|  |  |
| --- | --- |
|  |  |
|  | ReaLearn Requirements Document  Version: 2.1  Date Created: 08/07/2019  Name: Euan Macqueen Project: A Virtual Reality eLearning Authoring Tool Prototype |

|  |  |
| --- | --- |
|  |  |

# Table of Contents

[1. Table of Contents 2](#_Toc18626991)

[2. Overview 3](#_Toc18626992)

[2.1. Purpose of this document 3](#_Toc18626993)

[2.2. Project Scope 3](#_Toc18626994)

[2.3. Assumptions/Constraints 3](#_Toc18626995)

[2.4. Hardware/Software requirements 3](#_Toc18626996)

[3. Requirements 4](#_Toc18626997)

[3.1. Non-Functional Requirements 4](#_Toc18626998)

[3.1.1. Scalability 4](#_Toc18626999)

[3.1.2. Maintainability 4](#_Toc18627000)

[3.1.3. Security 4](#_Toc18627001)

[3.1.4. Usability 4](#_Toc18627002)

[3.1.5. Interoperability 4](#_Toc18627003)

[3.1.6. Manageability 4](#_Toc18627004)

[4. Prototype Design 5](#_Toc18627005)

[4.1. WireFrames 5](#_Toc18627006)

[4.1.1. Scene Editor 5](#_Toc18627007)

[4.1.2. Scene Editor Assets 5](#_Toc18627008)

[4.2. User Stories 7](#_Toc18627009)

[4.2.1. Actors 7](#_Toc18627010)

[4.3. MoSCoW statements 16](#_Toc18627011)

[4.4. Iteration Plan 16](#_Toc18627012)

[4.5. Schedule of Iteration Workflows 16](#_Toc18627013)

[4.6. Iteration Plan with Story Points 17](#_Toc18627014)

[4.6.1. Sprint One Tasks 17](#_Toc18627015)

[4.6.2. Sprint Two Tasks 18](#_Toc18627016)

[4.6.3. Sprint Three Tasks 18](#_Toc18627017)

# Overview

## Purpose of this document

This document proposes an abstraction of the requirements for the application ReaLearn. This document will focus primarily on the non-functional requirements as well as usability, maintainability and scalability.

## Project Scope

**Scope:**

The scope of this project is to provide a virtual reality authoring tool to offer the creation of virtual reality eLearning content. This project aims to simplify VR elearning content creation via a code-free solution.

The project will offer:

1. Allow Secure User Authentication.
2. Access to content relative to user privileges determined by role
3. Offer User Management with regards to registration, updating, deletion and adding new users.
4. Course Management with regards to adding, updating, deletion.
5. Scene Management with regards to organization, adding, updating, deletion.
6. Asset Management with regards to usability, organization, adding, updating, deletion.
7. Universal design for inexpert users to offer increased usability.
8. Scene Editing (MINIMUM REQUIREMENTS):
   1. Add Background, Text, Image, Video, Audio and Questions to scene.
   2. The application must offer editing capabilities of object position, rotation, scale and colour.
   3. Hotspot objects must allow the following events:
      1. Show object on click or view.
      2. Hide object on click or view
      3. Play media
      4. Stop media
      5. Link to Scene
9. Assign courses to standard users.
10. Export to appropriate standard: xAPI/SCORM.

## Assumptions/Constraints

The assumptions/constraints are listed below;

* Assumption: users will access through desktop/laptop and not mobile.
* Constraint: An internet connection is required
* Constraint: Availability of application is determined by A-Frame VR framework’s availability
* Constraint: Time for development limited to 8 weeks.
* Constraint: Due to budget limitations, all assets/elements/frameworks adopted throughout application will cost nothing

## Hardware/Software requirements

The hardware/software requirements are listed below

* Content delivered will require virtual reality hardware and software.
* The virtual reality hardware catered for from exported output are: oculus go, mobile, htc vive.
  + Additional Requirement: internet connection.

# Requirements

## Non-Functional Requirements

### Scalability

Database design must be in at least the third normalised form to reduce the number of update, insert and deletion anomalies. Furthermore, database queries must perform no lower than O(n).

