*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Feature Document

User Story ID #730

**Name:** Francisco Lozada

**Team Member(s):** Lukas Borges, Filip Klepsa, Nicolette Celli, Francisco Lozada, Cristian Cabrera

**Project:** AR-VR-VE for Computer Science (Circular Gesture Recognition API)

**Product Owner(s)**: Francisco Ortega

**Mentor(s)**: Francisco Ortega

**Instructor**: Francisco Ortega, Masoud Sadjadi

**User Story Name:** Translate the DirectionalEvents Class to C++

* Description: **As a** developer **I would like** to translate the DirectionalEvents class from the MTGRLibrary to C++ **so that** it can help implement the CircGesture class for the API

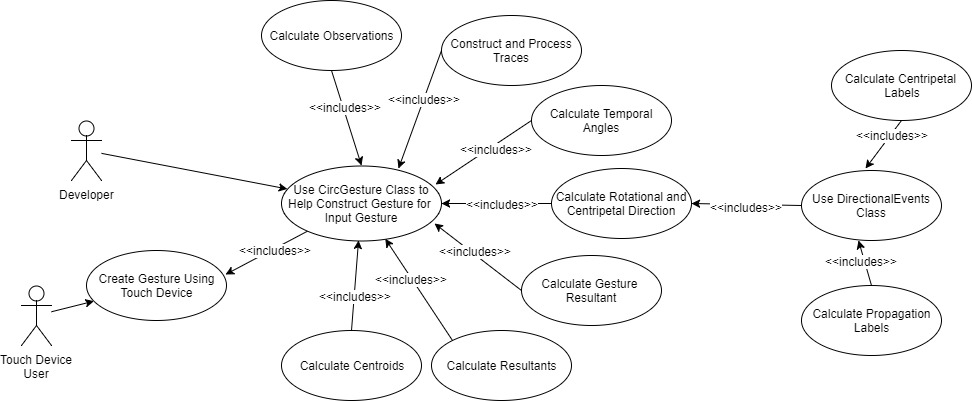
**Acceptance Criteria**

* Must have C++ API best design practices implemented
* Must have the required attributes and member functions that provide the exact translation of the class written in C# to C++ code

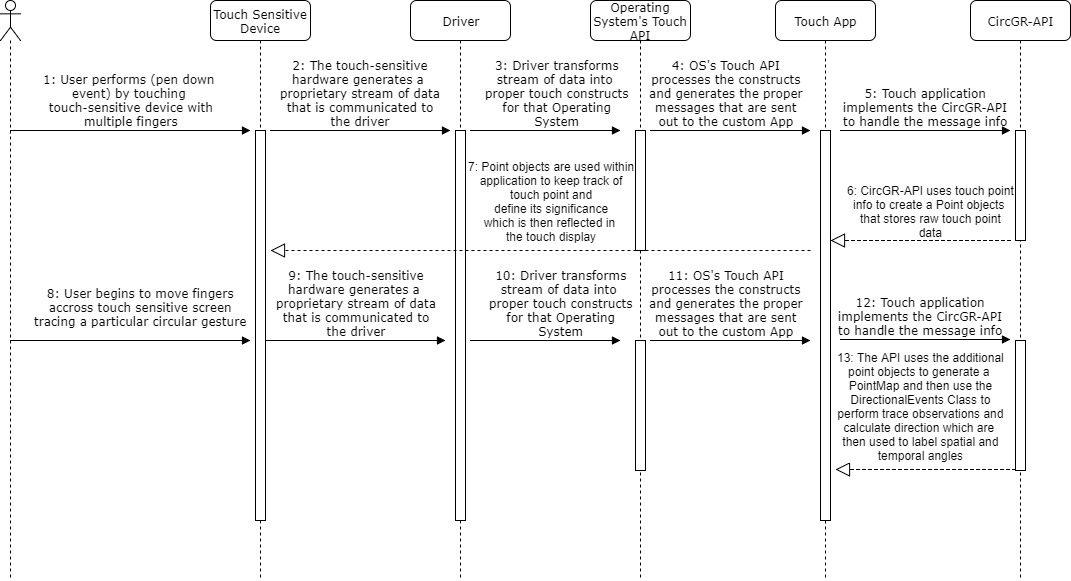
**Use Case**

* Name: Handle touch event directions
* Actor: Developer
* Preconditions: Have the CircGR-API library (all cpp and headers files inside namespace ‘GR’) and #include “DirectionalEvents.h” header within the code
* Description:
  + Developer uses the Point class to store a touch point’s properties
    - * System instantiates a Point object whenever it is called upon and provided the right parameters: X, Y, StrokeID, and Timestamp
  + Developer uses the PointMap class to store a list of Point objects inside a map whose key is the Stroke ID that is common among the Point objects in the list.
    - System instantiates PointMap which a map for traces where the key is the touch ID of trace and the value is a list of points for that trace.
  + Developer uses the Gesture class to use this Pointmap to create a gesture which is normalized with respect to scale, translated to origin, and resampled into a fixed number of SAMPLING\_RESOLUTION specified points
    - System instantiates a generic Gesture Object which consist of unprocessed traces, a name, a type, and the template name it is supposed to be a Gesture
  + Developer uses the DirectionalEvents class to handle touch event directions
    - System calculates the spatial angles of the traces performed. This is used within the CircGesture class to create a circular gesture object for later classification.

