ro.mta.se.lab.Controller

LocationFactory

+generateCitiesFromDefaultCsv(): CitiesData
+generateCitiesFromCsv(String): CitiesData
+addNewCityToCsvFile(String,String,String,String,String):CitiesData
+addNewCityToDefaultCsv(String,String,String,String):CitiesData

<-- Use ----

WeatherApiUser

+ static getServerResponse(String):String
+static getWeatherByCity(City): JSONObject
+static getWeatherByCoord(String, String): JSONObject

<-- Use ----

FlagApiUser

+static getCityByCrtIP(CitiesData):City

<-- Use ----

WeatherIconConstructor

+static getImageUrl(String)

<-- Use ----

FlagApiUser

+static getImageOfFlagUrl(): String

<-- Use ----

Logger

+static logToFile(String)

<-- Use ----

WeatherController

-@FXML GuiObjects

-loadedCities: CitiesData
-crtCity:City
-crtForecast:WeekForecast

-loadCountryDropdown()
-loadCityDropDown()
-setLocationOnIP()
-addNewLocation()
-initDaysContent()
-setForecastImg()
-loadMainPanel()
-loadDailyPanel()
-loadCountryImg()
-initialize()

<-- Use ----

Exception

Extends

MtaWeatherException

+getMessage():String

ro.mta.se.lab.Exceptions

Extends   Extends   Extends   Extends   Extends

BadCsvException	**BadConnException**	**BadJSONException**	**BadLogException**	**InvalidCityException**
-message: String	-message: String	-message: String	-message: String	-message: String
+getMessage():String	+getMessage():String	+getMessage():String	+getMessage():String	+getMessage():String

WeatherController

ro.mta.se.lab.model

CitiesData

-CitiesDB: Array<City>

+getCity(): City
+addCity(City)
+getIsoOfCountry(string): String
+getCountries(): ObservableList
+getCities(String): ObservableList

City

-name: StringProperty{get}
-country: StringProperty{get}
-id: StringProperty{get}
-iso2: StringProperty{get}
-lat: StringProperty{get}
-lng: StringProperty{get}

WeekForecast

-crtForecast:WeatherForecast{get}
-timezoneOffset: Long{get}
-forecast: Array<WeatherForecast>{get}

+getCrtDate(): StringProperty
+getCrtTime(): StringProperty

WeatherForecast

-lat: StringProperty{get}
-lng: StringProperty{get}
-WeatherMainDescription: StringProperty{get}
- WeaherIcon:StringProperty{get}
-WeatherTemp: StringProperty{get}
-WratherFeelsLike: StringProperty{get}
-tempMin: StringProperty{get}
-tempMax: StringProperty{get}
-humidty: StringProperty{get}
-cloudsPercent: StringProperty{get}
-dt: StringProperty{get}
--wind: StringProperty{get}

+getCrtDate(): StringProperty
+getCrtTime(): StringProperty