D5.4 WEKIT
Visualization Design Solutions

Editors: Jazz Rasool
Carl Smith
Brigitta Zics
## Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Contributor(s)</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>30.01.2017</td>
<td>Jazz Rasool</td>
<td>Collated work done by Ravensbourne UI meeting, compiled by Fridolin Wild. Added UI elements designed for Unity To be added: The initial mock-ups were created with a boxed, linear theme. It was discussed that menus in a 3D environment should not have a flat feel but a 3D curved feel.</td>
</tr>
<tr>
<td>0.2</td>
<td>30.01.2017</td>
<td>Carl Smith</td>
<td>Oxford AR Winter School UI meeting content added</td>
</tr>
<tr>
<td>0.3</td>
<td>14.02.2017</td>
<td>Carl Smith</td>
<td>Collated work done by Ravensbourne and Oxford Brookes</td>
</tr>
<tr>
<td>0.4</td>
<td>21.02.2017</td>
<td>Carl Smith</td>
<td>Internal Review</td>
</tr>
<tr>
<td>0.5</td>
<td>24.02.2017</td>
<td>Carl Smith, Mikhail Fominykh</td>
<td>Integrating feedback from peer reviews, style and formatting</td>
</tr>
<tr>
<td>0.6</td>
<td>27.02.2017</td>
<td>Carl Smith</td>
<td>Final revisions and polishing</td>
</tr>
<tr>
<td>1.0</td>
<td>28.02.2017</td>
<td>Cinzia Rubattino</td>
<td>Final style and overall revision</td>
</tr>
</tbody>
</table>

Disclaimer: All information included in this document is subject to change without notice. The Members of the WEKIT Consortium make no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Members of the WEKIT Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.
WEKIT Visualization Design Solutions

WP 5 | D5.4

Editors:
Jazz Rasool (RAVE)

Authors:
Jazz Rasool (RAVE)
Carl Smith (RAVE)
Brigitta Zics (RAVE)
Alla Vovk (OBU)
Daniele Di Mitri (OUNL)
Fridolin Wild (OBU)
Jaakko Karjalainen (VTT)
Kaj Helin (VTT)
Roland Klemke (OUNL)
Soyeb Aswat (MP)
Tre Azam (MP)
Will Guest (OBU)

Reviewers:
Mikhail Fominykh (EP)
Puneet Sharma (UIT)
Paul Lefrere (CCA)

Deliverable number D5.4
Dissemination level Public
Version 1.0
Status Final
Date 28.02.2017
Due date 28.02.2017
Table of Contents

EXECUTIVE SUMMARY .................................................................................................................. 5

1. METHODOLOGY ......................................................................................................................... 6
   1.1. IDENTIFIED UI ELEMENTS ................................................................................................. 7
       1.1.1. DOT CURSOR AND OBJECT HIGHLIGHT ..................................................................... 7
       1.1.2. SCANNER LINE ............................................................................................................. 7
       1.1.3. CONTEXT MENU ......................................................................................................... 8
       1.1.4. ANCHORED GHOST-HAND RECORDING ..................................................................... 8
       1.1.5. PHOTO OF THE SITUATION ‘TO BE’ - ‘LOOKING INTO THE FUTURE’ ......................... 9
       1.1.6. VIDEO WALL ................................................................................................................ 9
       1.1.7. SUPERIMPOSITION OF 3D MODEL OR ANIMATION ..................................................... 10
       1.1.8. BREAD CRUMBS NAVIGATION FOR ‘CURRENT TASK’ ................................................ 10
       1.1.9. DIRECTIONAL SOUND ................................................................................................ 11
       1.1.10. LOOK HERE ................................................................................................................ 11
       1.1.11. STAND HERE ............................................................................................................ 12
       1.1.12. SHARED POINTER ...................................................................................................... 13
       1.1.13. RECORD EVIDENCE .................................................................................................. 13
       1.1.14. FITNESS DASHBOARD ................................................................................................. 13
       1.1.15. VISUAL Glyphs (e.g. ARROWS) AND LABELS ............................................................. 14
       1.1.16. ACTIVATE BY STARE GAZE ....................................................................................... 14
       1.1.17. TOUCH POINTS .......................................................................................................... 15
       1.1.18. VIRTUAL POST IT NOTES .......................................................................................... 15

2. RESOURCE PLANNING ............................................................................................................... 16

