**Software Requirements Specification**

**For**

**Augmented Reality for Hydrology**

**(ARH)**

**Version 1.0 approved**

**Faculty Advisor:**

**Dr. Elaine Y Kang**

**Team Lead:**

**Wilbert Veit**

**Team Members:**

**Christopher Hung Nguyen, Ernesto Padilla, Cuong Pham, Kaichen Zhou**

**Sponsored By:**

**Jet Propulsion Laboratory (JPL)**

**September 5, 2017**

# Table of Contents

Table of Contents.................................................................................................................<pg 2>

Revision History...................................................................................................................<pg 3>

1. Introduction...............................................................................................................<pg 4>
   1. Purpose..........................................................................................................<pg 4>
   2. Intended Audience and Reading Suggestions.............................................. <pg 4>
   3. Product Scope............................................................................................... <pg 4>
   4. Definitions, Acronyms, and Abbreviations ................................................. <pg 7>
   5. References.................................................................................................... <pg 7>
2. Overall Description...................................................................................................<pg 8>
   1. Product Perspective.......................................................................................<pg 8>
   2. Product Functions......................................................................................... <pg 8>
   3. User Classes and Characteristics.................................................................. <pg 8>
   4. Operating Environment.................................................................................<pg 8>
   5. Design and Implementation Constraints.......................................................<pg 8>
   6. User Documentation.....................................................................................<pg 9>
   7. Assumptions and Dependencies...................................................................<pg 9>
   8. Apportioning of Requirements.....................................................................<pg 9>
3. External Interface Requirements..............................................................................<pg 10>
   1. User Interfaces..............................................................................................<pg 10>
   2. Hardware Interfaces.....................................................................................<pg 12>
   3. Software Interfaces.......................................................................................<pg 12>
   4. Communications Interfaces..........................................................................<pg 12
4. Requirements Specification......................................................................................<pg 13>
   1. Functional Requirements..............................................................................<pg 13>
   2. External Interface Requirements..................................................................<pg 17>
   3. Logical Database Requirements...................................................................<pg 17>
   4. Design Constraints.......................................................................................<pg 18>
5. Other Nonfunctional Requirements.........................................................................<pg 19>
   1. Performance Requirements..........................................................................<pg 19>
   2. Safety Requirements....................................................................................<pg 19>
   3. Security Requirements.................................................................................<pg 19>
   4. Software Quality Attributes..........................................................................<pg 19>
   5. Business Rules..............................................................................................<pg 20>
6. Other Requirements..................................................................................................<pg 21>

Appendix A: Glossary..........................................................................................................<pg 22>

Appendix B: Analysis Models.............................................................................................<pg 24>

Appendix C: To Be Determined List...................................................................................<pg 25>

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| First Draft | 11/21/2017 | First initial draft of the document | 1.0 |
| Second Draft | 12/08/2017 | Second draft of the document | 2.0 |
|  |  |  |  |
|  |  |  |  |

# 

# 1. Introduction

## 1.1 Purpose

This document has three major purposes:

1. Describe the Augmented Reality for Hydrology project, giving enough information that the reader will understand what it accomplishes, how people will interact with it, and how it will interact with other systems.
2. Describe any concepts that are necessary in understanding Augmented Reality for Hydrology requirements and operation.
3. Describe the requirements that will be used to determine completion of this product.

## 1.2 Intended Audience and Reading Suggestions

While this document is written with the intention of being accessible and useful to anyone who needs to learn about the purpose, operation, and requirements of the Augmented Reality for Hydrology project, this document is primarily intended for:

* Customers – To ensure that the product being developed is what the customer wanted built in the first place.
* Developers – To provide guidance on which systems should be built, and the required behavior of these systems.
* Testers – Describes the required behavior of the product so that testers can ensure such requirements are adhered to.
* Managers and Planners - Knowing the parts of the project that need to be built will allow

## 1.3 Product Scope

The Jet Propulsion Laboratory (JPL) is world famous for its work in space exploration. However, JPL also has many projects for advancing the Earth Sciences. One JPL project called Watertrek, provides online access to several types of Earth hydrology data. JPL has commissioned the Augmented Reality for Hydrology project with two main objectives:

* Create an application, using Augmented Reality Techniques, for visualizing the Watertrek data in the field.
* Create a framework that consists of the generally useful components created during the application’s development.