### Maintainability

With the employed virtual reality framework, A-Frame, at version 0.9.2, the development of A-Frame must be monitored. All new A-Frame version releases must be implemented ASAP to avoid lack of reliability as well as large backlogs of maintenance being required. Additionally, code must be thoroughly vetted to ensure it follows the DRY policy (Don’t Repeat Yourself). This will allow for easier maintenance and upgrades. A suitable and well layered system architecture, following at minimum the MVC design pattern, will allow for the appropriate separation of concerns to offer loose coupling and high cohesion. This will provide improvements to system maintainability.

### Security

Asp.net Core identity framework will be employed to serve as security for user accessibility. Areas prone to HTML/SQL injection of the application will be tested thoroughly before deployment. Additionally, to prevent cross site forgery, ASP.net’s anti forgery toolkit will be employed through all forms.

### Usability

Key areas of usability to focus on are site navigation and scene editor.

The scene editor must follow a minimalist but informative design whilst maintaining all core functionality. This will prevent information density and will offer clear functionality to the user. The application must be easy to learn with a user-friendly interface. Errors must be clearly displayed to the user. Site navigation must be clear and not restrict user.

### Interoperability

Virtual content must offer controls to various VR enabled devices; namely, mobile, oculus go, HTC Vive. Exported content must be interoperable between learning management systems. This will be done through the user of data standards regarding eLearning content: at minimum, xAPI/SCORM.

### Manageability

User privileges determined by role must be implemented. This will allow admin users complete control over insert, update and delete controls over courses and standard users.

