

# The MVVM architecture

Model-View-ViewModel is a software design pattern that is structured to separate program logic and user interface controls. At this project we separated the view to few views, each view for each fxml component. Every change at the GUI target the bind at the MainWindowController and ViewModel which change some values. At this project we connect with the backend by opening some sockets and connect with ip and ports. At timeCapsule tab we connect directly with the flightGear simulator.

## Model

```
+getPlans():arr[Plane]
  +refresh():void
+getAvailablePlanes():int
+getMonthlyMiles():arr[string]
+getAvarageMiles():arr[string]
+getFleetSize():arr[int]
+getCorrelativeFeature(String):String
+getRudder():Double
+getThrottle():Double
+getAileron():Double
+getElevator():Double
+getMonitoringFeatures():csvFile
+getMonitoringDash():String
+setRudder(Double):void
+setThrottle(Double):void
+setAileron(Double):void
+setElevator(Double):void
+getTeleoprarionDash():String
+sendInstructions(File):void
+setTimeSeries(:TimeSeries)

+setAnomalyDetector(:AnomalyDetector)
-initActiveTabHash()

+streamSocket()
+closeConnection()

+fleetActive()
+moniAvtive()
+teleActive()
+tcActive()
+activeTab()
+currentTab()
+getPlaneProp()
+sendTeleText()
+flightRecovery()
+ObjectInputStream ob;
+BufferReader in;
+String currentId;
Data data;
```

## View Model

```
+Model m;
+ExcecuterService es;
+DoubleProperty aileron, elevator,rudder,throttle;
+DoubleProperty verticalSpeed, heading, rollDeg, long, lat;
+IntegerProperty isRunPushed, isfleetPushed, inMoniPushed;
+loadprop(FileName);
+loadAlg(FileName);
-PrintWriter pw;
```