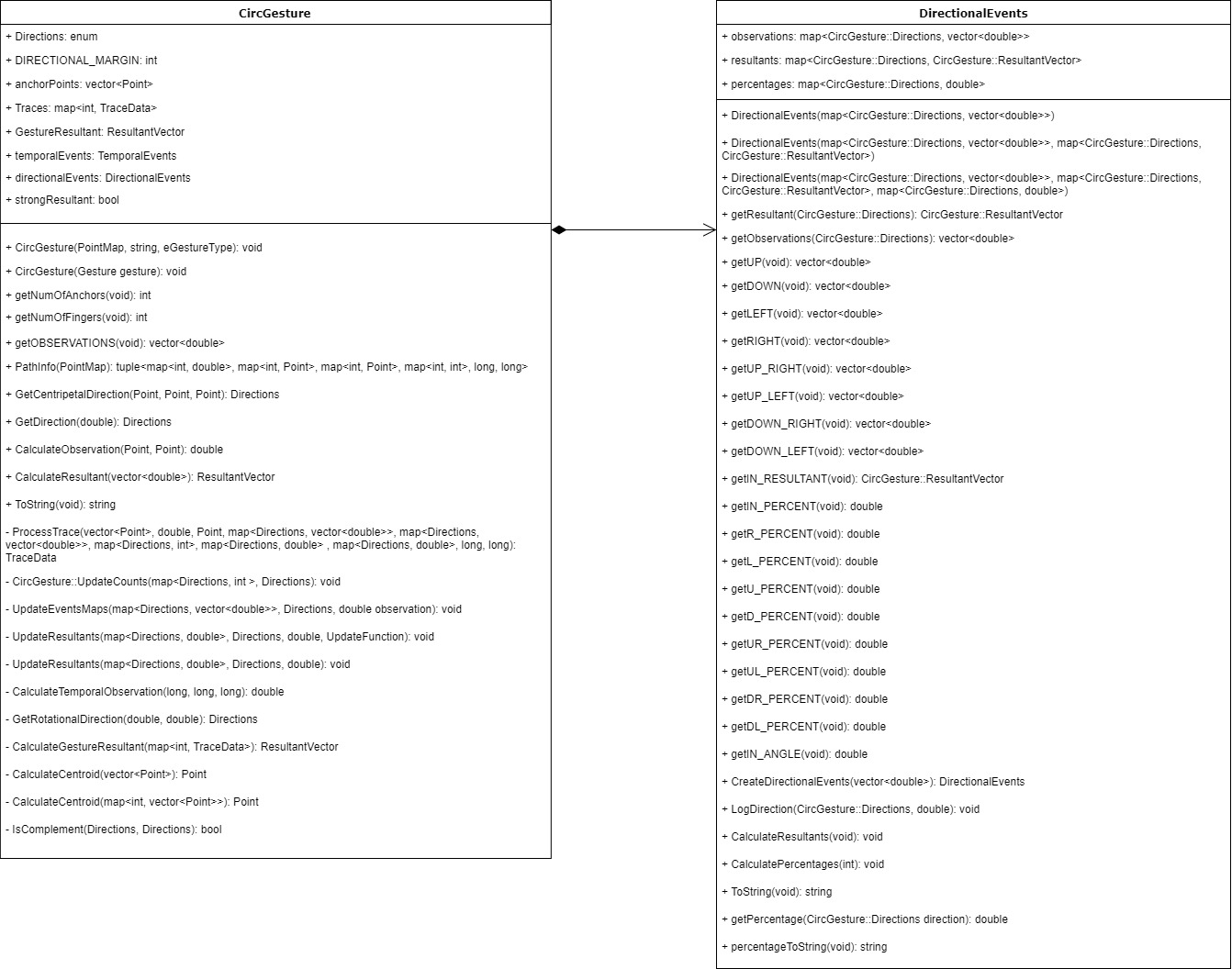
**Use Case Diagram**

****

**Sequence Diagram**



**Class Diagram**



**Unit Test**

* Test case ID: Handle\_Touch\_Event\_Directions
* Description/Summary of Test: Tests whether a touch events direction information is properly calculated and handled within the DirectionalEvents class so that it can be used within the CircGesture class to calculate other relevant information like spatial angles
* Pre-condition: Have the CircGR-API library (all cpp and headers files inside namespace ‘GR’) and #include “DirectionalEvents.h” and “CircGesture.h” header within the code
* Expected Results: Touch event directions correctly calculated
* Actual Result: same as expected results
* Status (Fail/Pass): Pass

**Integration Test**

To perform integration testing during this particular user story I first tested the interoperability between the same classes as those specified in user story #717 with the addition of the following classes: CircGesture and DirectionalEvents class

**Visual User Guide**