# Prototype Design

## WireFrames

### Scene Editor

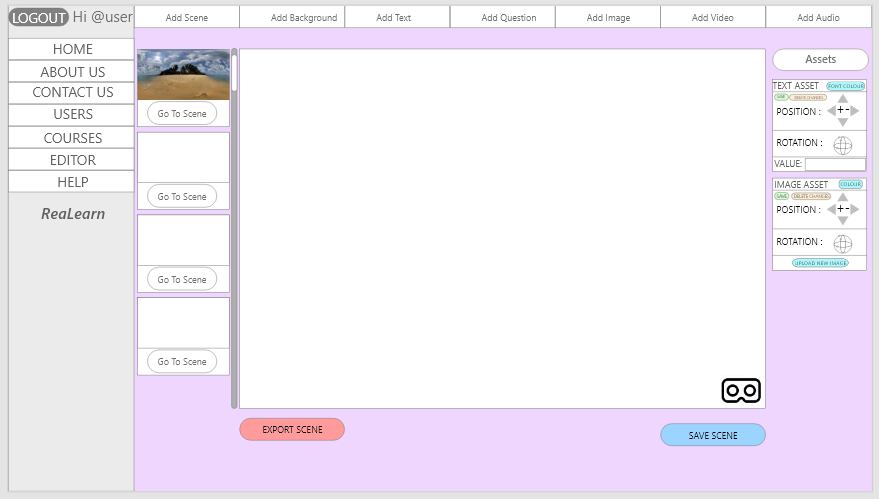


Figure - Scene Editor V.1

### Scene Editor Assets

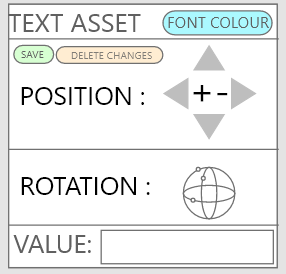


Figure - Text Asset Editor

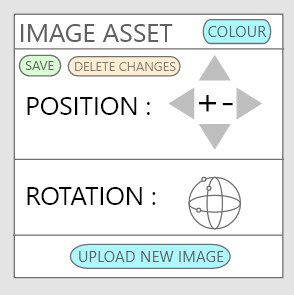


Figure - Image Asset Editor

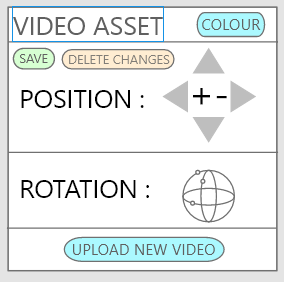


Figure - Video Asset Editor

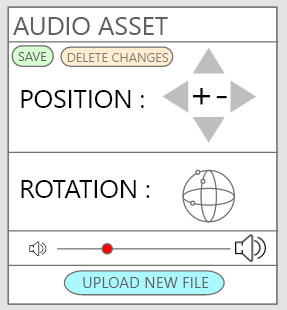


Figure - Audio Asset Editor

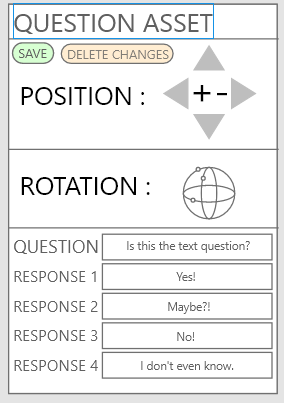


Figure - Question Asset Editor

## User Stories

The front of the user stories provides the user story, the priority level, notes for intended use and the story point.

The back of the user story provides the acceptance test to ensure the functionality works as expected.

### Actors

* General User
* Standard User
* Administrator
* Scene Designer
* System

|  |
| --- |
| FRONT – Register Account  As a new user, I must be able to register my account, so I am saved on the system and can access my account   * Use identity framework   Must. 1. |
| BACK   * Register Details in register page * Log in with details. * Enter email with wrong password to make sure password registration works. * View account page – are the registered details there? * Try and register the same email with another account * Try and register the same username with another account |

|  |
| --- |
| FRONT – Log In  As a user, i must be able to log in securely, so only I can access my account   * Use identity framework   Must. 0.5 |
| BACK   * Log in to account. * View printed logs of user activity. * View Account page – are the user’s details there? |

|  |
| --- |
| FRONT – Log out  As a user, I must be able to log out securely, so only I can access my account   * Use identity framework   Must. 0.5. |
| BACK   * Log in to account. * Log out of account. * View printed logs of user activity. * Try and access content only accessible to users that are logged in |

|  |
| --- |
| FRONT – Forget Password  As a forgetful user, I would like to be able to retrieve my password, so i am able to access my account if i forget my password  Would. 0.5. |
| BACK   * Select Forgot Password * Enter email address |

|  |
| --- |
| FRONT – Add User  As an admin user, I must be able to add standard users, so they can access their account.  Must. 1. |
| BACK   * Log in as admin – go to add user page * Enter user to be added details * Log out and log in as new user * Are admin pages available? If so, user was added incorrectly * Go to Account page – are the new users details visible? |

|  |
| --- |
| FRONT – Add Course  As an admin user, I must be able to add courses, so i can continue to create content.  Must. 0.5. |
| BACK   * Log in as admin – go to add course page * Enter details for courses to be added * View database to see if course has been added * If scene editor page is working, try and access course |

|  |
| --- |
| FRONT – Add Scene  As a scene designer, I must be able to add a new scene, so i can continue to create more content.  Must. 0.5. |
| BACK   * Navigate to scene editor page * Select add scene * Enter details for new scene * Does new scene appear? * Select “Go to scene” – are you redirected to the new scene? |

|  |
| --- |
| FRONT – Delete Course  As an admin user, I must be able to delete courses, so i am able to manage them.  Must. 0.5. |
| BACK   * In view courses page * Select “Delete Course” * Does course disappear from page * Check database to make sure course is removed. |

|  |
| --- |
| FRONT – Delete Scene  As a scene designer, I must be able to delete a scene, so i am able to manage my content.  Must. 0.5. |
| BACK   * In Scene Editor * Select “Delete Scene” * Does scene disappear from course * Check database to make sure scene is removed. |

|  |
| --- |
| FRONT – Delete User As an admin user, I must be able to delete standard users, so i am able to manage them.  Must. 0.5. |
| BACK   * In View Users page * Select “Delete User” * Does user disappear from page? * Check database to make sure user is removed. * Try log in with users details – if unable, user is removed. |

|  |
| --- |
| FRONT – Edit User Details as admin  As an admin user, I must be able to edit standard users details, so I am able to update their details.  Must. 1. |
| BACK   * Navigate to View Users page * Select “Edit” on user to be edited * Enter new details for selected user * Select submit * Refresh page – are new details displayed? * Check database to make sure. |

