# ArchiVR User Specification

# Description

The Archi-VR application offers the functionality of previewing architectural projects in different levels of immersion: Desktop, Mobile, AR and VR.

Rendering can be either:

* Mono-scopic
* Stereo-scopic

Supported input devices:

* Keyboard
* Mouse
* Touch
* Gyro
* …

Application offers following camera navigation modes:

* FPS (First-Person-Shooter)
* Fly
* Teleport
* AR
* …

The application is targeted for a wide variety of platforms and hardware:

* Desktop
  + Windows
* Mobile
  + 2D
  + GearVR(TODO)
  + Google Cardboard (TODO)
* VR Headset
  + Oculus
  + Google DayDream (TODO)
  + Windows Mixed Reality (TODO)

Every platform/hardware has its own set of supported immersion modes, navigation modes, graphical detail, etc… depending of the system-specific available capabilities:

* available input devices
* rendering performance
* display type
* **…**

**EG On a desktop, *gyro rotation* is not be available.**

**On a GearVR, *positional tracking* is not available.**

# Supported platforms

The application supports the following platforms:

* Desktop
* Mobile handheld
* Mobile HMD
  + Cardboard
  + GearVR

# ArchiVR Concepts and Objects

The application has notion of the concept ‘Project’.

## Project

A project represents all data related to a specific architectural project. This includes all data during the entire lifetime of the project. Data from different discrete snapshots in time are subdivided into project Phases.

The application has notion of the concept of ‘active Project’ versus ‘inactive project’.

The application has at any given time at most 1 ‘active’ Project. The active Project references the construction project (from the list of included available Projects) to be currently viewed.

All Projects but the active Project, are inactive.

The active Project might change in time.

## Phase

A project contains 1 or more phases. A project phase represents the state of the construction project at a specified point in time.

Example phases of a new build project

* Before
  + Initial
* During build
  + Step 1
  + Step 2
* After
  + Final

Example phases of a renovation project

* Before
  + Initial
* During
  + Tear-down
    - Step 1
    - …
  + Rebuild
    - Step 1
    - Step 2
    - …
* After
  + Final

# Menu

In all modes, and at all times, the application shows either:

* The default, basic menu fort he current mode, or
* One of the ‘extended’ menus.

The default menu is a very basic menu, that can differ depending on the current view mode and application state. All default menus offer the following common functionality:

* Button show the root extended menu.
* Button to set the view mode.

The root extended menu can be opened by

* Keyboard: Pressing the **M** key.
* Gamepad: Pressing the **Select** button.
* UI: LMB click or timed gaze on the button to open the root extended menu.

A menu can be closed by:

* Pressing the **Esc** key.
* Clicking the **‘X’** button at the top-right of the menu
  + Using the left mouse button.
  + Using timed gaze.
* Pressing the **‘Y’** button on the gamepad.

When a menu is closed, its parent menu is shown.

* Default
  + Settings
    - Controls
      * On-screen gamepad (enable)
    - Graphics
      * Level (select)
      * Dynamic vegetation (enable)
      * Clouds (enable, edit)
    - Sound
  + Time
  + …

A menu button can be clicked by:

* Clicking it using the left mouse button.
* Using timed gaze.
* Selecting it, and:
  + Keyboard: pressing the ‘return’ key.
  + Gamepad: pressing the ‘X’ key.

# Modes

The Archi-VR application will have the following application states:

* **Home** application state
* **Play** application state

# Startup

The application always starts up in the **Home Menu** application state.

# **Home menu** application state

When entering this mode, the **Home** menu is loaded and shown.

The **Home** menu exists in 2 flavours:

* **non-VR** mode
* **VR** mode

Upon first entry in the Home application state, it is checked whether the initial active VR device is Oculus (GearVR). If this is the case:

* The view mode is set to HMD stereo, and it view mode cannot be changed by the user.
* The UI mode is set to ‘VR’ and UI mode cannot be changed by the user.

## Non-VR mode Home menu

When the UI mode is set to **Non-VR**, the Home menu is represented using a screen-space full-screen overlay menu. The application has notion of a ‘selected’ project. There is always exactly one selected project. The user can browse the available projects, thereby changing the selected project. The user can also transition into **View Project** mode, for viewing the selected project, using the ‘Go’ button.