These two parts of the Augmented Reality for Hydrology project will be referred to in this document simply as “the Application”, and “the Framework”.

The Application will use Augmented Reality (AR) techniques as the primary method of data visualization. In this project, AR means displaying the output of a connected camera in a view or window, with additional text and graphics drawn on top of the camera view, thus augmenting the reality seen by the camera and user. As the device is moved, the camera view and the associated graphics will move as well, making it appear as if the graphics are actually a part of the real world. As an example, figure 1.3.1 shows a possible AR view that may be used in the application. The application will support Android devices that contain a camera and sensors for detecting movement.

The framework will provide generally useful, re-usable components that are used in the application development. At a high level, this includes accessing sensors, 3D rendering, camera UI component on which to draw, vector math functions, geo math functions, and structures for managing data from Watertrek. Making these components available in a separate framework will allow other developers to make similar applications, or simply to use isolated components from the framework that are useful to them.

For more detailed information about the application and framework, see section 2.

**Figure 1.3.1** One possible screen from the 

application, demonstrates AR by

displaying icons on top of the camera view

## 

**1.4 Definitions, Acronyms, and Abbreviations**

Please see Appendix A: Glossary for a list of Definitions, Acronyms, and Abbreviations.

## 1.5 References

Provided below are links to other resources or documents that pertain to the development of the product.

|  |
| --- |
| Augmented Reality for Hydrology wiki |
| <http://csns.calstatela.edu/wiki/content/JPL-1/Hydrology> |

|  |
| --- |
| PowerPoint presentation introducing the project and project concepts |
| <https://drive.google.com/file/d/1HrNhtybcGaqYVTzKRXVOHrPKsoKJXl2U/view?usp=sharing> |

# 2. Overall Description

## 2.1 Product Perspective

Both the application and the framework are part of an effort to create a more interactive product for students and those who are curious about the water flow of our mother Earth. The use of this product will be designed to allow users to visualize scientific data while explore the surrounding area.

## 2.2 Product Functions

* Query API Function
  + Will Access JPL Hydrology Data.
* Database storage Function
  + Will process queried data into local database. Will also propagate data in application.

## 2.3 User Classes and Characteristics

2.3.1 Application users: application users will use the app to visualize available scientific data in their surroundings.

2.3.2 Developers (Framework users): developers will utilize the framework in their apps in order to add augmented reality visualization functionality to their own sets of data.

## 2.4 Operating Environment

2.4.1. The application will run on Android devices with API 15 or higher.

**2.5 Design and Implementation Constraints**

The framework must be able to support all devices that have the necessary sensors and processing performance to handle an app (not limited to just the app that we shall deliver) that utilizes the framework.

The app that we shall deliver must utilize all functionalities provided by the framework, and likewise, the framework must provide all of the functionalities that the app requires aside from user interface.

Both the app and the framework, despite having an online component, must support some offline functionalities provided via caching a certain amount of online data, as specified by the user.

## 2.6 User Documentation

A manual will be provided to accompany the framework and its functionality in an easy to read pdf document. A short tutorial will also be provided for the application component in order to facilitate easy of use for all users also in a pdf document.

## 2.7 Assumptions and Dependencies

2.7.1 Users are expected to have consistent and stable internet connections to use this application for online mode.

2.7.2 Android Devices are assumed to have a working camera and location sensor.

2.7.3 Android Devices are assumed to have Android API level 15 and up.

2.7.4 Users are expected to have digital certificate signed by JPL installed on their devices.

## 2.8 Apportioning of Requirements

2.8.1 The framework shall support building similar apps on iOS, Android, and DJI drone platforms

2.8.2 The framework and the app shall provide the same functionality and interface across all supported platforms

# 

# 3. External Interface Requirements

**3.1 User Interfaces**

3.1.1. The user interface shall be prompted with the loading screen upon executed the application.

3.1.2. The preliminary load shall process the initial graphics and fetch the required data from the JPL databases.

3.1.3. Subsequently, the app shall immediately gain access to the device's camera and display the user interface elements.

3.1.4. User interface elements in the main screen shall include icons (billboards) of points of interest and the settings button (Figure 3.1.1).

3.1.5 The preferences screen shall include filter billboards options, toggle measurement system, and range adjuster (Figure 3.1.2).