|  |
| --- |
| FRONT – Edit own account details  As a standard user, I must be able to edit my own details, so i can update them.  Must. 2. |
| BACK   * Navigate to account page * Enter new details * Select submit * Refresh page – are new details displayed? * Check database to make sure. |

|  |
| --- |
| FRONT – Edit Course details  As an admin user, I must be able to edit courses, so I am able to change name and description.  Must. 1. |
| BACK   * Navigate to edit courses page * Select “Edit” on course to be edited * Enter new details * Select submit * Refresh page – are new details displayed? * Check database to make sure. |

|  |
| --- |
| FRONT – Edit Scene details  As an admin user, I must be able to edit scenes, so I am able to change scene title and description.  Must. 1. |
| BACK   * Navigate to edit scenes page * Select “Edit” on scene to be edited * Enter new details * Select submit * Refresh page – are new details displayed? * Check database to make sure. |

|  |
| --- |
| FRONT – Add Text Asset to VR Scene As a scene designer, I must be able to add text object to VR scene, so it can be seen in content.  Must. 2. |
| BACK   * In Scene Editor * Select “Add Text” * Enter Text object details * Select save * Does the asset appear in the scene? * Check database to make sure asset is stored under the scene. |

|  |
| --- |
| FRONT – Add Audio Asset to VR Scene  As a scene designer, I must be able to add audio object, so it can be heard in content.  Must. 2. |
| BACK   * In Scene Editor * Select “Add Audio” * Upload audio from folder * Select save * Does the audio icon appear in the scene? * Check database to make sure asset is stored under the scene. * \* If hotspot asset is in scene, interact with hotspot to play media |

|  |
| --- |
| FRONT – Add HotSpot Asset to VR Scene  As a scene designer, I must be able to add hotspot object, so users can initiate events within a scene.  Must. 4. |
| BACK   * In Scene Editor * Select “Add HotSpot” * Select event initiated from interacting with hotspot * Link hotspot to object within scene (if required) * Select save * Does the asset appear in the scene? * Check database to make sure asset is stored under the scene. * Does the event interact with the linked object? |

|  |
| --- |
| FRONT – Add Image Asset to VR Scene As a scene designer, I must be able to add image object to scene, so it can be seen in content.  Must. 2. |
| BACK   * In Scene Editor * Select “Add Image” * Upload Image from folder * Select save * Does the Image appear in the scene? * Check database to make sure asset is stored under the scene. |

|  |
| --- |
| FRONT – Add Question Asset to VR Scene  As a scene designer, I must be able to add question, so it can be interacted within the scene.  Must. 4. |
| BACK   * In Scene Editor * Select “Add Question” * Enter Question and up to 4 responses * Select save * Does the Question card appear in the scene? * Check database to make sure asset is stored under the scene. * Can you answer the question when previewing the scene? |

|  |
| --- |
| FRONT – Add Video Asset to VR Scene  As a scene designer, I must be able to add a video from a file, so i am able to use my own videos.  Must. 2. |
| BACK   * In Scene Editor * Select “Add Video” * Upload Video from folder * Select save * Does the Video appear in the scene? * Check database to make sure asset is stored under the scene. * \* If hotspot asset is in scene, interact with hotspot to play media |

|  |
| --- |
| FRONT – Add Background Image to VR Scene  As a scene designer, I must be able to add an image background, so viewers can experience 360 degree scene.  Must. 2. |
| BACK   * In Scene Editor * Select “Add Background” * Upload Image from folder * Select save * Does the background appear in the scene? * Check database to make sure background is stored under the scene. |

|  |
| --- |
| FRONT – Add Video Background to VR Scene  As a scene designer, I must be able to add a video background, so viewers can experience a 360 degree video scene.  Would. 2. |
| BACK   * In Scene Editor * Select “Add Background” * Upload Video from folder * Select save * Does the Video Background appear in the scene? * Check database to make sure asset is stored under the scene. * \* Play background media with hotspot object event |