3. RESPONSIBILITIES .................................................................................................................. 17

4. PROTOTYPING UI ELEMENTS IN UNITY ............................................................................ 18
   4.1. FUNCTIONS AND TRANSITIONS IN MANIKIN MEDICAL SIM ........................................... 20
       4.1.1. First scene with name, institution, submit button ......................................................... 20
       4.1.2. Scan / Start Scene ....................................................................................................... 20
       4.1.3. Scan mannequin ......................................................................................................... 20
       4.1.4. Show mannequin overlaid with AR features ............................................................... 20
       4.1.5. (Scenario specific): Asthma example ......................................................................... 20
       4.1.6. Visual Assessment ..................................................................................................... 21
       4.1.7. Audio ......................................................................................................................... 21
       4.1.8. Haptic ....................................................................................................................... 21
       4.1.9. Other Diagnostic Tool: ............................................................................................. 21

5. CONCLUSIONS ......................................................................................................................... 25
Executive summary

The objective the WP5.4 Deliverable relates to design areas of WP5 specifically Visualization Design Solutions and more generally the following WP5 challenges:

- To define a design methodology for wearability and for ergonomics in workplaces.
- To define design recommendations for wearable experience capturing and re-enactment.
- To perform user testing of hardware and software design solutions.
- To generate an interactive toolkit.
- To create an interactive repository of captured experience content.

These design, experience and interaction elements are focused into WP5.4 to explore,

- The look and feel of the User Interface (UI) and User Experience (UX). These were explored at the UI Workshop at Ravensbourne. The workshop revealed that many UI and UX elements would be good to include. Journey maps/breadcrumb trails as well as icons/symbols to mark equipment with for AR triggering and guidance. Sets of UI/UX for Capture and Replay stages, including common elements shared by both, were formulated, designed and collated.

- The challenges of effective workflow including information overload and decision making processes in new situations. Continuous partial attention and information overload are often big risks. How can user centred design create hybrid interventions that do not distract people from meaningful engagement with the objects they are designed to augment.

This document takes the collective work done so far on the types of interaction supported by the WEKIT first prototype. It includes

- An initial set of User Interface design elements jointly elaborated in a workshop in London at Ravensbourne University and compiled by Fridolin Wild
- Samples of UI and UX elements currently in use in Microsoft Hololens AR Applications, compiled and collated by Jazz Rasool and Brigitta Zics of Ravensbourne.
- Initial prototype menu elements and sample medical scenario created in Unity 3D Engine software by Alla Vovk.
1. Methodology

An initial workshop was held on October 25th, 2016, at Ravensbourne University, London, UK

Attended by: Alla Vovk, Will Guest, Fridolin Wild, Kaj Helin (bringing in also input from Jaakko Karjalainen), Daniele Di Mitri, Roland Klemke, Tre Azam, Soyeb Aswat, Jazz Rasool, Brigitta Zics.

Some A2 sheets were taken to brainstorm different User Interface (UI) element families.

The final results were photographed and then sample prototype elements created in Unity 3D Engine software for the Medical Scenario.

Figure 1. A2 Brainstorm Sheets Identifying UI Families

A follow up meeting also at Ravensbourne was held on the 6th December.

A further meeting took place at Oxford Brookes on the 25th January, 2017 to continue work on the User Interface Design.

A further UI meeting took place in Holland on the 7th February, 2017
2. **Identified UI Elements**

To support all transfer mechanisms on the relevant sensors, triggers as well as hardware or software interface elements, the following items were identified in the London workshop.

### 2.1. Dot cursor and object highlight

![Figure 2. Dot cursor and object highlight](image)

Imagine a laser beam coming out of the centre of the field of view. This laser beam would be invisible (unless explicitly made visible), but where it is clear to see its reflection (concentric circles, cross-hairs, or similar) from objects and surfaces is touches.

This should be combined with the ability to highlight (think ‘glow’) known objects relevant to the task at hand.

Could be updated with the sound. When cursor is on the target, the trainee gets an audio signal (something like a "click" sound) as an approval that it on a right place. A few applications with and without sound were tried when a button was highlighted, and it encouraged presence and engagement when the sound effect was triggered, giving an awareness that the user is on the right track of learning.

### 2.2. Scanner line

![Figure 3. Scanner Line](image)

Like with a barcode scanner, a striding line could be superimposed on the manikin to allow for the selection of the MRI scans. Other than the dot cursor above, the scanner line has angle information as well, indicating along which axes/angle an image slice is available and can be requested.
2.3. Context menu

![Context Menu](image)

**Figure 4.** Context Menu

Air tapping brings up a radial menu dial on the cursor location. The segments of the dial are the menu items, icons indicate what functionality an entry stands for. Striding the cursor over an item brings up a text label. Air-tapping a highlighted item selects it.

2.4. Anchored ghost-hand recording

![Anchored Ghost Hand Recording](image)

**Figure 5.** Anchored Ghost Hand Recording

A recording of the expert’s hand movement (plus audio think aloud explanation) is anchored on the object it refers to. This allows the trainee to move freely around the object, pausing/replaying/inspecting the expert’s movement as needed.
2.5. Photo of the situation ‘to be’ - ‘Looking into the future’

Virtual sticky notes can be left stuck on (floating on the location) or tangled (connected with a dangling line) to objects. One special case of this is the ability to leave a 'to be' photograph showing what the completed step of the task should look like. Probably this deserves its own menu item for authoring, though.

![Figure 6. To Be Photo](image)

2.6. Video wall

A window with an embedded video can be attached to a wall or surface. It is tangled with a virtual rope to the location where it matters.

![Figure 7. Video Wall](image)
2.7. Superimposition of 3D model or animation

(Animated) 3D models can be superimposed over the actual object. Or to the side of it to avoid obstructing the view on the real object. If animated, this should allow for slow motion or zoom.

Figure 8. 3D Model Superimposition

2.8. Bread crumbs navigation for ‘current task’

The task is depicted with simple shapes (line with circles? See bottom scribble). The current step is highlighted. Buttons for ‘back’ and ‘next’ allow navigation through the task.

Figure 9. Breadcrumbs Navigation
This is probably the only user interface element that is placed in a fixed way on the display to be always on. Alternatively it could be placed in space just like the menu for the Galaxy explorer (floating in space just a bit below the line of sight and close to the user’s body, thus allowing to place the cursor on menu item and air tap to select). Or similar to the main menu on the HoloLens (float in front of the user, follow the cursor).

### 2.9. Directional sound

Unity allows placing sound in space - and works out the details of directional sound then automatically. This is a subtle form of moving the user around: when looking in the wrong direction, s/he hears where the action is. It might make sense to possibly lower the volume with distance to indicate a need to move closer.

Using different AR applications, it was found that sound usually has the same level for all instructions (in both cases — general guidance and delayed reaction when user for some reason inactive in interaction).

This argument could consistently complete the following point about gaze direction. When user is inactive the cursor with "Look here" appears in front with an audio comment "Follow the cursor".

### 2.10. Look here

![Figure 10. Directional Sound](image)

![Figure 11. Look Here](image)
A look here glyph and arrow appears, always trying to move to the centre of the field of view of the user (with a slight delay) up until the user looks where required.

More difficult might be the implementation of a generic event handler for ‘autopausing’ video/animation/other action to make sure the user is not missing anything while dreaming away in the wrong direction.

![Auto pause](image1.png)

**Figure 12. Auto pause**

### 2.11. Stand here

![Stand Here](image2.png)

**Figure 13. Stand Here**

Spot is marked up where the user should stand to have the experience. Best place the glyph in the real environment.

This and the above item include working out how to select good spots in the room model: to work out, for example, where there is enough space to place an animated 3D character. And to where then move the user to see the character best (and where to direct the gaze).
2.12. Shared Pointer

![Shared Pointer](image1)

**Figure 14. Shared Pointer**

When two or more users work in the same work space collaboratively, they need to see what each other is focusing on. The laser pointer cursor of the other person should become visible in a different colour, a connected label should list the user name (initials, first name, or some such).

2.13. Record Evidence

![Record Evidence](image2)

**Figure 15. Record Evidence**

The user takes a photograph of the finished step. This is triggered by the activity script, prompting the user to do it where the instructional designer thought it would be useful for assessment purposes. The evidence can then be mailed off to an assessor, who provides feedback (and hands out brownie points for high performance).

2.14. Fitness dashboard

The bio data gathered is made available on a fitness dashboard (that can be snapped to a wall or surface). It at least shows current/last mental effort and physical activity (vhr, posture).
2.15. **Visual glyphs (e.g. arrows) and labels**

Figure 17. Visual Glyphs (e.g. arrows) and labels

Animated 3D symbols can be placed directional in space. Labels can be attached to it.

