Developer Guide

Johannesburg Emergency management System

Emergency management system is created by Targarian Technlogies Team 23

15/10/2015

Web development

Coding style and tools used

Scope

This section entails guidelines to be used by developers who are working on Johannesburg emergency management system code.it will describe the structure of the code or layout and the choices that were made in order to develop the system. all code that relating to front-end which involves web design and front-end web development can be found in .aspx,.css and jquery files and all the of the backend or server side code will be found can be found in .cs files

Goals

Main goal was to make sure that the style used to develop this software system was consistent and of higher standards throughout the development process so that it can be easy to understand and read our code. the standard style used helped us to produce a code of high quality and it also helped with code abstraction. We also wanted to ensure that anyone who wants to use our code can understand it much easier by using packages, and also making we avoid repeating code by use of code-reuse technique, we also documented our code for easy understanding.

Tools used

Framework that is used to develop the system is .NET and the IDE used is Microsoft visual studio 2012,and the programming language that was used is C# because it is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines.

To download visual studio follow this link : https://www.visualstudio.com/en-us/downloads/download-visual-studio-vs.aspx

CSS,BOOSTRAP,HTML and JQuery

Asp.net web pages are the same as html web pages the only difference is the extension name, so it was important for us to make sure that the mark-up language we used conforms to World Wide Web Consortium standards as it is main international standards organization for the World Wide Web. CSS was

used to style web pages and Bootstrap together with JQuery were used to make sure that the page is responsive.

To get JQuery UI Bootstrap you can just follow this link : https://jquery-ui-bootstrap.github.io/jquery-ui-bootstrap/

API

API is the most important part of the system because it specifies protocols, routines and tools for building software application it also specifies how software components should interact.

We have a page API which specifies the structure of the page it includes the following

- Setting up a page url
- Setting up page title
- Setting the heading for a page

Johannesburg emergency management system web application is made up of the following pages

Home.aspx	AddEmployee.aspx
ChangeAdmin.aspx	CompleteEmergency.aspx
FireReport.aspx	Logs.aspx
NumEmergenciesMonth.aspx	OperatorReports.aspx
StatusCancelled.aspx	STatusDispatched.aspx
updateEmergency.aspx	ViewEmployees.aspx
Dispatch.aspx	DriverStatus.aspx
EmergenciesPerDay.aspx	Login.aspx
MedicalReport.aspx	NumEmergenciesReport.aspx
Profile.aspx	StatusComplete.aspx
StatusOnRoute.aspx	UsersOfSystem.aspx

Home.aspx : this is the page that you will be redirected to after logging in.

AddEmployee.aspx : this is the page that can only be accessed by an administrator when he/she wants to add details of a new employee to the database.

ChangeAdmin.aspx : this is the page that allows for a change of administrators.

Logs.aspx : this is the page that shows emergencies that have been reported by civilians.

viewEmployees.aspx : this is the page that can be viewed if administrator want to view employee details.

Profile.aspx : this is the page that is used by employees to check if their profiles are up-to-date.

Login.aspx : this is the login page whereby a user will enter their user id and passwords

Dispatch.aspx : this is the page that is used to dispatch a vehicle.

updateEmergency.aspx : this is the page that is used to update emergency priority level and other necessary information.

statusDispatched.aspx : this is the page that will notify an operator about dispatched vehicles.

statusCancelled.aspx : this page checks if the emergency has been cancelled.

statusComplete.aspx : this page checks if the vehicle has arrived safely at the emergency scene.

statusOnRoute.aspx : this page is when the operator checks vehicles that are still on route.

completeEmergency.aspx : this page display report of an emergency that has a status of complete

fireReport.aspx : this is the page to show a report for number of fire emergencies that have been reported.

operatorReports.aspx : this page shows number of employees that are active, which can be administrator or operator.

driverStatus.aspx : this is the page that an operator can view if he/she wants to view status of a driver.

numEmergenciesMonth.aspx : this page display a report of number of emergencies that have been reported for every month.

emergenciesPerDay.aspx : this is the page to use when you want to view number of emergencies that have been reported daily.

medicalReport.aspx : this is a page to view reports of medical emergencies.

numEmergenciesReport.aspx : this all emergencies that are being handled.

usersOfSystem.aspx : this are all users of the system.

