# TODO

* P046
  + POI fixes
    - Terras POI: too low
    - POI rechtergevel: verplaats naar hoekje van terras, en fix hoogte.
    - POI leefruimte: plaats in midden van leefruimte.
    - Create POI:
      * “VooraanRechts”
      * WC?
  + SKP fixes
    - Texture/material names
    - Add WC pot
    - Add kooking plates
    - Add dampkap
    - Add kraantjes, douchekop
    - Fix height of zetel zwart 1P in leefruimte: te hoog.
    - Fix orientatie van kastdeur in slaapkamer.
* Maquette mode: Implement showing/hiding parts of the model
* Implement help menu (HUD menu, attached either to front of head or controller(s)) explaining controls.
* Walkthrough mode: Implement showing the name of active POI
  + World-space Tekst GameObject
    - lokated in fixed location in front of user, or
    - attached to front of head?
* Implement aplication states:
  + Previewing While previewing a project. Input is active. Model is visible
  + Loading: While (un)loading a project to preview. All input is blocked in order to prevent requests to activate Project or POI,....
    - First fade out
    - Then show title of new project while async loading it
    - Then fade in again
* Implement toggling lights on/off
* Implement custom(prettier) skyboxes
* Tweak unity GFX settings
  + Reflection
  + Ambient occlusion
  + ???
* Implement Edit mode:
  + If active, user can define, edit, save POI for a given project.
    - Will we do this? Not usefull without making POI transferable between builds which requires additional development… Only do when we have a workflow that needs it! For now prepare projects in Unity editor, which is fine.
* Write out a ‘How to prepare Sketchup for Archi-VR import’ document
  + Material names:
    - Texture name == Material name
    - No spaces
    - -> Use Pascal Casing? (eg. VloerTegel512x512, Boom2D256x512) forxture/material names
    - Use only JPG? (or is png also fine?)
  + Face orientation should be consistent and correct. Tip: Use Edit/Face Style/Monochrome to check face orientation.
  + Use outliner to divide model into the entities that should be visiblility-togglable in Maquette mode. Eg:
    - Model
      * Context (Street, garden,…)
      * Outside (Gevel)
      * Roof
      * Basement2
      * Basement 1
      * Groundfloor
      * Floor1
      * Floor2
      * …
* Write out a ‘How to create an ArchiVR project’ document.
  + Hierarchy:
    - Project
      * Model
      * POI
      * Lighting
  + Copy-paste-rename an existing ArchiVR project
  + Open it
  + Remove the content of the ‘Model’ node
  + (create a new folder under assets/Projects/ProjectName to store the necessary assets fort he ArchiVR project in.
  + Import the assets
  + Drag in models under the ‘Model’ node.
  + Adjust the POI.
    - Add/Edit/Remove POI to match the project.
    - Tip: Create additional POI by copy/paste/rename existing POI. (start from an original on the same level)
    - Relocate the a POI by selecting it (not its children!), and dragging it to the correct position/orientation.
  + Adjust the lighting
    - Add/Edit/Remove lights to match the project.
    - Tip: Create additional Light by copy/paste/rename existing lights. (start from an original on the same level)
* When teleporting, make sure the user ends up in the POI (IE offset trackingspace, so that not center of tracking space is == POI, but location of head.) this will prevent the sporadic ‘awkward teleporting’, eg when after a teleport, the user ends up on top of a stair, inside a wall, outside the gevel in thin air, etc…)
* Implement OVR driven input.

# DONE

* Implemented Unity Input (buttons and kb).
* Fix texture load from SKP files.
* Implement loading projects from separate scenes asynchronously
* Implement immersion modes: Walkthrough and Maquette
* Set menuMode default to ‘None’