Technical Design

Weather Application

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

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
| Version: | 0.1 |
| Authors: | Megha Punj |
| Issued: | 07/06/2016 |

Table of Contents

[2. Requirement Summary 4](#_Toc453143016)

[3. Physical Data Model 4](#_Toc453143017)

[3.1 Database Changes 4](#_Toc453143018)

[3.2 Migration Changes 4](#_Toc453143019)

[4. Detailed Design Overview 4](#_Toc453143020)

[4.1 Technical Specification 4](#_Toc453143021)

[5. System Components and Interfaces 4](#_Toc453143022)

[6. Appendix A 6](#_Toc453143023)

[6.1 Glossary 6](#_Toc453143024)

*Revision History*

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Issue Date | Description of version changes |
| v0.1 | Megha Punj | 06/06/2016 | Initial Creation |

# Requirement Summary

We need to build a web application to display current weather for 3 Australian cities: Sydney,

Melbourne and Wollongong. There should be a dropdown list on web page for city selection,

when city is changed corresponding real-time weather information should be displayed

dynamically like below:

City: Melbourne

Updated time: Thursday 11:00 AM

Weather: Mostly Cloudy

Temperature: 9°C

Wind: 32km/h

# Physical Data Model

## Database Changes

None.

## Migration Changes

None.

# Detailed Design Overview

This section provides an overview of the system and details of technical changes. Components added to achieve the desired functionality are exmplained below.

## Technical Specification

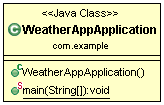
The Weather application is written in Java using the following:

* **Spring and Spring boot** – to provide the framework;
* **Maven** – used to build and deploy the application;
* **openweathermap.org/api:** API used to get the weather Information.

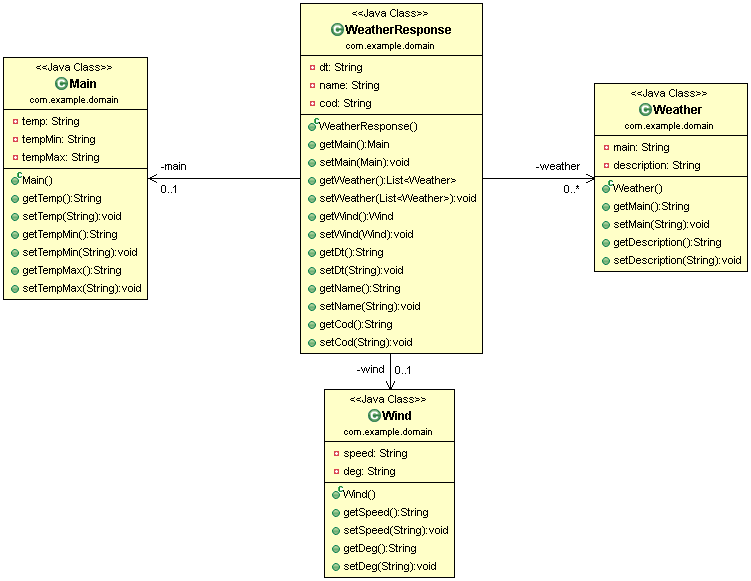
# System Components and Interfaces

The following classes are added for the application

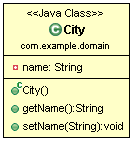
1. WeatherAppApplication.java: The file with main function. The program runs from this file.



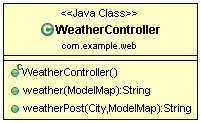
1. Domain related classes: The following classes are added as domain classes:



To fetch the city from the user, the domain class used is : City.java



1. Controller classes: Since this is a MVC application, the controller used is ‘WeatherController.java’.

****

# Appendix A

## Glossary

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| 1. API | 1. Application Programming Interface |
|  |  |