2.16. **Activate by Stare Gaze**

Segment fills slowly while the cursor is locked in on a virtual button. When the circle is completed, the function triggers.

Figure 18. Activate by Stare Gaze
2.17. Touch points

Spherical highlights (maybe even with glowing core) signal what the user should come closer to as well as allow to launch functionality by touching (or cursor-locking and air tapping) the spot.

![Figure 19. Touch Points](image)

2.18. Virtual Post It notes

The Post-Its are attached to object or snapped to surfaces. See above, point 5.

![Figure 20. Virtual Post-It Notes](image)

A special case of this is the Post It with images embedded.

![Figure 21. Embedded Post-It notes](image)
### 3. Resource Planning

**Table 1.** Resource Planning. The columns on the right for Cost and Priority will be completed in future iterations of designs for the WEKIT prototypes for software and the capture and re-enactment functions.

<table>
<thead>
<tr>
<th>UI # and name</th>
<th>Cost  (A=\text{hours, } B=\text{day/s, } C=\text{week/s})</th>
<th>Priority  (A=\text{mandatory, } B=\text{desired, } C=\text{optional})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dot cursor and object highlight</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>scanner line</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Context menu</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Anchored ghost-hand recording</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Photo of the situation ‘to be’ - ‘Looking into the future’</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>video wall</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Superimposition of 3D model or animation</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Bread crumbs navigation for ‘current task’</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Directional sound</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Look here</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Stand here</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Shared Pointer</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Record Evidence</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Fitness dashboard</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Visual glyphs (e.g. arrows) and labels</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Activate by Stare Gaze</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Touch points</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Virtual Post-It notes</td>
<td></td>
</tr>
</tbody>
</table>
## 4. Responsibilities

<table>
<thead>
<tr>
<th>UI # and name</th>
<th>Current Status (proof of concept)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dot cursor and object highlight</td>
</tr>
<tr>
<td>2</td>
<td>scanner line</td>
</tr>
<tr>
<td>3</td>
<td>Context menu</td>
</tr>
<tr>
<td>4</td>
<td>Anchored ghost-hand recording</td>
</tr>
<tr>
<td>5</td>
<td>Photo of the situation 'to be' - 'Looking into the future'</td>
</tr>
<tr>
<td>6</td>
<td>video wall</td>
</tr>
<tr>
<td>7</td>
<td>Superimposition of 3D model or animation</td>
</tr>
<tr>
<td>8</td>
<td>Bread crumbs navigation for 'current task'</td>
</tr>
<tr>
<td>9</td>
<td>Directional sound</td>
</tr>
<tr>
<td>10</td>
<td>Look here</td>
</tr>
<tr>
<td>11</td>
<td>Stand here</td>
</tr>
<tr>
<td>13</td>
<td>Record Evidence</td>
</tr>
<tr>
<td>14</td>
<td>Fitness dashboard</td>
</tr>
<tr>
<td>15</td>
<td>Visual glyphs (e.g. arrows) and labels</td>
</tr>
<tr>
<td>16</td>
<td>Activate by StareGaze</td>
</tr>
<tr>
<td>17</td>
<td>Touch points</td>
</tr>
<tr>
<td>Audio recording for think aloud protocol</td>
<td>OUNL</td>
</tr>
</tbody>
</table>
5. Prototyping UI Elements in Unity

Some of the elements for the menus and associated interactions were designed and implemented in a sample Medical Scenario scene with 3D graphics representing bodies of patients and Unity User Interface elements to represent the User Interface Elements that were selected at the Ravensbourne meeting.

The first step was to create an initial draft of the UI mockup connected with the medical use case in order to understand the basic structure of the future application and the user’s path through it.

This prototype can be understood as a storyboard/timeline, where one scene is one cluster of objects (represented as a panel in the beginning of the storyboard and as a patient with overlaid information in the end).

Colours, shapes and additional UI elements were not considered at this stage as the main goal (sketches are on their way). It is not about elaborate design, it is about scenario structure and functions.

Below are the images with detailed descriptions of functions for each scene:

![Figure 22. Overview of Unity UI Mockup](image)

![Figure 23. Unity Logon mockup](image)
Figure 24. Unity Mannequin Interaction

Figure 25. Unity Mannequin Close up

Figure 26. Unity Mannequin Overview
5.1. Functions and transitions in manikin medical sim

5.1.1. First scene with name, institution, submit button

1) a) Log both, generate xml file

2) b) Air-tap ‘submit’ to go to [2]

5.1.2. Scan / Start Scene

a) Scan manikin: air-tap opens scene [3]
   Tick shown once physical/AR coherence is achieved

b) Select Training Scenario: air-tap opens scene [4] (so long as TICK present)

5.1.3. Scan mannequin

- Use markers on the mannequin or object scanning to locate in room
- Use floating text to instruct wearer on what and where to look
- Once all marker found / object well defined - close view, display scene [2]

5.1.4. Show manikin overlaid with AR features

- Display mannequin with locators for various ailments
- Each locator is a coloured circle (2D) with name which:
  - rotates with camera so as to face the wearer
  - Gaze (hover) triggers short description of each to appear
  - Air-tap will start scenario (open relevant scene [5]-[X])

5.1.5. (Scenario specific): Asthma example

- Patient information / History chart at feet

---

**Figure 27.** Unity Mannequin Lung Close up
Scenario title on headboard (shows either 'training mode' or 'examination mode')

Manikin displayed with key areas highlighted
  - Each area is interactive (e.g. trachea, bronchioles, alveoli, diaphragm)

**Training Mode**

- Air-tapping gives expanded view of each area including:
  - Description of normal / abnormal function
  - The test used to assess the function
  - Quantitative metric for measurement (with nominals)
  - Ability to access training record, in the form of elements:
    - Visual: external symptoms shown as AR objects
    - Audio: recording of trainer
    - Haptic: ghost-hands demonstrating action
    - Other diagnostic tool: e.g. pulse, blood pressure, reflex etc.
    - Each element has a number of assessable methods (or procedures)

**Examination Mode**

- Air-tapping brings up expanded view of area requiring:
  - Title ('Patient assessment for ____' (e.g. Asthma))
  - Placeholders for each of the required procedures

5.1.6. **Visual Assessment**

- Trainee should select a particular visible symptom from a list and correctly locate it on the body
- Alternatively, for more diagnostic work, select symptom on body and select which ailment it corresponds to.
  Trainee is learning about a particular ailment which they should they be able to diagnose from what is presented to them? (E.g. of pulmonary embolism suggests the former, whereas a mannequin showing symptoms is suited more to the latter).
- Correct selection and mesh overlay accuracy to determine pass/fail

5.1.7. **Audio**

- Trainee should speak to the patient, repeating a phrase from one given in a list
- If choice is identified, show selected, otherwise ask to repeat (x3)
- Recognition of the spoken phrase should determine pass/fail

5.1.8. **Haptic**

- Trainee should replicate expert's ghost hands
- HoloLens to determine correct placement and action to give pass/fail
- HoloLens can use Leap Motion to match trainee's action to those shown as ghost-hands

5.1.9. **Other Diagnostic Tool:**

- Choice of available tests available to trainee (check pulse, measure blood pressure etc.)
  - X-ray / MRI more for doctors, radiologist
Selections can be made according to profile of user (Nurses, doctors, students?)

- Result and trainee's assessment of data ("this is okay")
- Selection of test, reasonable number given and correct assessment used to give pass/fail

The initial mock-ups were created with a boxed, linear theme. It was discussed that menus in a 3D environment should not have a flat feel but a 3D curved feel. These new designs will be added to this document shortly.

6. Hololens Applications UI Survey

The Microsoft Hololens had a variety of publicly available applications run on it to see the type of UI elements that have been used to date.

During the UX exploration the following observations and recommendations were made:

1) User experience actions have a variety of difficulties in learning as is indicated in Figure 28.(a)

2) It is very important that the main UI should be visible from one general point of view as in Figure 28.(b). It is not effective to make the user walk around all the time to access the relevant functions. At a later date it was decided that we would experiment with a main menu which may move with the user to make sure that the important information is always at the front.

3) As training is being done in a 3D space it makes sense for menus not to be 2d but at least have a 3D feature to them like having a curved menu surround an object as is indicated in Figure 28.(c)

4) Additional menu points can be added through additional menu boxes (Drop down boxes may not be effective but this can be looked into.

5) A menu should be left relevant and simple. If more information is needed to be displayed then layers of menus should be constructed each having different colours based on their content. They can be expanded using a '+' symbol or using a voice command such as the designs below:

6) Menus with voice command activation can be displayed with a different colour background or icon such as the one below:

7) The HoloAnatomy app used little window tags as in Figure 28.(d) and we found this useful. These could also be voice activated.

8) If an object needs to be observed from different angles the menu could move with the angle or view, as with the menu in Figure 28.(e) but this might add cognitive load and needs experimentation.

9) A user pointing with their head is often frustrating as the menu or target object is out of the field of view and while they are trying to hit the field of view. It is important to find the right angle for pointing – as obvious ones are not always efficient or effective.

10) Because of this we found that using the basic voice command is very useful and many apps display voice menus.

11) Common libraries available for Hololens apps can be found in the HoloToolkit UI elements that can be referenced at [https://github.com/Microsoft/HoloToolkit-Unity](https://github.com/Microsoft/HoloToolkit-Unity)
Figure 28. Hololens Applications UI examples. (a) Main control mechanisms (b) Main Interface –all menu items visible from one point of view. (c) Example of curved menu that can embrace object (d) Tagging as in the HoloAnatomy app. (e) Example of a floating menu that moves with the view or object taking from the HoloArchitect app.
Table 3. Main menus

<table>
<thead>
<tr>
<th>What</th>
<th>Details</th>
<th>Why</th>
<th>When</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sound</td>
<td>Gaze Mixing</td>
<td>highlighting objects</td>
<td>almost always, to show the feedback</td>
</tr>
<tr>
<td></td>
<td>Audio Haptics</td>
<td>reactive audio for touchless interactions</td>
<td>activate sound of the objects (additional interaction)</td>
</tr>
<tr>
<td>Immersion</td>
<td></td>
<td>ambient sounds surrounding the user</td>
<td>to create atmosphere</td>
</tr>
<tr>
<td>Gestures</td>
<td>Navigation</td>
<td>to open, select objects/menus</td>
<td>almost always, but when relevant try to use voice commands to have a hands free navigation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Air tap — relatively easy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bloom — easy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Gaze — difficult</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Voice commands — easy</td>
<td></td>
</tr>
<tr>
<td>Manipulation</td>
<td></td>
<td>to move, resize or rotate objects/menus</td>
<td>more specific work with an objects/menus</td>
</tr>
<tr>
<td>Voice commands</td>
<td>Different commands</td>
<td>Open specific menus, do specific tasks</td>
<td>hands free navigation (need to be in a quite env)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>like (take a picture), &quot;type&quot; text</td>
<td></td>
</tr>
<tr>
<td>Gaze</td>
<td>Gaze cursor (dot)</td>
<td>main pointer for hovering over everything in the app</td>
<td>almost always</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Gaze (laser pointer)</td>
<td>shared pointer for group work or detailed navigation</td>
</tr>
</tbody>
</table>

Table 4. Toolbox for structuring the main control mechanisms and UI displays

<table>
<thead>
<tr>
<th>Menu type</th>
<th>Details</th>
<th>When</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body locked display</td>
<td>always connected to your point of view</td>
<td>Menu which is always attached to your point of view and follows your body (main menu activated with a gesture command)</td>
</tr>
<tr>
<td>Anchored display</td>
<td>menus attached to the certain place in space</td>
<td>For example curved menu in the object centered UI when you are working mainly in one place (manikin)</td>
</tr>
<tr>
<td>Floating display</td>
<td>menus floating in space and not cone</td>
<td>Extra menus which you can arrange as you want in the environment</td>
</tr>
<tr>
<td>Pop-up display</td>
<td>additional menus in the hierarchy with extra information</td>
<td>To show additional information from all other menus</td>
</tr>
</tbody>
</table>
7. Conclusions

A fly through tour of these elements is available to watch on a video in the WEKIT Project shared folder in the WP5/Unity UI Video. The actual Unity assets can be imported into Unity from the Unity UI mock-up folder also under WEKIT Project/WP5 folder.

It was found that many UI elements needed to reflect the fact that the workspace was 3D. Many AR and VR applications neglect 3D design of UI elements and use 2D elements such as rectangles and circles as well as icons that are planar.

To enable the brain to encounter more natural interfaces found in a 3D world it was proposed that elements and their arrangements should be 3D. This includes having curved menu panels as well as 3D icons in the work space.
WEKIT project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 687669. http://wekit.eu/