Space Explorer

John Cho

Lamar Taylor

Antonio Brown

Goal

- virtual environment to simulate the exploration of our solar system
- 3 Modules
  - ISS Internal Exploration
  - ISS External Space Walk
  - Solar System Exploration

Objectives

- navigate a space craft through the solar system,
- move around inside the space craft, and complete simple task
- leave the spacecraft to conduct a space walk

Modeling

- Downloaded multiple 3d Models with Texture from NASA web site
- Some challenges encountered when importing downloaded models
  - Importing Texture were not consistent
  - Some models were too big to be incorporated to the main scene

Vision

- Planets
  - Modeled from scratch
    - Used spheres and textures to build planet models
- Star Maps
  - Created environment map textures to be used in the scene using graphic tools

User Interaction

- Menus are used to switch from one module to another
- Keyboard keys were used to control avatars
- Mouse gestures were used to change view point in some module

Sensors

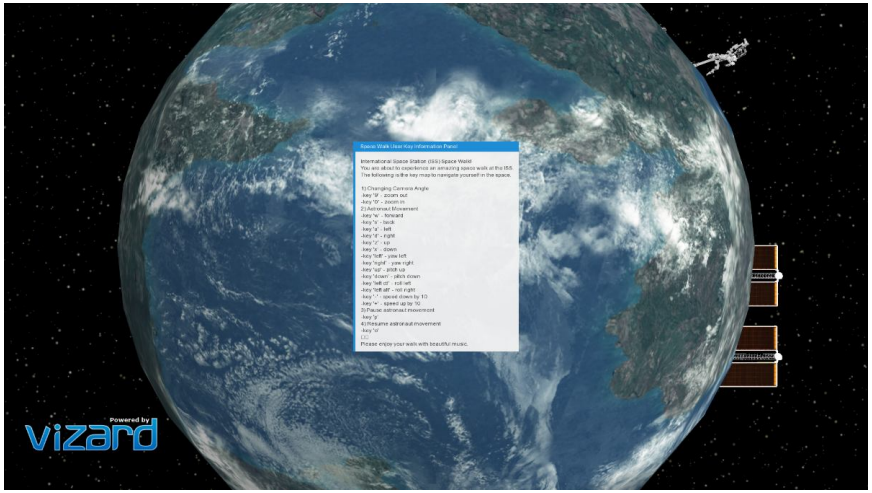
- Proximity sensors were used to detect proxy events between sensors and targets

Animations

- Animations were used to simulate solar particles
- Animations were used for movement of planets
- Animations were used to simulate solar wave/radiations

Space Walk Module

- **Purpose**: The user will be able to experience a simulation of Space Walk using keyboard controls. The scale of ISS model is proportional to the scale of the astronaut model.

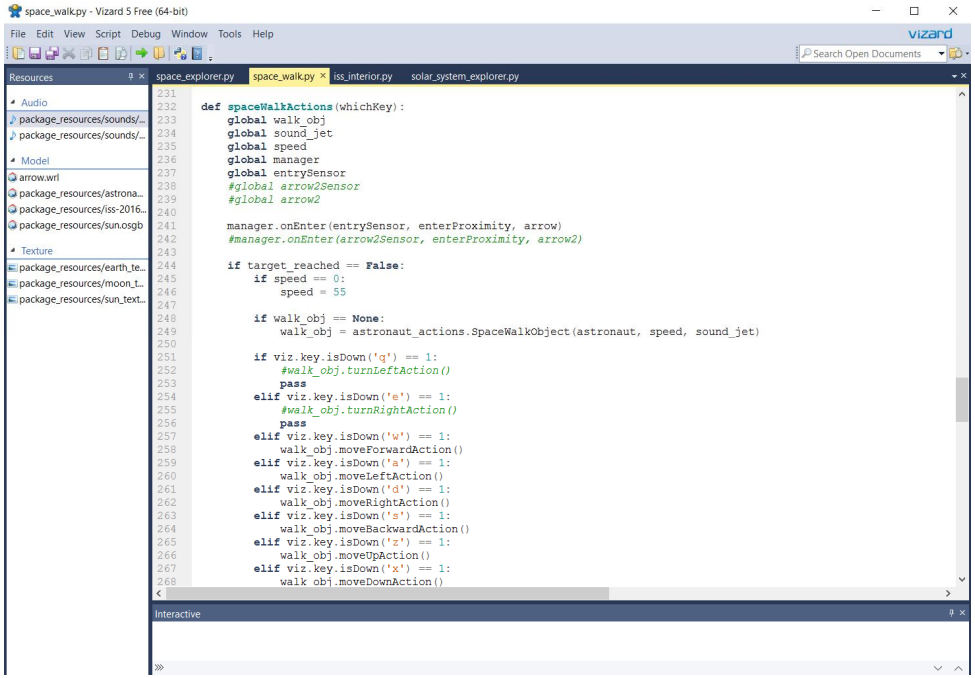


Space Walk Module – Key features

1) The user can control the astronaut using various keyboard keys. The key map of the module is as the following:

Key	Action	Description
'9'	Camera Zoom out	Main view is zoomed out exposing astronaut's back side
'0'	Camera Zoom in	Main view is zoomed in as 1st person view
'w'	Move Forward	Astronaut moves forward
's'	Move Backward	Astronaut moves backward
'a'	Move Left	Astronaut moves left
'd'	Move Right	Astronaut moves right
'z'	Move Up	Astronaut moves up
'x'	Move Down	Astronaut moves down
'left arrow'	Yaw Left	Astronaut yaw left
'right arrow'	Yaw Right	Astronaut yaw right
'up arrow'	Pitch Up	Astronaut pitch up
'down arrow'	Pitch Down	Astronaut pitch down
'left ctl'	Roll Left	Astronaut roll left
'left alt'	Roll Right	Astronaut roll right
'pad -'	Speed down by 10	Astronaut's movement speed is up by 10 unit
'pad +'	Speed up by 10	Astronaut's movement speed is down by 10 unit
'p'	Pause Astronaut actions	Astronaut's actions are paused
'o'	Resume Astronaut actions	Astronaut's actions are resumed

Space Walk Module – Key features -IDE



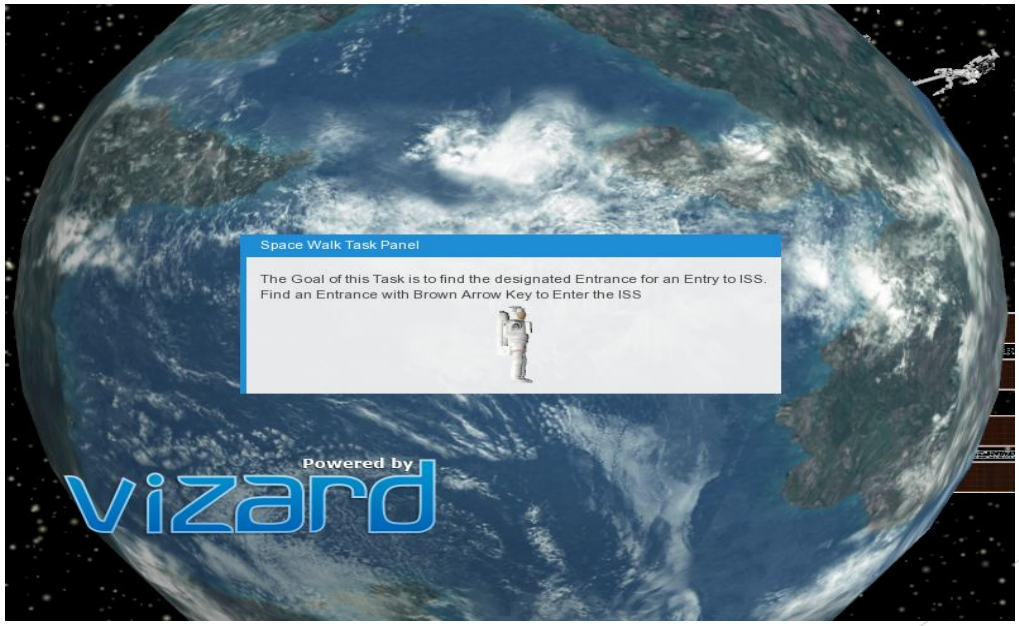
Space Walk Module – Key features

2) Beautiful background music is played while the user is experiencing the simulation

3) When movement keys are pressed, sound of air jet simulated sound is played

Space Walk Module – Key features

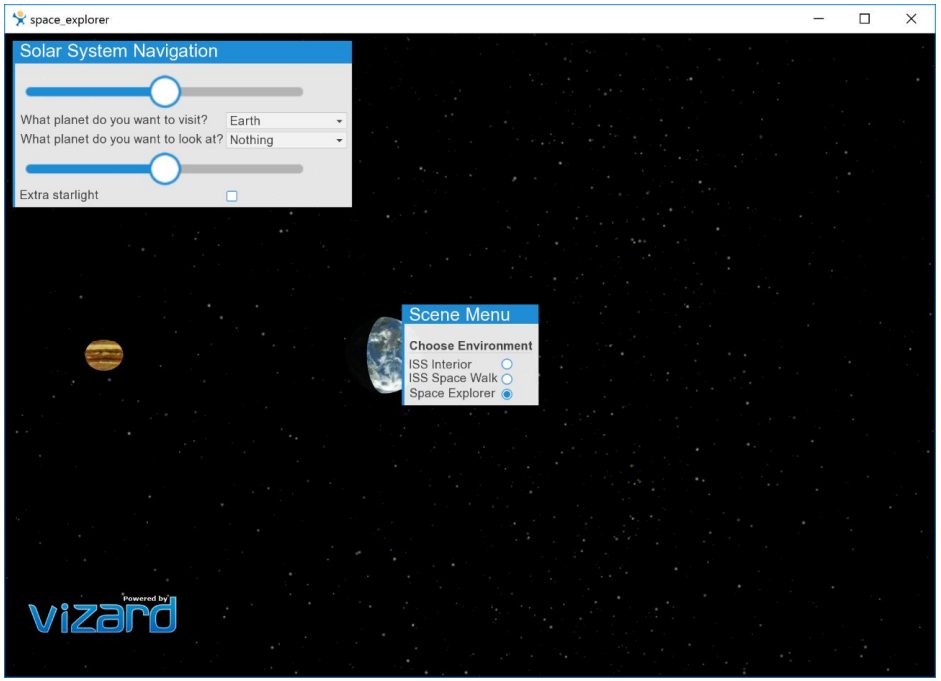
4) A task is given to find an entry point to the internal of ISS. It is marked with yellow down arrow key.



