*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

HoloLens:

Gaze and Gesture Recognition

User Story ID: #1188 & #1190

**Name:** *Andres Chalela*

**Team Member(s):** *Paola Jiron and Andres Chalela*

**Project:** *Augmented Reality for Computer Science Education using HoloLens*

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

**Mentor(s)**: *Francisco Ortega and Mohsen Taheri*

**Instructor**: *Masoud Sadjadi*

**Contents**

**1** **Gaze and Gesture Recognition 2-3**

**2** **Behaviour Requirements 4**

**3** **Sequence Diagrams 5-6**

**4** **Class Diagrams 7**

**5 Unit Test 8**

**6 Integration Test 8**

**7 Visual User Guide 9**

**Appendix 10**

# 

**1. Gaze and Gesture Recognition**

**US#1190:**

- Description:

*As a developer, I have to be able to implement HoloLens Gaze scripts on Unity, so that Gaze input capabilities are functional within created Unity HoloLens apps.*

Acceptance Criteria:

1. Gaze scripts have been implemented within created Unity HoloLens projects.
2. Able to use Gaze functionality within launched HoloLens app.

**US#1188:**

- Description:

*As a HoloLens Developer, I need to be able to implement HoloLens gesture scripts within Unity projects, to be able to create HoloLens apps that use hand gestures as input*

Acceptance Criteria:

1. Create Unity project that uses scripts that allow hand gestures as input on HoloLens platform
2. Use hand gestures to affect input within created Unity HoloLens app
3. Use hand gestures to successfully launch & close created HoloLens app

Use Case ID: US1190-1

1. Name: Manipulate 3D cursor with Gaze functionality
2. Actor: Player, Developer
3. Preconditions: HoloLens project has been launched and player is wearing the Hololens device
4. Postcondition: Actor manipulates cursor with Gaze and head movements
5. Flow of events:

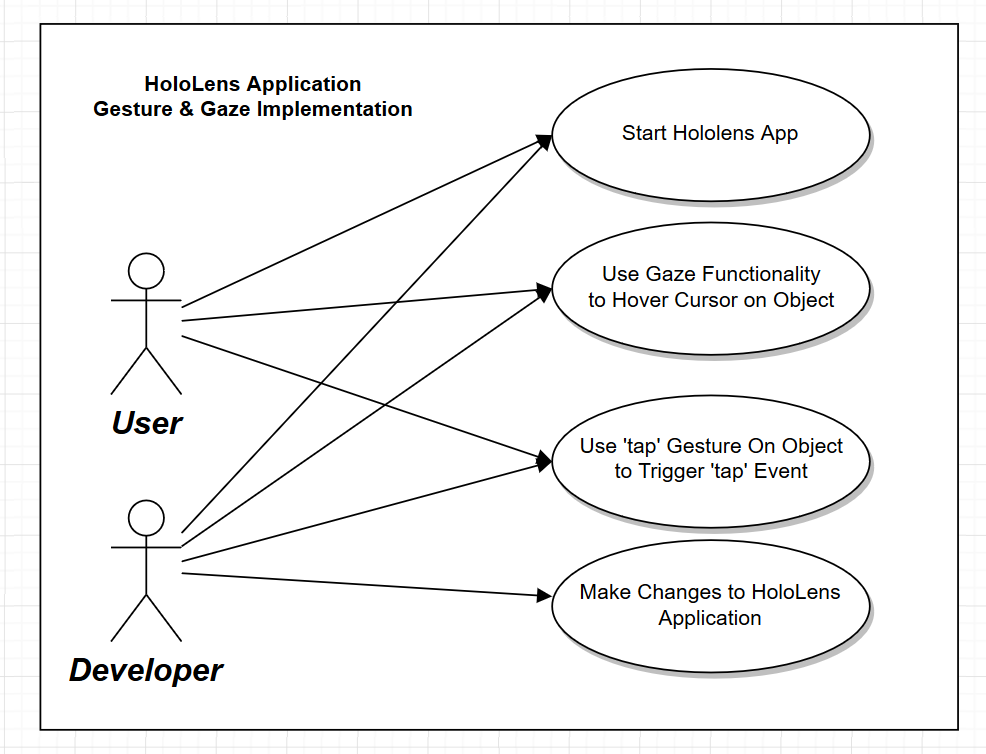
|  |  |
| --- | --- |
| 1. Actor turns head and looks around room to work with HoloLens Gaze functionality |  |
|  | 2. Actor uses Gaze and points camera pointer to 3D object. |
| 3. Gaze causes cursor to show on surface of object once camera pointer collides with object. |  |
|  | 4. Object is ready for interaction. |