|  |
| --- |
| FRONT – Delete Asset  As a scene designer, I must be able to delete any object asset (text, image, video, audio, question, hotspot), so i am able to manage my scene.  Must. 1. |
| BACK   * In scene editor * Select “Delete Asset” * Does asset disappear from scene? * Refresh page to make sure asset does not appear * Check database to make sure asset is removed. |

|  |
| --- |
| FRONT – Resize Asset  As a scene designer, I must be able to change an assets size, so objects suit my scene.  Must. 2. |
| BACK   * In scene editor * Edit Size * Does asset resize in scene? * Select save * Refresh page to make sure asset size is stored * Check database to make sure size is updated. |

|  |
| --- |
| FRONT – View all Assets  As a scene designer, I must be able to view all assets edit properties in the scene.  Must. 2. |
| BACK   * In scene editor * Add each type of asset * Do their property boxes appear in the scene? * Refresh page to make sure assets all still appear |

|  |
| --- |
| FRONT – Change Text Asset Font Colour  As a scene designer, I must be able to change font colour, so all text objects can suit my scene.  Must. 1. |
| BACK   * In scene editor * Select “Change Colour” of text asset * Does the assets font colour change in scene? * Select “Save.” * Refresh page to make sure colour is still changed. * Check database to make sure colour is stored. |

|  |
| --- |
| FRONT – Change Asset Position  As a scene designer, I must be able to edit any assets position within the scene, so i am able to place objects suitably.  Must. 2. |
| BACK   * In scene editor * Edit each type of assets position * Does the assets position change in scene? * Select “Save.” * Refresh page to make sure position is still changed. * Check database to make sure position is stored. |

|  |
| --- |
| FRONT – Change Asset Rotation  As a scene designer, I must be able to edit objects rotation within the scene, so i am able to rotate objects suitably.  Must. 4. |
| BACK   * In scene editor * Edit each type of assets rotation * Does the assets rotation change in scene? * Select “Save.” * Refresh page to make sure rotation is still changed. * Check database to make sure rotation is stored. |

|  |
| --- |
| FRONT – Export Course  As a scene designer, I must be able to export my course from within the scene editor, so i am able to test it in a learning management system.  Must. 4. |
| BACK   * In scene editor * Select Export * Does the export go to your download folder? * Upload course to Rustici Software’s free LMS. * Does course function with all assets included? |

|  |
| --- |
| FRONT – Display Questions during scene  As a scene designer, I would like to be able to display questions at specific times during videos, so to engage content viewer.  Would. 4. |
| BACK   * In scene editor * Enter in Question property box “Time to be displayed” * Refresh the page, does the question appear when expected? * Check database to make sure time is stored |

|  |
| --- |
| FRONT – Assign Course to User  As an admin user, I must be able to assign courses to standard users, so standard users can only access content suited to them  Must. 2. |
| BACK   * In View Courses Page * Select “Assign Course” * Select Users to assign course to * Log in as the selected user * Does the course appear on their home page? |

|  |
| --- |
| FRONT – View Courses as Standard User  As a standard user, I must be able to view courses that have been assigned to me, so i can become a scene editor and edit them.  Must. 0.5. |
| BACK   * \*PRIOR: As admin assign courses to user * Log in to application as standard user * Do only assigned courses appear on home page? |

|  |
| --- |
| FRONT – Upload Images from URL  As a scene designer, i would like to be able to upload images from a URL, so i am able to user images from online.  Would. 1 |
| BACK   * In Scene Editor * Select “Add Image” * Enter URL as value * Select save * Does the image appear in the scene? * Check database to make sure asset is stored under the scene. |

## MoSCoW statements

## Iteration Plan

The goal of this iteration plan is to maintain organisation throughout development. This plan will also assist in identifying the realistic scope of the project. The proposed user stories above will be incorporated into the iteration plan to perform as the product backlog. The gantt chart below proposes an estimated timeline of the project.