Figure 3.1.1

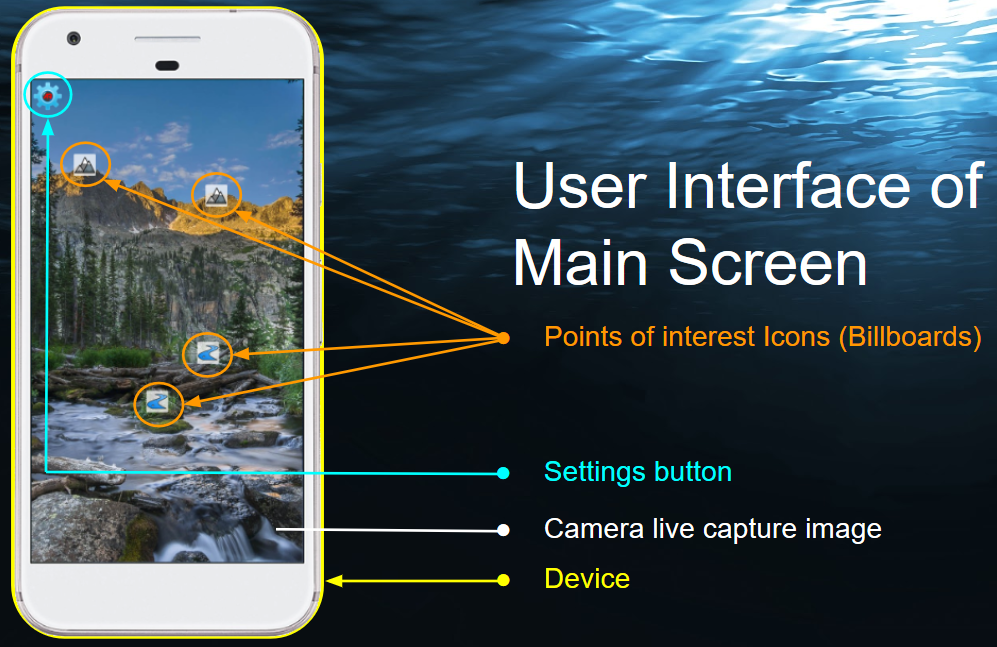
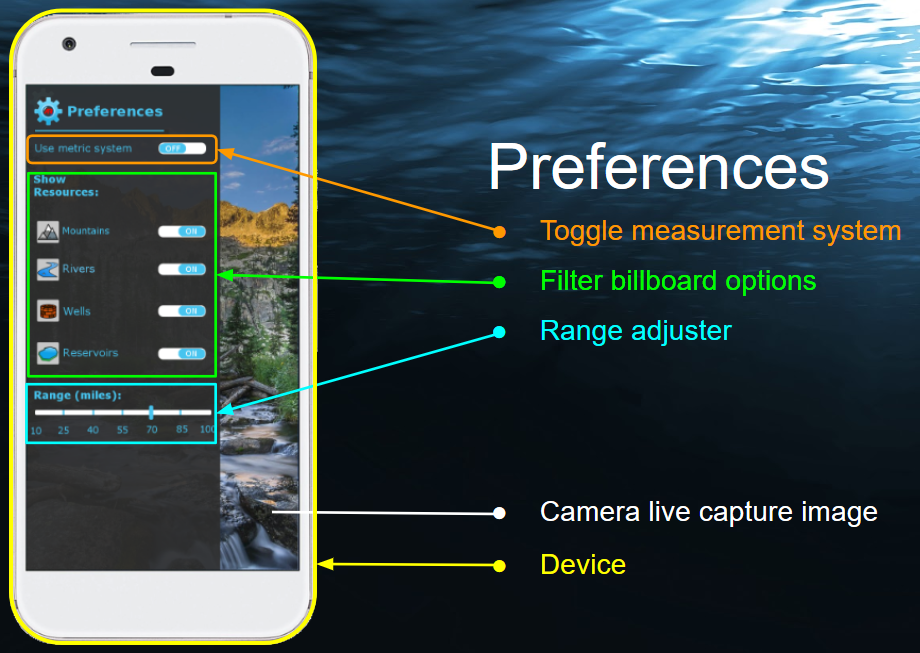


