Give the name of the Course

Introduction

Short paragraph about the objectives of this course

Location

Describe the area where this course is located, including topography, terrain, buildings.

If relevant, mention how the individual subtopic areas are included. Notice that the area of a subtopic can be a part of the area of the topic, but it could also be a separate area, accessible by teleporting (after clicking some button/link).

You can include images. A top-view sketch with the subareas can be very helpful.

List of Subtopics

Fill the table with the individual subtopics of this Course. Specify if each subtopic is located in the same area (Subarea) or a separate one (Separate)

Possible Location: Subarea / Separate Area

SUBTOPIC NAME	LOCATION

Scenario / Pedagogical Methods

Describe the overall scenario. Mention how will the individual subtopic scenarios will be connected (e.g order, prerequisites) with one another.

Include details about any rewards, collectables, puzzle pieces or other items and how can the user receive them.

List of Presentations

Specify the Theory Presentations. There is a separate file template for the presentations.

Presentation Filename	Presentation Title	Location

List of Activities

Number	Activity Title
1	
2	
3	
4	

For each activity, copy and fill the following Table:

Activity ID	Give the number of the activity
Activity Name	Give the name of the activity
List of Individual Objects, buildings and NPC Characters	Give a list of objects that are necessary for the activity • object1 • object2 • NPC Character1 • Building1

Location	Describe the location:
Description	Describe in detail the activity, including the individual objects and how the user interacts with them.
Text:	Provide the text used in the scenarios (e.g. Dialogues, Feedback Messages etc.).
Estimated Duration	How much time the user will be engaged in the activity?

References

Add any links to web pages or videos that may be helpful for implementation

Assessment Questions Template

Question #: 1	Copy table as many times as needed	
Text of question:	**1. Enter here the text of the question**	
Potential Answer A	Text of Answer	**Enter here the text of the first answer that the user can choose**
	Is Correct (Yes/No)	**Enter Yes if this answer is correct.**
	Feedback Message	**Enter the text displayed when the user selects this option**
Potential Answer B	Text of Answer	
	Is Correct (Yes/No)	
	Feedback Message	
Potential Answer C	Text of Answer	
	Is Correct (Yes/No)	
	Feedback Message	

Question #: 2	Copy table as many times as needed	
Text of question:	**2. Enter here the text of the question**	
Potential Answer A	Text of Answer	

	Is Correct (Yes/No)	
	Feedback Message	
Potential Answer B	Text of Answer	
	Is Correct (Yes/No)	
	Feedback Message	
Potential Answer C	Text of Answer	
	Is Correct (Yes/No)	
	Feedback Message	

Types Of Learning Activities

1. Read a panel, with a presentation
2. Pick Up / Receive a notecard and read it
3. Have a scripted conversation with an NPC Character
4. View and interact with a panel that displays a web page
5. Participate in a 3D activity by interacting with 3D objects
6. Complete a Quiz with multiple choice questions

ANNEX: Events and Actions in OpenSimulator

1. Read a panel, with a presentation

A presentation is displayed on a specific face of a 3D object (e.g on a blackboard, or a TV screen). The student can click on the Object/screen to load the next slide or we can have left/right buttons to go to the next or previous slide.

The template for displaying this activity is a simple Powerpoint template. Each presentation will be displayed on a separate panel in the 3D World. We should try to keep these presentations short. We don't want these to be too detailed and tiring for the students. Also keep in mind that we will have to translate the text later. It is very possible that we will have to transform the presentations to images before importing them to the world. So we should avoid links or animations.

The author of the scenario should also describe where the panel is located in the context of the area that hosts the course.

2. Pick Up / Receive a notecard and read it

Notecards are simple text files that users can have in their Inventory (folder structure) in OpenSim/Second Life and open whenever they want. We can use scripts to control how the scripts are given to the avatar (e.g when they click a leaflet object or given by NPC characters).

The text cannot be formated so the template for describing a notecard can be a simple .txt file.

However we can include in the notecard various links such as:
- links to other notecards (when the user clicks the link, the other notecard opens)
- links to images (when the user clicks the link, the image is displayed)

- landmark links (when the user clicks the link, they can store or teleport to a specific location in the 3D World)

Notecards are a good way to provide theory to the student, especially if there is a lot of text that they need to study.

3. Have a scripted conversation with an NPC Character

The NPC avatar is located in a specific location in the 3D World. When the student approaches the NPC (or some other specific event takes place) the NPC character approaches the character and talks with him with text messages displayed on screen.

The conversation with the NPC is scripted, so the student can in each step choose from one of the available dialogue options. Based on the option selected, the NPC gives a relevant answer and the student is presented with the next dialogue options. In other words, the dialogue will work like a series of linked multiple choice questions.