## Schedule of Iteration Workflows

This section provides a summary of the activities contained within each stage of the project. The development iteration plan will be discussed below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TASK NAME** | **START DATE** | **DAY OF PROJECT\*** | **END DATE** | **DURATION\* (WORK DAYS)** | **DAYS COMPLETE\*** | **DAYS REMAINING\*** | **PERCENT COMPLETE** |
|
| **PLANNING** |  |  |  |  |  |  |  |
| Research ASP.net | 01 June 2019 | 1 | 14 June 2019 | 14 | 14 | 0 | 100% |
| Research eLearning | 01 June 2019 | 1 | 07 June 2019 | 7 | 7 | 0 | 100% |
| Research VR Framework / Libraries | 07 June 2019 | 7 | 14 June 2019 | 8 | 8 | 0 | 100% |
| Prototype VR Frameworks / Libraries | 10 June 2019 | 10 | 20 June 2019 | 11 | 11 | 0 | 100% |
| Research Design Patterns | 17 June 2019 | 17 | 01 July 2019 | 15 | 15 | 0 | 100% |
| Requirements Gathering | 17 June 2019 | 17 | 08 July 2019 | 22 | 22 | 0 | 100% |
| Prototype Scene Editor | 05 July 2019 | 34 | 12 July 2019 | 8 | 8 | 0 | 100% |
| Develop Wireframes and plan view | 24 June 2019 | 24 | 16 July 2019 | 23 | 23 | 0 | 100% |
| **DEVELOPMENT** |  |  |  |  |  |  |  |
| Design basic Database - Code First migrations | 05 July 2019 | 35 | 12 July 2019 | 8 | 8 | 0 | 100% |
| Follow Iteration Plan | 05 July 2019 | 35 | 23 August 2019 | 50 | 50 | 0 | 100% |
| **TESTING AND REPORT** |  |  |  |  |  |  |  |
| Report | 22 July 2019 | 52 | 30 August 2019 | 40 | 40 | 0 | 100% |
| Usability Testing | 23 August 2019 | 83 | 30 August 2019 | 8 | 8 | 0 | 100% |
| Unit Testing | 23 August 2019 | 83 | 04 September 2019 | 13 | 13 | 0 | 100% |
| **DEPLOYMENT** |  |  |  |  |  |  |  |
| Unit Testing | 02 September 2019 | 94 | 04 September 2019 | 3 | 0 | 3 | 0% |

## Iteration Plan with Story Points

The development iteration plan will progress over 7 weeks. This will allow 49 full days of work. Story points will represent 4-12 hours of development work. With 49 days, at 12 hours a day, the total number of hours available is 588. By working roughly 6-8 hours a day, the hours of development will fall between 294 and 392.

The iteration plan shows a large backlog of story points which doesn’t start to balance out until the third iteration. With appropriate time management and incorporating productive best practices into development, the project is achievable in the designated time.

### Sprint One Tasks

|  |  |
| --- | --- |
| TASK | STORY POINT |
| Register Account | 1 |
| Log In | 0.5 |
| Log Out | 0.5 |
| Add User | 1 |
| Add Course | 0.5 |
| Add Scene | 0.5 |
| Delete User | 0.5 |
| Delete Course | 0.5 |
| Delete Scene | 0.5 |
| Edit User Details | 1 |
| Edit Own Details | 2 |
| Edit Course Details | 1 |
| Edit Scene Details | 1 |
| Add Text Asset to VR Scene | 2 |
| Add Audio Asset to VR Scene | 2 |
| Add HotSpot Asset to VR Scene | 2 |
|  |  |
|  |  |

### Sprint Two Tasks

|  |  |
| --- | --- |
| TASK | STORY POINT |
| Add HotSpot Asset to VR Scene | 2 |
| Add Image Asset to VR Scene | 2 |
| Add Question Asset to VR Scene | 4 |
| Add Video Asset To VR Scene | 2 |
| Add Background Image to Scene | 2 |
| Add Video Background to Scene | 2 |
| Delete Asset | 1 |

### Sprint Three Tasks

|  |  |
| --- | --- |
| TASK | STORY POINT |
| Resize Asset | 2 |
| View All Assets | 2 |
| Change Text Asset Font Colour | 1 |
| Change Asset Position | 2 |
| Change Asset Rotation | 4 |
| Export Course | 4 |
| Assing Course to User | 2 |
| View Courses as Standard User | 0.5 |