Figure 3.1.2



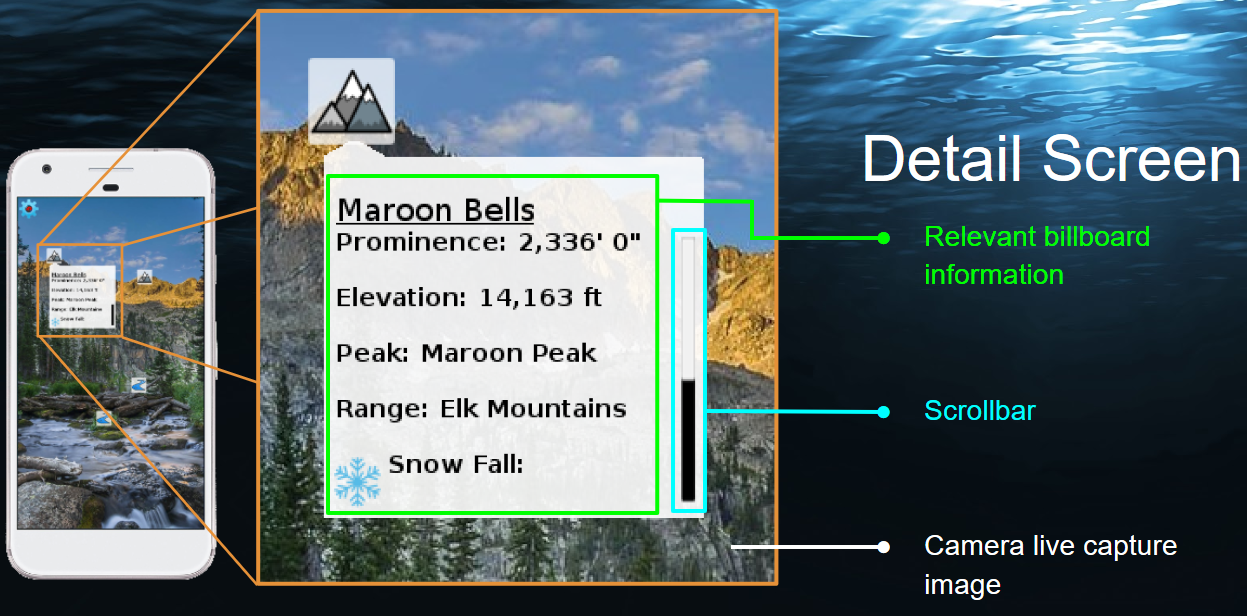
3.1.6. The interface shall allow users to select billboards and display its detail screen (Figure 3.1.3).

3.1.7. The detail screen shall contain relevant information of the billboard from the database.

3.1.8. It shall adjust its size depending on how many other billboards are active on the device's screen to avoid interfering with other billboards.

3.1.8 A scrollbar shall appear on the right of minimized detail screens when not all of its information can be displayed at once.

Figure 3.1.3 (User Interface 3.1.6 - 3.1.8)



## 3.2 Hardware Interfaces

3.2.1 Android devices (with API level 15 & up)

3.2.1 Android devices with camera, location sensor and magnetic sensor

3.2.3 Android devices with internet access

## 3.3 Software Interfaces

3.3.1 Android API level 15 & above

3.3.2 OpenGL ES 2

3.3.3 JPL Water Trek API

3.3.4 SQLite

## 3.4 Communications Interfaces

3.4.1 HTTP GET

3.4.2 HTTPS GET

3.4.3 Server side events

# 4. Requirements Specification

## 4.1 Functional Requirements

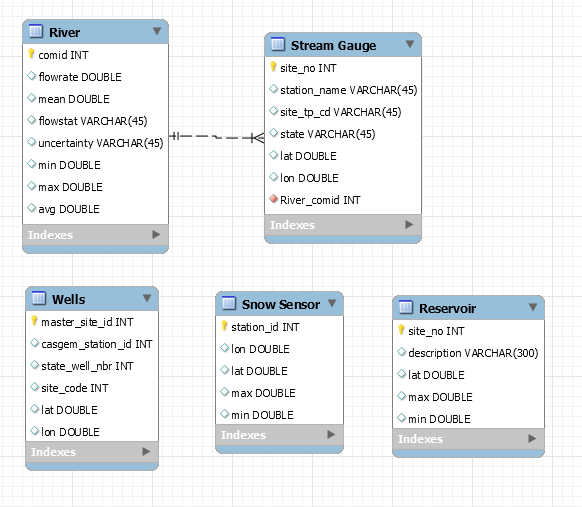
1. Application Requirements
   1. The application shall maintain a data structure that holds Points of Interest (POIs) as follows:
      1. The data structure shall be loaded with POIs from Watertrek upon application startup, and periodically updated as needed to meet the additional POI data structure requirements.
      2. At a minimum, the data structure shall contain all Watertrek POIs that are within the load-radius of the device’s current location.
   2. The application shall provide a screen, called the “Billboard Screen” that meets the following sub-requirements:
      1. The Billboard Screen shall display the device’s front facing camera output as its background.
      2. If the device does not have a front facing camera, or if permission for the camera is not granted, the Billboard Screen shall display a white background instead of the camera output.
      3. On top of the camera output (or white background if camera is unavailable), the Billboard Screen shall render billboard objects on top of the camera output to represent currently active POIs that are within the view of the camera.
      4. A POI shall be considered active if it is loaded in the POI data structure, is located within the user specified view-radius of the device’s current location, and user controlled filters do not prevent that type of POI from being displayed.
      5. The POI billboards on the billboard screen shall be rendered in the appropriate position on the screen, such that the pixels of each billboard appear to be located where the physical POI would be located in the camera picture.
   3. The application shall provide a method by which the user can set the view-radius.
   4. The application shall provide a filter capability, by which the user can determine which types of POIs will be displayed on the Billboard Screen.
   5. The application shall provide a method by which the user can select a billboard which is displayed on the Billboard Screen.
   6. Upon selecting a POI’s billboard on the Billboard Screen, the Application shall display a details screen, which will display the detailed information about the POI downloaded from Watertrek.
2. Framework Requirements
   1. The Framework shall provide methods for accessing the following location and orientation information for the device:
      1. Latitude, Longitude, Altitude of the device
      2. Gravity vector which represents a vector from the device to the center of the earth in device coordinate system.
      3. Pitch, Yaw, and Roll angles for the device
      4. Compass Bearing and Elevation Angle for the device.
      5. Front vector, Up Vector, Right vector for device in world coordinate system.
   2. The framework shall the following 3D Rendering abstractions:
      1. The framework shall provide a drawing surface class, on which camera preview, 3D objects and 2D objects can be rendered
      2. The framework shall provide a camera abstraction.
         1. The camera shall allow setting the location, angle, and orientation vectors.
         2. The camera shall provide a view matrix for 3D rendering.
      3. The framework shall provide an abstraction for a 3D drawable entity.
         1. The entity abstraction shall hold information about location, orientation and scale of the entity.
         2. The entity abstraction shall hold the following information needed for rendering: vertices, normal vectors, shader, texture and texture coordinates.
         3. The entity abstraction shall provide a drawing function
      4. The framework shall provide an abstraction for displaying Billboards, which represent a 3D sign that always faces the camera.
         1. The framework shall provide a method that allows setting texture, position and scale of each Billboard.
      5. The framework shall provide a method for tracing from screen pixel to an entity or Billboard in the 3D world allowing the user to select the object by touch/click
   3. Math
      1. The framework shall provide functions for the following vector operations:
         1. Add
         2. Subtract
         3. Cross Product
         4. Dot Product
         5. Calculate device pitch/yaw/roll
         6. Calculate device up vector/front vector/right vector
         7. Calculate compass bearing and elevation angle
         8. Convert vector from world coordinates to device coordinate (and vice versa)
      2. The framework shall provide functions for the following geographical math operations:
         1. Convert between lat/lon/altitude and an absolute x/y/z;
         2. Given two coordinates calculate compass bearing and elevation angle from the first coordinate to the second.
   4. Point of Interest (POI) management
      1. The framework shall provide an abstraction of Points of Interest (POIs) that includes at minimum a type identifier, geographic location (latitude, longitude, altitude), and name.
      2. The framework shall provide an abstraction of a POI data structure that allows adding, removing, filtering, saving, loading, iterating through POIs efficiently
   5. Watertrek Hydrology access
      1. The framework shall provide a method to securely query Water Trek with various parameters for points of interest.
      2. The framework shall provide methods for storing and locally querying data.
      3. Provide method of programmatically detecting or being notified of connection error, data compatibility error, or other type of error associated with Water Trek access.

## 4.2 External Interface Requirements

## This project does not have requirements placing restrictions on the external interfaces

## 4.3 Logical Database Requirements

An internal database will be utilized to cache queried data. As such, the information to be used pertains to the Hydrology data requested by the framework and application. The information will be kept in memory and will be used as long as it is relevant to the user’s location. Should there be out of date data, the functions will query or access the relevant data. Current entities are reserved but not limited to: Wells, Rivers, Rapids, Snow Sensor, Stream Gauge.



## 4.4 Design Constraints

Specify design constraints that can be imposed by other standards, hardware limitations, etc. This should be a more technical description of the overview given in section 2.5.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

The application should receive and output desired motion immediately. Input device should maintain its fluid response throughout and saving and loading of setting should take less than 1 second. Updates received from the server should take no longer than 1 second to appear on screen.

## 5.2 Safety Requirements

The possibility of harm, loss, or damage from the use of the application are highly improbable being that the product is intended for use in what is assumed to be controlled environment. Additionally, the intended use of the product does not present any foreseeable scenarios where a user would have to encounter harm of any kind in order to fulfill the maximum utility of the product.

## 5.3 Security Requirements

The Application will only require a unique digital certificate to determine the appropriate route for a connection to be established and maintained. No user information will be collected and used.

## 5.4 Software Quality Attributes

The application shall be developed and tested to ensure that there is minimal latency in

the background processes that occur during the use of the application. The application

shall be designed to ensure that the user is able to quickly acclimate to the user

interface and is able to quickly learn how to navigate through the application be able to

use all the functions of the application with ease.

## 5.5 Business Rules

The application shall not be used as an enterprise application to assist in the operations of any organization involved in business, government, education, etc. However later implementations may allow the application to be used as an educational tool and as an application used for research. The domain of use for this application shall fall in the realm of consumer leisure and entertainment as it will be used as an application by users, which are assumed to have little to no scientific or technical knowledge, to interact with JPL’s Water Trek portals.

# 

# 

# 6. Other Requirements

The reuse objective for this project aims to create a fully functional framework that other developers will be able to easily modify and implement for their own uses. As such, the objective is to create easy to use abstract, and malleable classes and structures to handle the heavy lifting for the developer and only have them worry about designing their application using the tools and framework we have provided.

# Appendix A: Glossary

**Augmented Reality (AR)** - Various techniques for providing computer generated information on top of a person’s perception of the real world. Usually, and in the case of our application/framework, this is accomplished by drawing computer text and graphics on top of a camera view.

**Augmented Reality for Hydrology project** - The software project described in this document, which includes both an application, referred to as ‘the application’, and a framework, referred to as ‘the framework’.

**The Application** - One of the two major software deliverables from the Augmented Reality for Hydrology Project. It’s purpose is to allow visualization of JPL’s Watertrek data, using Augmented Reality techniques, on a mobile device.

**The Framework** - One of the two major software deliverables from the Augmented Reality for Hydrology Project. It consists of generally useful software components that were created to support development of The Application portion of the project. These components are packaged in a reusable format, such that other developers can use these components in their own apps. The framework functionality includes Sensor, 3D rendering, UI components, and POI management functionality.

**Hydrology** - The study of earth’s water, where it is located, and how it moves.

**Watertrek** - A database and its associated REST API, provided by JPL, which provides, names, locations and various data for hydrology data.

**Application Programming Interface (API)** - A defined interface for accessing a software service. Usually an API consists of class names, function names, or HTTP request url formats that have defined behavior.

**REST API** - An API that uses HTTP or HTTPS requests to GET, PUT, POST and DELETE data. Uses a GET call to request exposed resources or data.

**Endpoint -** The endpoint is a connection point where HTML files or active server pages are exposed. Endpoints provide information needed to address a Web service.

**Billboard** - A rectangle object that is drawn in a 3D virtual world that has graphics drawn on it, always faces the user and functions as a sign. The size, location in the 3D world, and graphics to be displayed can all be specified for each Billboard object.

**Point-Of-Interest (POI)** - POI objects are a method of representing real world places or objects the Augmented Reality for Hydrology project. Each POI has a name, a type, and a location (latitude, longitude, altitude). In addition, the various types of POIs can have their own data fields. For example, a river segment POI object will have data representing the current and historical water flow measurements, and the physical boundaries of the river segment.

# Appendix B: Analysis Models

No analysis models are provided as part of this SRS

# Appendix C: To Be Determined List

There are no TBD items in this SRS.