Use Case ID: US1188-1

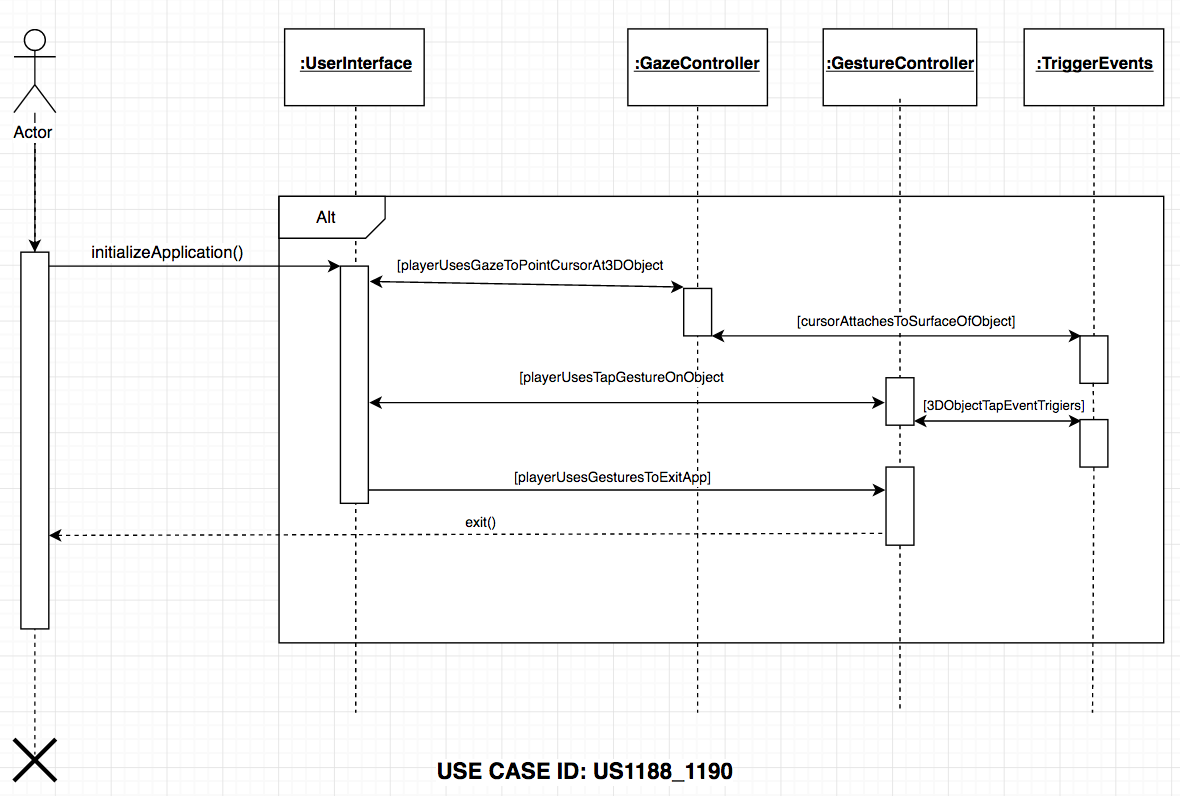
1. Name: HoloLens Hand Gesture Recognition
2. Actor: Player, Developer
3. Preconditions: HoloLens project has been launched and player is wearing the Hololens device
4. Postcondition: Actor has initialized the game application
5. Flow of events:

|  |  |
| --- | --- |
| 1. Actor turns head and looks around room to work with HoloLens Gaze functionality |  |
|  | 2. Actor uses Gaze and points camera pointer to 3D object. |
| 3. Gaze causes cursor to show on surface of object once camera pointer collides with object. |  |
|  | 4. Actor uses hand gesture input to ‘tap’ on object with finger. |

**2. Behaviour Requirements**



**3. Sequence Diagrams**



|  |
| --- |
| **Sequence Diagram** |

|  |  |
| --- | --- |
| X | Includes actors |
| X | Includes lifelines |
| X | Includes activations |
| X | Includes entity objects |
| X | Includes control objects |
| X | Includes method call in the receiving class for each received message |
| X | Includes method call in the sending class for each sent message |
| X | Includes messages |

|  |
| --- |
| **Standards** |

|  |  |
| --- | --- |
| X | Actors are a stick figure with full name |
| X | Message is a solid line with a filled arrow ahead |
| X | Return message are a dashed line with an open arrowhead |
| X | One outgoing action arrow per action box |

