Waterfall" approach, the whole process of software development is divided into separate phases. The outcome of one phase acts as the input for the next phase sequentially. This means that any phase in the development process begins only if the previous phase is complete.

Characteristics of Waterfall Model

1.Adapts to Shifting Teams
2.Used when requirements are well understood and risk is low
3.Work flow is in a linear (i.e., sequential) fashion
4.Forces Structured Organization
5.Allows for Early Design Changes

2.2 Roles and Responsibilities

The team members in this project are:

1. Sonal Dharmik
2. Harshita Bayeti

Both the team members will be responsible for all the activities of the project The activities are:

1. Collecting the data
2. Requirements gathering
3. Deciding the technology to be used
4. Learning the new technology

5. Feature extraction
6. Clustering images
7. Developing the neural network model using tensorflow for classification of images.
8. Refining the model for accuracy (Re-training)
9. Testing
10. Documentation of the project

2.3 Tools and Techniques

1. Language that will be used is Python for the software.
2. Hardware requirements for the system will be
   (a) Central Processing Unit (CPU) — Intel Core i5 6th Generation processor or higher.
   (b) RAM — 8 GB minimum
   (c) Graphics Processing Unit (GPU) — NVIDIA GeForce GTX 960 or higher.
3. Software Requirements for the software:
   (a) Deep learning library that will be used to train the model.
   (b) Operating system that the software can work on is Ubuntu or Windows 10
   (c) Python IDE etc will also be required.
4. Documentation will be done in Latex with given format only.

Chapter 3

Project Management Plan

3.1 Tasks

3.1.1 Task 1: SRS(Software requirement Specification) - T1

Description : The SRS document will identify and describe the requirements needed to develop the application.

Deliverables and milestones: The deliverable in the form of a verified SRS document.

Resources needed: The SRS documents will be generated by the team members using Latex based on the literature review and research done by the team.

Dependencies and constraints: The project must be approved by the panel.

Risks and contingencies: The SRS may go under Change Control Processes in case there is a requirement.

3.1.2 Task 2: SDD (Software Design Document) - T2

Description : This document will be used to specify the design for implementation of the project in the form of various diagrams that will specify the workflow of the project.

Deliverables and milestones: The team members will generate a Software Design Document which will be verified and submitted as the deliverable,.

Resources needed: Latex, Rational Rose

Dependencies and constraints: T1-Software Requirement Specification Document.

Risks and contingencies: During the progress of the project, it may be required to revise the document.

3.1.3 Task 3: Code and Test - T3

Description : The objective of the coding phase is to implement the conceptual model into a working application.

Deliverables and milestones: To transform the design of a system into a working application.

Resources needed: Internet, CPU, GPU

Dependencies and constraints: The project must be approved by the assigned panel. To start coding the design of the project must ready, Hence SRS and SDD needs to be prepared prior to the implementation.

Risks and contingencies: Code or class libraries have poor quality which may cause extra testing, defect correction and rework.

3.1.4 Task 4: User Interface - T5

Description : This task aims at integrating the modules and provide the user an attractive, easy-to-use interface.

Deliverables and milestones: A fully completed Software system for detection of damaged areas in the images.

Resources needed: Satellite images, Internet

Dependencies and constraints: T2 - Software Design Document should be ready. T3 - Code to develop the software application in python should be known.

Risks and contingencies: The complexity of modules may cause problems in integration.

3.1.5 Task 5: Code and Test - Bug Fixes - T5

Description : A Software Test Plan is a document describing the testing scope and activities. It is the basis for formally testing any software or product in a project.

Deliverable and milestones: Determine the scope and the risks that need to be tested and that are not to be tested.

Resources needed: Internet

Dependencies and constraints: T4 - Software should be ready so that it can be tested.

Risks and contingencies: If attempts are made retroactively to repair code, it can end up making bug fixing a potentially damaging and costly pursuit.

3.2 RIS:

Risk	Category	Probability	Impact
Change in requirements	PS	15	2
Lack of training on tools	TI	40	3
Late deliverables	BU	20	2
Accuracy not achieved	TI	25	1
Computer crash	TI	10	3

1–Catastrophic
2–Critical
3 - Marginal
4 -Negligible

3.3 Timetable: