* **Home** 
  + **Project title**: UCL, GOSH and Avanade - Situated Cognition 360 Editor 2021
  + An abstract should include three paragraphs and the following contents
    - Problem statement
      * Building an editor with layers and interactive timeline hotspots that apply situated cognition 360 videos to a handful of human factors training sessions during a series of emergency scenarios.
    - Your solution
      * We solved this problem by building a player and an editor separately using Unity.
      * For the editor, users can add hotspots by right-click on their mouse and open an input panel with a left-click. The input panel allows user to add a name, some description text, an explanatory image and a link to another video to the hotspot. All the information on hotspots will be stored into a json file.
      * For the player, it can read the json file and add the hotspots to its location with all its attached information. The user will be able to watch the video and click on the hotspots for extra information.
    - Achievement and impact
      * We managed to release 2 executable files (for the editor and the player) which allows the user to edit and present the annotated 360 videos.
      * Doctors are able to deliver training sessions more easily with hotspots and 360 videos, and their students can learn more clearly and efficiently.
  + The 8 minutes video to introduce the project and go through the finished functionalities
  + The development team
    - (including the team photos where possible) name, email, role or main contribution of each member. The roles include client liaison, UI design, researcher, programmer, report editor, tester.
      * Kelly Ding - [kelly.ding.19@ucl.ac.uk](mailto:kelly.ding.19@ucl.ac.uk)
        + Client liaison, programmer, report editor, tester
      * Min Wang - [min.wang.19@ucl.ac.uk](mailto:min.wang.19@ucl.ac.uk)
        + Researcher, programmer, report editor, tester
      * Michael Chen - [szu-han.chen.19@ucl.ac.uk](mailto:szu-han.chen.19@ucl.ac.uk)
        + UI design, programmer, report editor, tester
  + Project management
    - Gantt chart (from October 5 2020 to March 24 2021)
  + Links to github repos and release zip (for both source code and deliver package)
* Requirements
  + Project background and Client introduction
    - UCL, GOSH and Avanade are working together on this second iteration of this open source project. This work will be led with the simulation team at GOSH. The project will involve building an editor with layers and interactive timeline hotspots that apply situated cognition 360 videos to a handful of human factors training sessions during a series of emergency scenarios. Speech transcription with Deepspeech or other local speech recognition should be included to generate searchable summary notes that allow you to jump through the timeline and position in 360, as well as generate multilingual text for all users. The open source project will reside in the NHS index at Apperta Foundation.
    - Clients
      * Sheena Visram - [s.visram@ucl.ac.uk](mailto:s.visram@ucl.ac.uk)
      * Dr Dean Mohamedally - [d.mohamedally@ucl.ac.uk](mailto:d.mohamedally@ucl.ac.uk)
      * Fergus E.Kidd - [fergus.e.kidd@avanade.com](mailto:fergus.e.kidd@avanade.com)
      * Chris Lloyd-Jones - [chris.lloyd-jones@avanade.com](mailto:chris.lloyd-jones@avanade.com)
  + Project goals
    - Our goal is to build a 360-degree based video for training workforce.
    - The group last year did a web-based unity player and a very light-weight episodes list in HTML.
    - For this year, we are going to build cross platform electron app for our player and editor. XML or JSON files are used to store data about the 360-video.
      * For the editor, we need a timeline which plays from the start to finish with the video and an interactive HTML layer on the top of the 360-video.
      * The coordinate space is mapped to the video field: if you start with a particular viewpoint, that is the starting origin and so anyway you rotate in the 360-video, you would have that mapping of that layer.
      * Pop up textboxes, images, other videos
      * Event trigger: point trigger, area trigger; open the HTML layer with the interactive text, or take you to another video file
      * Points would exist in the timeline (as you play the video)
      * Special coordinates for the layer of interactive text and images will be static in time (always be there), or manipulated in the timeline
      * Tree diagram of the video and the json/xml file: if you click on a region, it will open another video
      * Tree structure: parse the video chain and load into sequence
      * Scenario: one decision, use a for something; another decision, use b for something and load another video.
      * User friendly for clinician
      * AGPL version 3 license
* Requirement gathering
  + How did you collect the requirements?
    - We had a meeting with our clients, and we take down the requirements they have.
  + Mind map about requirements
    - <https://www.xmind.net/m/TfmL8Z/>
  + Sent the summary mind map to clients for feedback
* Personas
  + HCI PowerPoint
  + Scenario from Sheena
    - Imagine these are 360 videos and being used in place of face to face education.
    - The scenario: My clinical background is as a Dietitian and I specialised in a field known as artificial feeding. This is where a patient receives nutrition via a feeding tube as they are unable to eat and drink due to an unsafe swallow. This can either be via a tube into their stomach (NG feeding is an example), or via a tube that delivers nutrients into their blood stream (this is known as TPN).
* Use cases (HCI PowerPoint)
  + Use case diagram
    - Import video – watch video and add hotspots – save to json file.
    - Load player with package – watch the annotated video
  + List of use cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | User Use Case | Actor | Description | Main flow | Result |
| ED1 | Browse the video file | Doctor | Open a video file inside the computer. | Click on the “file” logo on the top left corner and browse a file. | A 360-video file is opened. |
| ED2 | Play/Pause the video, jump with timeline | Doctor | Basic video player functions including play/pause, and jumping into different time spots using the timeline. | Click the pause/play button.  Click the coordinate on the red timeline which indicates the rough location that the user wants to jump to. | Video played  Paused/  Jumped to another location according to time. |
| ED3 | Add/Delete a hotspot | Doctor | Descriptive hotspots can be added/deleted on specific locations on the video screen. | Add: By right-click on the location that we need to add a hotspot.  Delete:  ①Open the input panel of the hotspot that needs deleted by left-clicking on it.  ②Click the delete button on the top right corner of the input panel. | Hotspot is added/ deleted. |
| ED4 | Add more information to the hotspot | Doctor | Explanatory information, including title, descriptive text, image and another video can be added to the hotspot. | Adding title: Type in the input textbox(which has Green-blue background).  Adding text: Type in the white input textbox.  Adding photo: Click the “Choose Photo” button and browse a photo. The photo URL will be appear on the right hand side of the button.  Adding Video:  ①Click the “Change video” button to browse for the video. The new video URL will be printed on the right-hand side of the button.  ②Click the “play video” button to play the video we selected. Click the “back to previous video” button to jump back to the previous video.  Click the “save” button to store the information added. | Add information to the hotspot. |
| ED5 | Store information to json file. | Doctor | Store all the hotspots and the information attached. | Click the second button on the left column. | Information saved. |

* MosCow List
  + A table for function requirements
  + Another table for non-functional requirements: performance, security, usability, open source, maintainability, extensibility, etc.
* Research
  + Related Projects Review
  + — For each of the similar projects, please list the project name, main features, and what you can learn from this existing application.
    - The project done by the group last year.
    - <https://github.com/Apperta-IXN-for-the-NHS/IXN_Learning360>
    - The group from last year built a web-based player, which the player is done in unity and the file management part is done with web utilities.
  + Technology Review
  + Please compare the possible solutions, describe what you choose, and explain why.
    - Possible solution A: Just like the project done last year, we build the web app on electron and attach a unity player/editor to it.
      * Pros: The web-based website can be deployed on different machines.
      * Cons: There might be bugs involved when merging the 2 systems.
    - Possible solution B: All parts of the project are done in Unity.
      * Pros: Unity is open and flexible to changes.
      * Cons: Unity itself is very complex and has different versions, so project built in old version of unity will have bugs when it is deployed into a newer version of unity.
  + Please compare the possible programming languages, frameworks, libraries, APIs, describe what you choose, and explain why.
    - Unity can only use C# as its programming language.
  + A summary of your technical decisions
    - We decided to implement the whole project inside unity, so the project will be safer compared to web-based. Also, many NHS videos has its own GDPR regulation with it so it’s better to hold everything locally.
  + References
* UI Design
  + Design Principles (e.g. simplicity, consistency, visibility, feedback, tolerance, etc.)
  + Hand-drawn sketches
  + Online interactive wireframe (High Fidelity wireframe e.g. with Figma)
  + Prototypes(from HCI ppt)
  + Player UI
  + Editor UI
  + File type we created
* System Design
  + System architecture diagram and a brief description of each component.
  + Site Map (if applicable)
  + Sequence Diagram (if applicable)
  + Class Diagram (if applicable)
  + Data storage (if applicable)
* Implementation
* Testing
* Evaluation
* Appendices
* Development Blog