## View

```
+ViewModel vm

$Fleet Overview
+Canvas fleetMap
+TextFlow planeProp
+GraphicsContext canvasGc;
+PieChart availablePlanes
+BarChart monthlyMiles
+BarChart avgYearlyMiles
+Linechart FleetSize
+Image background
+Image airplane
+arr[string] airplanesDetails
+Boolean isClicked
- Boolean isDoubleClicked
+refresh():void
+drawMap:void
_changeAirplaneButton():void
+PlanesLocation():void

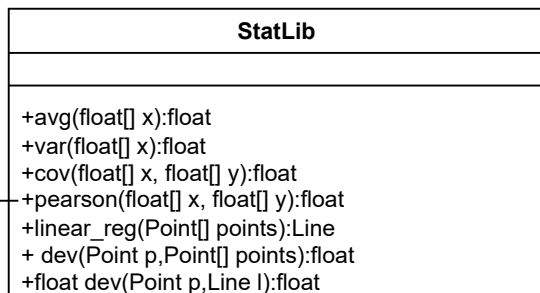
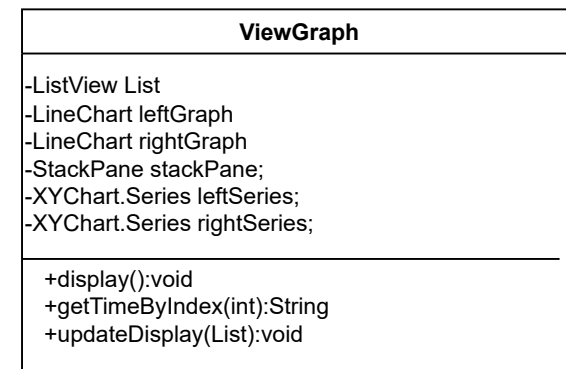
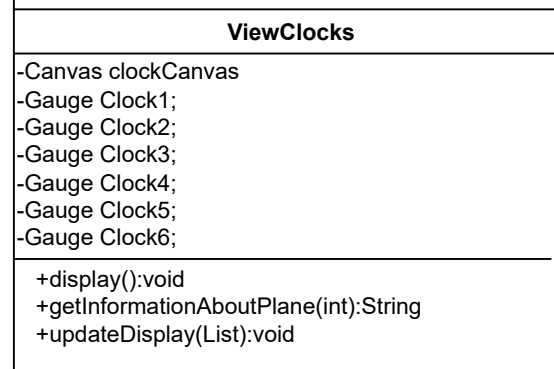
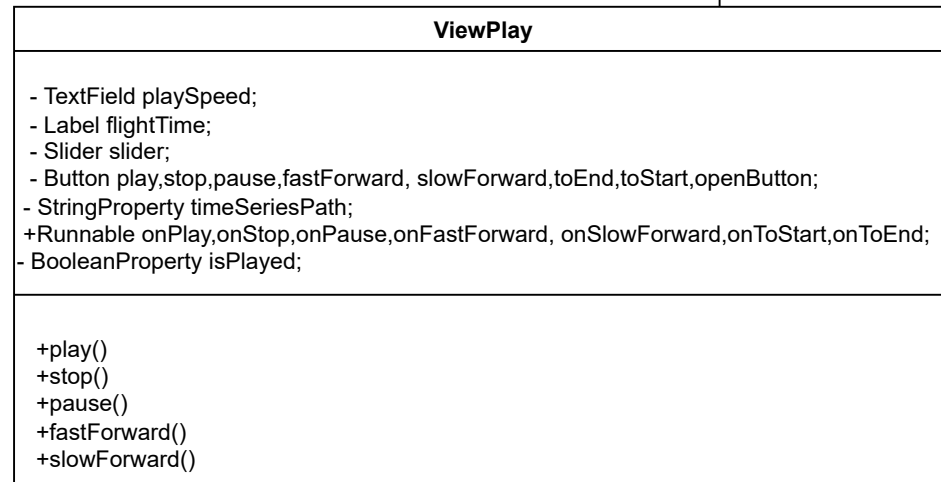
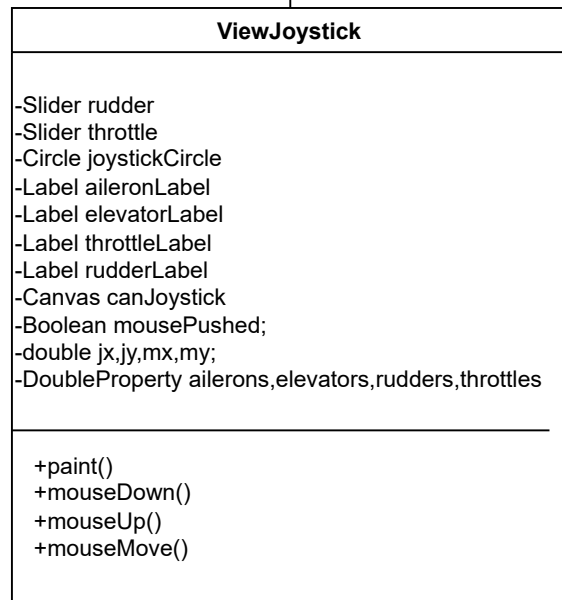
$Monitoring
-Joystick joy
-Canvas dashCanvas
-Canvas leftGraph
-Canvas rightGraph
-Canvas bottomGraph
-ListView properties
-String currntProp

