CODING STANDARDS FOLLOWED DURING THE HACKATHON PROJECT

	Prepared By	Reviewed By	Approved By
Name	Unhandled Exceptions		
Role	Programmer Analyst Trainees	Batch Trainer	

CONTENTS

1.INTRODUCTION

1.1.PURPOSE :

The goal of these guidelines is to create uniform coding habits among software personnel in the engineering department so that reading, checking, and maintaining code written by different persons becomes easier. The intent of these standards is to define a natural style and consistency, yet leave to the authors of the engineering department source code, the freedom to practice their craft without unnecessary burden.

Experience over many projects points to the conclusion that coding standards help the project to run smoothly. They aren't necessary for success, but they help. Most arguments against a particular standard come from the ego. Few decisions in a reasonable standard really can be said to be technically deficient, just matters of taste. So, in the interests of establishing the engineering department as a showcase software development environment, be flexible, control the ego a bit, and remember any project is a team effort.

1.2.SCOPE :

This document describes general software coding standards for code written in the hackathon project submitted along with and can be used for any text based programming language (including high-level languages like C, C++, Basic, Visual Basic, and assembler languages). This will be used as the base document for the code written throughout the project.

2.Coding Standards Used

2.1. Naming Conventions :

Java naming conventions are sort of guidelines that application programmers are expected to follow to produce a consistent and readable code throughout the application. If teams do not follow these conventions, they may collectively write an application code that is hard to read and difficult to understand.

Java heavily uses Camel Case notations for naming the methods, variables etc. and Title Case notations for classes and interfaces.

1. Packages : Package names used are a group of words starting with all lowercase names starting with the project name. Subsequent parts of the package name may be different according to different packages.

Examples : findinghospitals.utilities , findinghospitals.testcases.

2. Class : Class names generally used in the project are nouns, in title-case with the first letter of each separate word capitalized.

Examples : Testcases, LandingPage etc.

3. Methods : Method names used in the project are verbs. They represent action and the method names clearly state the action they perform. The method name can be single or 2-3 words as needed to clearly represent the action. Words are in camel case notation.

Examples : fillName(), fillEmail(), printCityList() etc.

4. Variables : All instance, static and method parameter variable names used in the project are in camel case notation. They are short and enough to describe

their purpose. Temporary variables are a single character e.g. the counter in the loops.

Examples : nameElement, hospitalList etc.

2.2. Code Comments Standard :

The comments are consistent with what they describe and they do not state the obvious. They are clear and concise making it readable for any future maintainer.

2.3. Indentation Standard :

Default indentation standard of the Eclipse IDE is used.