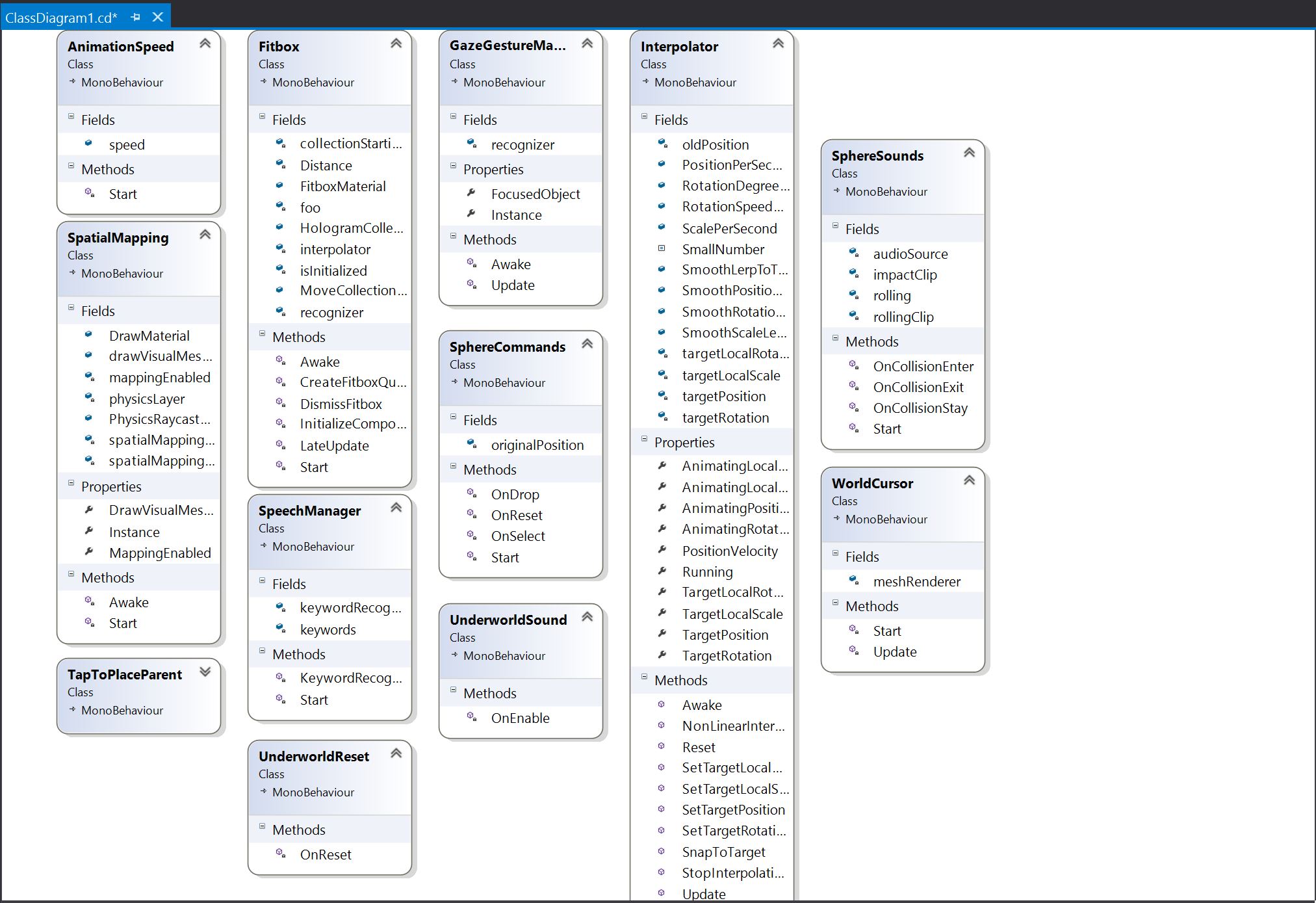
|  |
| --- |
| **Compliance of the document guidelines** |

|  |  |
| --- | --- |
| X | Is UML syntax correct? |
| X | Is there an Actor? |
| X | When there is more than one object for a class (also in other sequence diagrams) are those objects referenced with proper names? |
| X | Does the diagram cover the user case scenario? |

|  |
| --- |
| **Internal consistence of the model** |

|  |  |
| --- | --- |
| X | Is there a scenario for the sequence diagram? |
| X | Is there a documentation to which scenario belongs a sequence diagram? |
| X | Is the scenario information correctly described? |
| X | When an object of a class receives notifications, are those notifications mapped as events? |
| X | When an object of a class receives notifications, are those notifications mapped as methods defined in the class? |
| X | When an object of a class receives notifications, are those notifications mapped as actions in the State Diagrams? |
| X | Are all objects the same as shown on the class diagram? |
| X | Are there activation lifelines (rectangles) to show created and deleted objects? |
| X | Can each message be sent? |
| X | Is each class in the class diagram? |
| X | Is there a method call in the receiving class for each received message? |