* A preview image of the selected Project is shown using a full-screen 2D preview image.
* The selected project’s name is shown by a Text UI control, located at the bottom-center of the screen.
* 2 Project selection browse buttons are shown to the lef, respectively right side of the selected project name Text at the bottom of the menu:
  + ‘Previous project selection’ button:
    - Caption ‘<’
    - Tot he left of the selected project name
  + ‘Next project selection’ button:
    - Caption ‘>’
    - To the left of the selected project name
* The portfolio name is shown by a Text UI control at the top-center of the menu.
* The credit ‘Powered by Archi-VR’ is shown by a Text UI control at the top-right of the menu.
* A ‘Go’ button
  + Caption ‘Go’
* An ‘Exit’ button
  + Caption ‘X’

## VR mode Home menu

When the UI mode is set to **VR**, the Home menu is represented using world-space UI controls. Because mouse/keyboard/touchscreen/gamepad input is not always present in VR systems, the UI control scan be interacted with, by using timed gaze. When the user starts gazing at a control, it is highlighted to visualize this. (For now the control is scaled a bit bigger while gazing at it.)

The **VR mode Home** menu consists of the following components:

* Project selection menu
* Floor menu

### Project selection menu

The project selection menu consists of **project select** buttons. For each available project, a project select button is shown as a world-space button, that shows a preview image of its corresponding project. Project select buttons are layed out in a circular order around the user’s view position. Project select buttons are ofsetted at a fixed distance from the user view position, and face the user view position. The user can transition into **View Project** mode for a specific project, by gazing at the corresponding **project select** button.

### Floor menu

The default menu is shown as a floor menu. Extended menus are shown as front menus.

The floor menu consists of the following controls:

* The portfolio name is shown by a Text UI control at the top-center of the menu.
* The credit ‘Powered by Archi-VR’ is shown by a Text UI control at the top-right of the menu.
* ‘Settings’ button
  + Image: Cog wheel
* ‘View Mode’ button
  + Image: Icon corresponding to current view mode.
* ‘Exit’ button
  + Caption: ‘X’

The user can:

* Open a specific project for viewing.
  + By gazing at the corresponding **Project Select** button.
* Open the Settings menu.
  + By gazing the **Settings** button.
* Exit the app.
  + By gazing the **Exit** button.

When opening a project:

1. The application leaves the Main Menu.
2. The application enters ‘View Project’ mode, for the active architectural project.

# **Play** application state

When entering this application state:

1. The application opens the 3D model of the active project.
2. The application loads and shows the UI controls for the **View Project** menu.

## View Project menu

The View Project menu is shown when ‘Show UI’ option is on. It is hidden when ‘Show UI’ option is off.

### View Project menu controls

The View Project menu contains the following controls:

* **Home** button
* **Show/Hide UI** button
* **Rotate Mode** button
* **View Mode** button
* **Time** button
* **Construction Lighting Mode** button

### MR and Non-MR flavours

The ‘View Project’ menu exists in 2 flavours:

* Non-MixedReality: screen-space UI
* MixedReality: World-space UI.

The **View Project** menu is always represented by a single canvas that contains all of the UI controls that represent the menu.

In the **non-MR** flavour, the ‘View Project’ menu is represented by a screen-space overlay canvas. The controls are standard UI controls that can be interacted with by mouse clicks and taps.

In the **MR** flavour, the ‘View Project’ menu is represented by a world-space canvas. The controls are special UI controls targeted at VR, that can be interacted with by gazing. The canvas is attached to the player position, and is located on an horizontal ground plane a bit below the player position. When the user is looking forward or up, the menu is oriented automatically to the user viewing direction. This ensures that the menu is always located in front of the player when he starts looking down. When the player looks down, and the menu cones in view, the canvas orientation becomes fixed, in order to enable the player to gaze at controls in order to interact with them.

## Keyboard key functions

|  |  |
| --- | --- |
| L | Toggle Construction Lighting mode |
| M | Show/hide Menu |
| N | Toggle Navigation mode |
| U | Move Up |
| D | Move Down |
| Arrow up | Move Forward |
| Arrow down | Move backward |
| Arrow Left | Move left |
| Arrow Right | Move right |
| Shift | Fast movement |
| L | Show/hide debug Logging |
| Fx | Show debug logging menu x |
| C | Capture Screen |
| Q | Toggle Graphics Quality Level |

TODO?

* F: Show/Hide FPS counter
* Home: Open main menu

## Show/Hide UI

While in the View Project mode, all UI can by shown/hidden by:

* On devices with keyboard: pressing the ‘U’ key. (from UI)
* On touch-enabled devices: tapping anywhere on the window, where there is no UI control. 🡪 Broken currently?

## Time animation

The animation of time can be controlled with the following keyboard keys:

* F: Increase animation speed Forward.
* B: Increase animation speed Backward.
* S: Stop animation. (Set animation speed to 0)

The user can also tap or click the corresponding on-screen overlay controls. (TODO)

The time animation controls the celestial lighting in the scene.