$Teleoperation
-Joystick Joy
+JoystickController teleJoystickController;
-Canvas dashCanvas
-TextField CodeFromUser
-Button submit
-IntegerProperty isRunPush;

$Time Capsule
-Joystick joyTC
-Canvas dashCanvasTC
-Canvas leftGraphTC
-Canvas rightGraphTC
-Canvas bottomGraphTC
-ListView propertiesTC
-String currntPropTC
-Play p

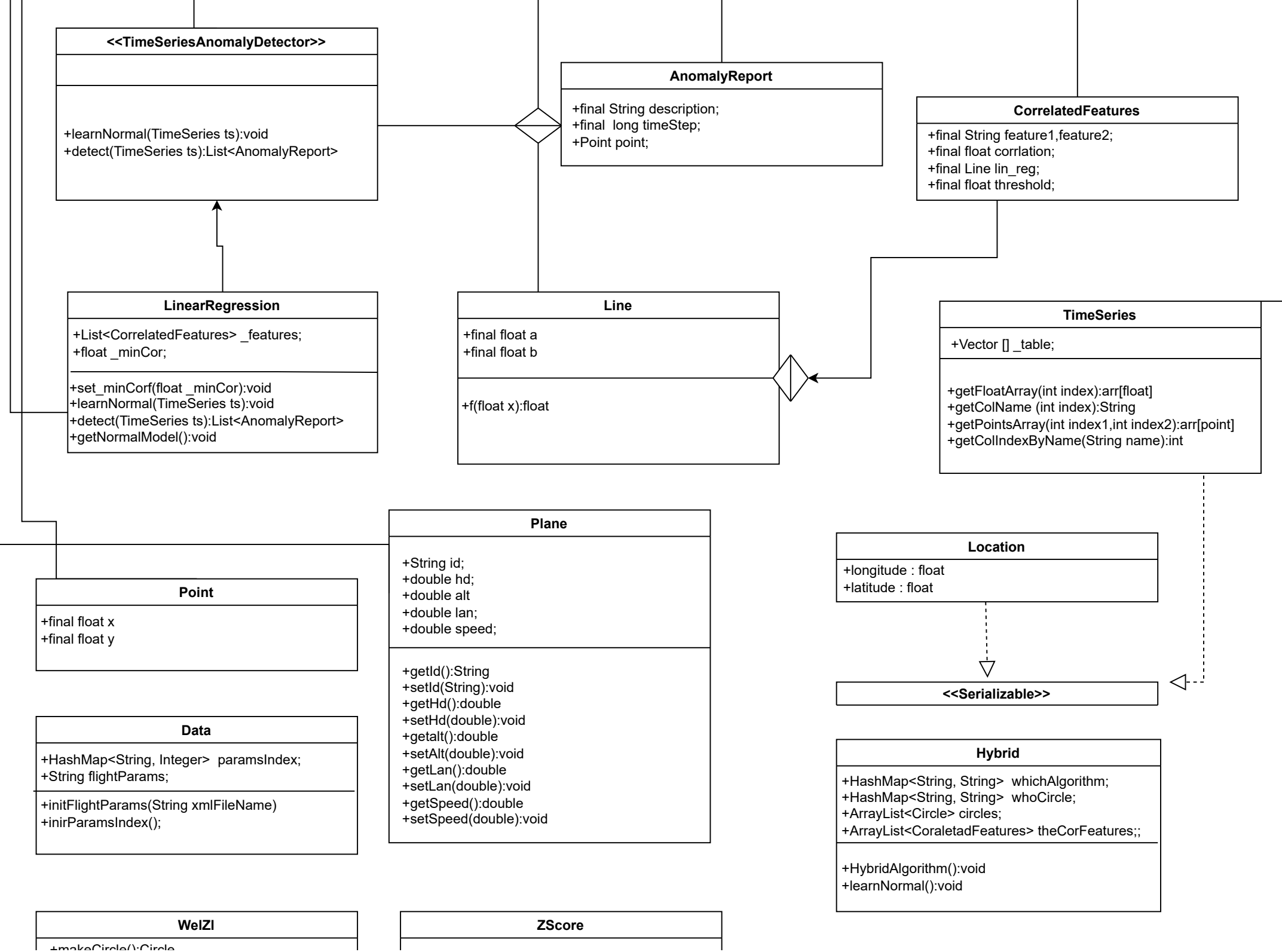
-ExcecuterService es;
-HashMap<Integer, Number> fleetSize;
-HashMap<Integer, Number> milesYear;
-HashMap<String, Number> miles;
-ArrayList<Plane> planesArray;
```

data binding



+sendCodeSubmit():void





+makeCircle():Circle  
+makeCircleOnePoint():Circle  
+makeCircleTwoPoint():Circle  
+makeCircumcircle():Circle

+learnNormal();void  
+detect():void  
+getZMax():void