**4. Class Diagram**

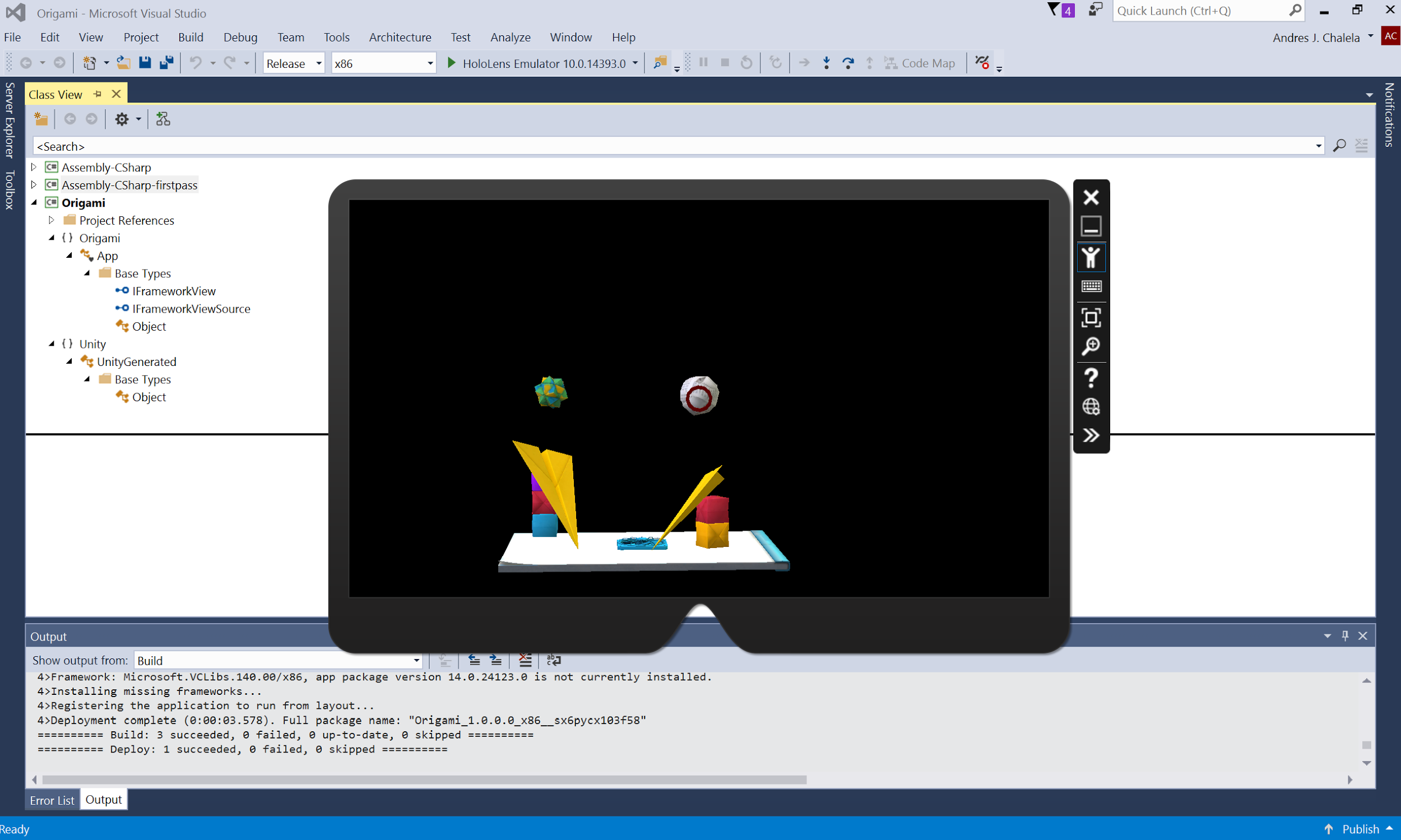


**5. Unit Test: N/A**

* Test case ID:
* Description/Summary of Test:
* Pre-condition:
* Expected Results:
* Actual Result:
* Status (Fail/Pass):

**6. Integration Test : N/A**

**7. Visual User Guide**



**Figure 1. Gaze Feature**

**APPENDIX**

*References:*

[1] "Academy." *Academy*. N.p., n.d. Web. 27 Feb. 2017. <https://developer.microsoft.com/en-us/windows/holographic/academy>.

[2] "Unity development overview." *Microsoft Developer.* Web. 27 Feb. 2017. <http://developer.microsoft.com/en-us/windows/holographic/unity\_development\_overview>.