## Construction lighting mode

The **Construction Lighting Mode** can be set to one of the following modes:

* AUTO: Construction lights are automatically turned ON or OFF, dependant on the current time.
* ON: Construction lights are always ON, irrespective of the current time.
* OFF: Construction lights are always OFF, irrespective of the current time.

When entering the ‘View Project’ state, the **Construction Lighting Mode** is initially set to AUTO.

In the menu, a ‘Construction Light Mode’ button is present, that represents the current ‘Construction Light Mode’ by means of a dynamic icon:

* ON: lightbulb with rays.
  + 
* OFF: lightbulb without rays.
  + 
* AUTO: lightbulb with caption ‘A’. (from Auto)
  + 

While in the View Project mode, the construction light mode can be changed by:

* On devices with keyboard:
  + Pressing the ‘C’ key. (from Construction light Mode)
* On devices with mouse input:
  + Clicking the ‘Construction Light Mode’ button.
* On devices with touch input:
  + Tapping the ‘Construction Light Mode’ button.

## FPS counter

The application keeps track of the Frames-Per-Second. The FPS counter is shown be default initially at the first launch of the application, and its last show/hide setting is stored when closing the application, and reapplied at the next startup of the application. In non-VR mode, the FPS counter is located in the screen-space overlay canvas. In VR mode, the FPS counter is shown on a world-space Text UI control that is attached at a fixed offset in front of the player head position.

Show/hide FPS counter can be done as follows:

* On all devices: by enabling/disabling **Settings > Graphics Settings > Show FPS** menu option
* On devices with keyboard: pressing the ‘F’ key. (from FPS)

Note: The FPS counter is subject to the **Show UI** option. If **Show UI** option is off, the FPS counter is hidden, along wit all other UI, irrespective of the **Show FPS counter** option.

The FPS counter also computes statistics, like the minimum/maximum and mean FPS. These statistics are saved in a file ‘FPS stats.txt’, located under the Application persistent data path.

# Navigating through the model

The user can navigate through the model using different input devices and methods:

* Virtual D-Pad
  + Supports both Touch and Mouse input.
* Touch
  + Multi-touch
  + Swipe
  + Tap
* Gamepad
  + Physical DPad
  + Physical Button
* Gyro
* Mouse
  + Drag
  + Click
* Keyboard
  + Key presses
* GearVR trackpad
  + Swipe
  + Tap
* 6DOF tracking
  + Kudan SLAM
  + WMTracker

Camera navigation can be devided in 2 operations:

* Camera translation
* Camera rotation

While a screen-space menu is visible, camera navigation is off by default. To enable camera navigation while a screen-space menu is visible:

* Press the Left Control key
* Press the Physical gamepad button XXX

TODO? Instead of forcing the above buttons tob e pressed continuously, toggle Camera Navigation on/off upon single-clicking them.

## Camera navigation mode

Camera translation can be done in one of the following modes:

* Fly mode
* FPS Mode
* Teleport mode
  + Timed gaze
  + Click
  + ? Constrained: Collision with model: User can not move through model

Camera translation can be controlled depending on the mode as follows:

### Camera rotation

Camera rotation is unconstrained (softwarematically speaking) when controlled by gyro or 6DOF tracking. Camera rotation by Gamepad, Virtual D-Pad, Mouse, Keyboard is softwarematically constrained in the vertical direction to [-90,90] degrees (IE User can not flip view upside-down.)

Camera rotation can be controlled using:

* Gyro
* Mouse
  + Drag
  + ?Mouse-lock mouse move?
* Swipe
  + Only on touch-enabled devices
  + ?Looks suspiciously like MouseDrag?
* ?Virtual D-Pad?
* ?Keyboard keys?

### Fly Mode

In fly mode, the virtual user translation is unconstrained. The virtual user is not subject to gravity or collisions with objects in the virtual world. The user can move (fly) wherever he likes. User can translate in all directions (including up/down) at will and move freely through objects in the virtual world.

#### Inputs

In fly mode the possible inputs are:

* Move forward
* Move backward
* Move left
* Move right
* Move Up
* Move down
* Fast movement
  + Increase the speed of movement.
* Slow movement?
  + Decrease the speed of movement.

The default controls on the different supported devices are:

* Virtual D-Pad
  + Y-only D-Pad on left side of the screen:
    - Move Up/Down
  + XY D-Pad on right side of the screen:
    - Move Forward/Backward/Left/Right
* Keyboard
  + Arrows: Move Forward/Backward/Left/Right
  + u/d: Move Up/Down
  + Shift: fast movement
* Physical GamePad
  + Left D-Pad: Move Up/down
  + Right D-Pad: Move Forward/Backward/Left/Right
* GearVR Trackpad
  + Swipe to any direction and hold pressed at the end of the swipe to move:
    - Horizontal swipe = Move Fwd/Backwd
    - Vertical swipe = Move Up/Down
* Touch Swipe
  + Same as GearVR Trackpad?