Master.masterPage : this is the page that is used as a template for all other pages.

Form elements

The following are the most important form elements used when creating emergency management system.

Button	Inputboxes
Div element	Form group

Maximum Line Length

Our aim is to keep the our code line length at 150 characters if it is convenient,it is not recommended to use more than 190 characters as we were very strict when it comes to the standards regarding line length.

Naming convention

Everything should be named as follows:

- Names of methods and variables should be as short as possible
- Use camel case to name methods
- When you name variables you can use camel case or start with an underscore
- All naming should be done in English
- File extensions should either be .html, .aspx , .cs ,.js,.xml

Classes

Classes should be named in such a way that they explain the type of data it holds and it should be named in English. Class names can be written in camel case only

public class Employees

public class UserDataService

Every class must have documentation that explains its existence and it should conform to C# standards,member variables should be named according to our standards of naming.Any global variable that exist should be declared first before any method,visibility for member variables should always be specified.it can either be public, protected or private. Declaring a member variable as a public variable is discouraged instead you should use get and setter methods.

Methods

Method names should be written in English and explains the existence of that method,naming convention should be camel case.

```
public string dateOfBirth
    {
        get
        {
            return Emp_dateOfBirth;
        }

        set
        {
            Emp_dateOfBirth = value;
        }
    }
```

Variables

A variable name should be meaningful and easy to understand,they should be kept as short as possible. Don't use terms like variable1 for example when you are declaring them.

```
[DataMember]
        private string EmpName;
        private string EmpSurname;
        private string EmpType;
        private string EmpAddress;
        private string EmpGender;
        private string EmpEmail;
        private int EmpCellno;
        private string EmployeeId;
        private string EmpDateOfBirth;
```

Exception Handling handling

We used exception handling to ensure that the system can recover from crashes without causing any loss availability and also to handle unexpected user inputs.

```
try
    {
```

```
        MailMessage mail = new
        MailMessage("201317579@student.uj.ac.za",
        "kgotso.chacha@gmail.com");
    SmtpClient clients = new SmtpClient();
    clients.Port = 25;
    clients.DeliveryMethod = SmtpDeliveryMethod.Network;
    clients.UseDefaultCredentials = false;
    clients.Host = "smtp.google.com";
    mail.Subject = "Reference Number";
    mail.Body = "Your emergency is being attended to. Here is your
reference number" + ref_num;
    clients.Send(mail);
}
catch (SmtpException ex)
{
    string error = "Messsage could not be sent";
}
```

Database and Data Manipulation

Microsoft Sql Server 2014 was used as a DBMS for Johannesburg Emergency Management System it can be downloaded on this site :
https://msdn.microsoft.com/en-us/sqlserver2014express.aspx

Instructions passed include:

CREATE TABLE creates a table with the given name.

UPDATE TABLE update a table with the given information.

SELECT TABLE select specified table columns.

DROP TABLE delete a table with a specified name.

Following are the tables that are used to handle Johannesburg Emergency Management system data:

Person : this table holds information of all the users of the system

Employee : this table holds information of employees of Johannesburg Emergency Management Services

Civilian : this table holds all information of civilians

CivilianNextOfKin : this table holds name and contact details of civilian next of kin

EmergencyList : this table holds names of all available emergencies

Dispatch : This table holds all of the information relating to the vehicle that has been dispatched

Driver : this table holds information about available drivers

Emergency : this table holds all of the information relating to the emergency that has been reported

Vehicles : this table holds information about all of the vehicles that the emergency management services have

Documentation and comments
Comments explains the purpose of the code,what methods does, and also give a brief summary of classes used.

C# commenting was used to explain backend code

```
    /// <summary>
    /// Function to validate the user of the system
    /// </summary>
    /// <param name="type">the type of user</param>
    /// <param name="UseriD">the username of the user</param>
    /// <param name="password">the password of the user</param>
    /// <returns name="session">the username</returns>
```

HTML commenting style was also used

```
<%-- Button for dispatching --%>
    <asp:Button ID="BtnDispatc" runat="server" OnClick="BtnUpdateEmergency_Click"/>
```

Mobile development
Coding and style tools
Johannesburg Emergency Management System mobile application is an android application and android studio was used as an IDE to develop the application and the languages used are java and xml.

Commenting
Basic java commenting style was used to