The template for describing the activity will be a presentation, where each slide describes a specific step of the conversation, that is the text that the NPC character says and the list of possible answers for the student to choose. Each option will have a unique number, and will point to another slide of the template presentation.

Please bear in mind that the text we can use for each dialogue step is limited to 514 characters (including the possible answers), so the NPC character's answers have to be really short. If you want the character to give large amount of information about something we can have them giving a Notecard file to the user that they can read later, or directing them towards a presentation panel.

Apart from the text that the character says in each step, we can have the NPC character do specific scripted actions such as an animation (e.g the NPC character points towards a location, or walks a couple of meters). There are free libraries with animations that we can use to find specific commonly used animations. We can also use special software to create a custom animation to do something specific but it is a difficult and time consuming process.

We can also have other actions taking place in each step of the conversation. Many of these actions are described in "5. Participate in a 3D activity by interacting with 3D objects". Here are some examples:

The NPC character gives a 3D Object to the student when they give a specific answer (e.g as a reward for solving a riddle).

The NPC character gives a different answer to a question depending on whether the student has completed some previous activity. For example an NPC character gives some instructions to the student to fetch a specific item. If the student returns to the NPC character and has not picked up the item yet , the NPC character tells him to search for it (or gives some additional clue). If the student has found and picked up the item, the NPC character proceeds with the conversation.

4. View and interact with a panel that displays a web page

This is similar to the presentation panel described in 1, but instead of a static presentation we display a web page.

We can use technologies like HTML, Javascript, Flash, PHP, JSP etc. to create interactive pages. We can embed images, gif animations ,videos etc.

This is our best option to display some interactive activity or content that is difficult to implement in 3D, so we use a 2D webpage instead.

It is difficult to associate the visitor of the webpage with the avatar that interacted with the panel, so we will probably NOT be able to combine this activity with others in the 3D world. We can use this type of activity for self assessment purposes but we cannot check in the 3D world whether a specific avatar has completed the activity inside the web page or not.

It is difficult to provide a specific template for this type of activity. The authors have to describe the functionality of the webpage as best as they can using text and images.

5. Participate in a 3D activity by interacting with 3D objects

This is the most complicated type of activity. It includes various distinct 3D Objects that the student can interact with as part of an activity (e.g. an experiment).

It is difficult to provide a specific template. The author should first try to describe in detail each individual object or the parts that it comprises of, using text and images. The author should also include some description of the location where the activity takes place (e.g indoors, outdoors, on a desk).

Most importantly the author has to describe the functionality of each individual object and how it is connected to the others. For example: "When the student clicks/touches object A, then some text is displayed and object B moves higher and it's color turns to red"

In general the functionality can be described by a series of EVENT, ACTIONS pairs. When the EVENT takes place then ACTIONS are executed.

I don't want to put concrete limitations but in order to help you understand how Opensim works, i have included an ANNEX that describes EVENTS and ACTIONS. Please do not give focus on the technical details. I only want to provide some ideas of things that are possible in the 3D World.

6. Complete a Quiz with multiple choice questions

Multiple choice questions with specific answers to choose from are the best way for assessment.

Please note that we can also have assessment in other activities such as 3 (the NPC character can make the questions and expect the correct answer) , 5 (multiple choice questions can be part of a 3D activity) and 4 (multiple choice questions can be used in the web pages).

However I have included this type of activity as a separate one, because it seems like a common one for all courses, and it would be useful to provide a specific template for preparing them.

We can have a couple of multiple choice questions at the end of every scenario (learning topic) to test what the student has learned.

Finally we can collect all these questions and create a Trivia Machine with content from all courses, where the students will try to achieve a high score, competing with their classmates.

ANNEX: Events and Actions in OpenSimulator

In general, objects have different **states**. While in a state the object may do some **actions** (e.g rotate or display a label above), and may wait for a particular **event** that will trigger it to do some other actions or change state. As an example a lamp is initially in a state called "off" and waits for the event of some user clicking/touching it. When a user clicks it then it executes an action that makes the lamp turn on (emits light) and then changes to another state called "on". While on state "on" the object waits for a user to click it so it turns off and returns to state "off".

GENERAL EVENTS AND ACTIONS

Events:
- User touches (clicks) an object.
- User wears or equips/attaches an object (right click on an object in his inventory, wear in available slot (e.g in his right hand)).
- User takes a copy of an object (right click, take copy).
- User approaches an object.
- User collides with an object
- User selects an option in a dialogue menu or types some text in a dialogue text box
- User writes some message in a particular chat channel (e.g typing "/24 hello" in chat will send the "hello" message to channel 24)
- Event repeats every X seconds.

Actions:
- Object changes position, size, color, opacity, texture, rotates etc
- The texture assigned to the object moves/animates (e.g running water)
- Object "says" some text in chat or plays a sound
- Label/text is displayed on top of an object.
- Object stops/sleeps X seconds between two actions.
- User avatar is teleported to another position in the region or in another region.
- The walking speed of the avatar is reduced/increased.
- An object is given to the avatar (becomes available in his inventory)

Examples:

- A door is initially closed and opens (<u>action</u>: rotate) when the user clicks (<u>event</u>: touch) on it (the door can be locked so user needs to first make some action e.g type a password.
- The user is asked to select a color in a dialogue menu (<u>event</u>: option in dialogue menu) and the object is then painted with it (<u>action</u>: change color).
- An elevator moves to another floor (<u>action</u>: move to target) when the user clicks (<u>event</u>: touch) a button.
- On a presentation screen, the next slide is loaded (<u>action</u>: change texture) when user clicks Next button (<u>event</u>: touch)
- On a presentation screen, the next slide is loaded automatically every 10 seconds.
- When the user approaches an object (or area) he gets a text message.

ACTIONS AND EVENTS ASSOCIATED WITH NPC CHARACTERS

Actions:
- NPC character is created at target position
- NPC walks (or runs, flies or uses any other animation) to a target position.
- NPC rotates to face towards a direction.
- NPC does some animation (wave, point, dance, attack etc).
- NPC saying a message (text) in nearby chat that appears on screen.
- NPC randomly wanders around a point.
- NPC follows the user-avatar.

Events:
NPC executes one or more of the above actions after the user...
- touches an item
- writes something to chat in a particular CHANNEL   e.g "/5 hello"
- selects an option from a dialogue box
- writes specific text in a dialogue text box
- approaches (or collides) some object
- any other event described above.

Application Example: The NPC appears when the user-avatar approaches an area or clicks some object (e.g a closed Door). The NPC avatar waves and starts 'talking' with text messages on screen. The NPC character asks the user to follow him and starts walking a predetermined path around the region. The NPC character stops in various positions and points to something, speaking all the while with messages and after a while continues walking to another position. The NPC character stops and asks the user some question. A dialogue box with possible answers appears and the user selects one. Alternatively a text box appears and the user types his answer. Alternatively the user clicks a particular object. When the user gives the correct

answer or does the necessary actions the NPC avatar continues (e.g moves to the next position or gives a reward to the user).

An example of a simple NPC character acting as a guide:
https://www.youtube.com/watch?v=3zHfMg74AGA

SPECIAL SCRIPTS

Vehicle: The character sits in the object and can drive around with it (e.g driving a car)

Gun: The character throws specified objects (e.g bullets or basketballs) towards a direction, with adjustable speed.

Particles: The object omits small particles. The script is parameterized allowing to change the shape and number of the particles, the direction they move and the duration before they disappear (e.g smoke, laser beam, emission of electrons, butterflies, snowflakes, rain, leaves).

PHYSICS ENGINE

An object can be set as 'physical' so gravity affects it. The object will fall to the ground if placed in the air, will slide down a slope or move when the user or another item collides with it. The user can also click and drag this item to push/pull it towards a direction.

Example: It is relatively easy to set up a bowling game. The pins and the bowling ball are set as physicals objects. The pins are initially placed vertically so they stay up. The user drags the ball towards the direction of the placed pins. When the ball collides with the pins, the pins affected will drop.

HEADS UP DISPLAYS

http://en.wikipedia.org/wiki/Head-up_display

We can have custom objects, textures and text appearing on a specific part of the screen. The user receives a HUD object in his inventory. He wears it (double click for wear, or right click -> Attach Hud -> Preferred Area) and the HUD display appears on his screen. The HUD is very useful for storing data about the user's interaction in the game and providing relevant information.

Two examples are presented in detail next:
- Collection of Tokens
- Collection of Puzzle Pieces

HUD EXAMPLE: TOKEN COLLECTION

The user wears an item that displays a small window on his screen with a score (Tokens collected and Prizes earned).

While wearing this item he gains token points when he clicks (collects) a specified item or does some specific action. Some objects may award more points than others.

Also while wearing this item he may lose token points if he triggers some traps (approaching or touching a specified object).

Some tokens may only be awarded only if a condition is met, e.g the user wears or has equipped a particular object/tool or if he has previously earned a specific reward. As an example the user clicks on a broken bottle and loses points because he was cut. If the user has previously equipped gloves then he clicks in the broken bottle and is awarded tokens.
In general any event mentioned previously may trigger as one of the executed actions to increase or decrease the token score of the user.

HUD EXAMPLE: PUZZLE PIECES / PIECES OF MAP

The user wears the HUD item and then proceeds to explore the world and solve some quizzes/quests. When he successfully completes a quest, he is awarded with a puzzle/map piece which appears on screen. When all quests are completed the complete Map is displayed on his screen showing the location of a hidden chamber/treasure..