### FPS Mode

In FPS mode, translation is constrained in a somewhat more realistic First-Person perspective way. In FPS mode, translational movement is constrained: The virtual user is subject to gravity and collisions with objects in the virtual world. The user can translate in horizontal directions, and jump/crouch. Movement is constrained: Collision with model: User can not move through model.

#### Inputs

In fly mode the possible inputs are:

* Move forward
* Move backward
* Move left
* Move right
* Move Up
* Move down
* Fast movement

### Teleport Mode

In this mode, the user can translate around in the virtual world by picking (mouse click or gaze) objects in the virtual world. Upon a picking operation, the user translates towards the picked position (or better, to an ofsetted position from it).

#### Inputs

The inputs for this mode are:

* Pointing out the target for the teleport operation
  + Mouse pointer
  + Gaze operation
* Confirmation of the teleport operation
  + LMB Click
  + Touch tap
  + Timed gaze (need special teleport targets fort his mode)

### Vuforia AR Mode

In this mode, the user can review the architectural design as a virtual maquette, attached to a physical VUforia marker image.

Mouse and keyboard input are not supported in this mode. Gamepad is only supported in case of a headmounted XR device being active.

Modifying model-marker offset  
In the left of the screen, a vertical D-pad is shown that enables modifying the vertical translational offset of the model from the image marker: increase/decrease offset of model from the image marker, along the marker image normal.

In the left of the screen, a horizontal D-pad is shown that enables modifying the rotational offset of the model from the image marker: rotate the model around the image marker normal.

TODO? Enable POI navigation : upon selecting POI, this becomes the new ‘anchor point’ fort he model.

## Supported navigation modes

Supported navigation modes depend on the system running the application:

### Desktop

|  |  |
| --- | --- |
| Translation | Rotation |
| Keyboard | Keyboard  Mouse  Gyro  Swipe (Only mono view modes)  Virtual GamePad (Only mono view modes)  Physical Gamepad |
| Virtual Gamepad | Keyboard  Mouse  Gyro  Swipe (Only mono view modes)  Virtual GamePad (Only mono view modes)  Physical Gamepad |
| Physical Gamepad | Keyboard  Mouse  Gyro  Swipe (Only mono view modes)  Virtual GamePad (Only mono view modes)  Physical Gamepad |
| Tracked | Tracked |

### Mobile

|  |  |
| --- | --- |
| Translation | Rotation |
| Virtual Gamepad | Gyro  Swipe (Only mono view modes)  Virtual GamePad (Only mono view modes)  Physical Gamepad |
| Physical Gamepad | Gyro  Physical Gamepad |
| GearVR Trackpad | Gyro |
| Tracked | Tracked |

### GearVR

|  |  |
| --- | --- |
| Translation | Rotation |
| Physical Gamepad | Gyro  Physical Gamepad |
| GearVR Trackpad | Gyro |
| Tracked | Tracked |

## Default navigation mode

The default Camera Navigation mode depends on the System:

* Desktop
  + Rotation :
    - Gamepad
    - Mouse
    - Touch
    - Keyboard
  + Translation
    - Gamepad
    - Touch
    - Keyboard
* Mobile
  + Rotation
    - Gyro
    - Virtual D-Pad
  + Translation
    - Gamepad
    - Virtual D-Pad
* GearVR
  + Rotation:
    - Gyro
  + Translation
    - Gamepad
    - TrackPad
    - Teleport
      * Which method? TimedGaze/Picking/UIButtonClick/UIListBoxSelection

When quitting the application, the current Navigation mode is stored. This previously stored Navigation Mode is then reapplied upon the next Application launch.

# Settings menu

The settings menu consists of the following controls:

* An ‘Exit’ button.
* A button to open the **GraphicsSettings** menu.
* A button to open the **ControlsSettings** menu.

# ControlsSettings menu

The **ControlsSettings** menu consists of the following controls:

* An ‘Exit’ button
* An option button to set the camera rotation control mode:
  + Gyro
  + Physical Gamepad
  + Virtual Gamepad
  + Swipe
  + Tracked
* An option button to set the camera translation control mode:
  + Keyboard
  + Physical Gamepad
  + Virtual Gamepad
  + Tracked

# GraphicsSettings menu

The **GraphicsSettings** menu consists of the following controls:

* An ‘Exit’ button
* A checkbox to enable/disable dynamic vegetation
* A checkbox to show/hide the FPS counter
* An option button to set the graphics Quality level:
  + Low
  + Medium
  + High
  + Very High
  + Ultimate