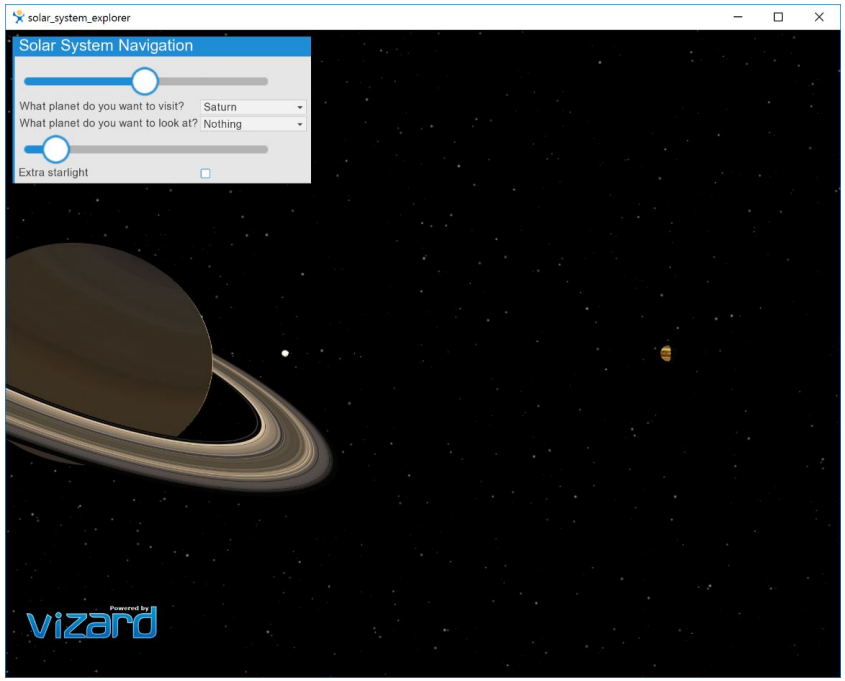
Solar System Exploration

- Purpose: Explore different planet system within our solar system
- Key Features:
  - Users can navigate to different planet using the menu system
  - Users can view different planet regardless of their location
  - Simulate solar particles
  - Simulate movement of planets

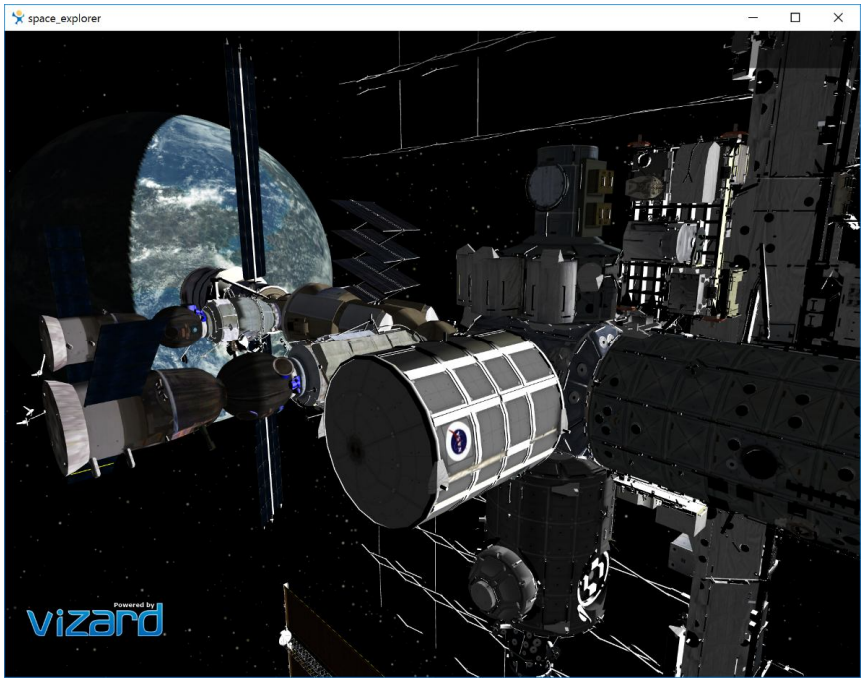
Solar System Exploration – screen captures



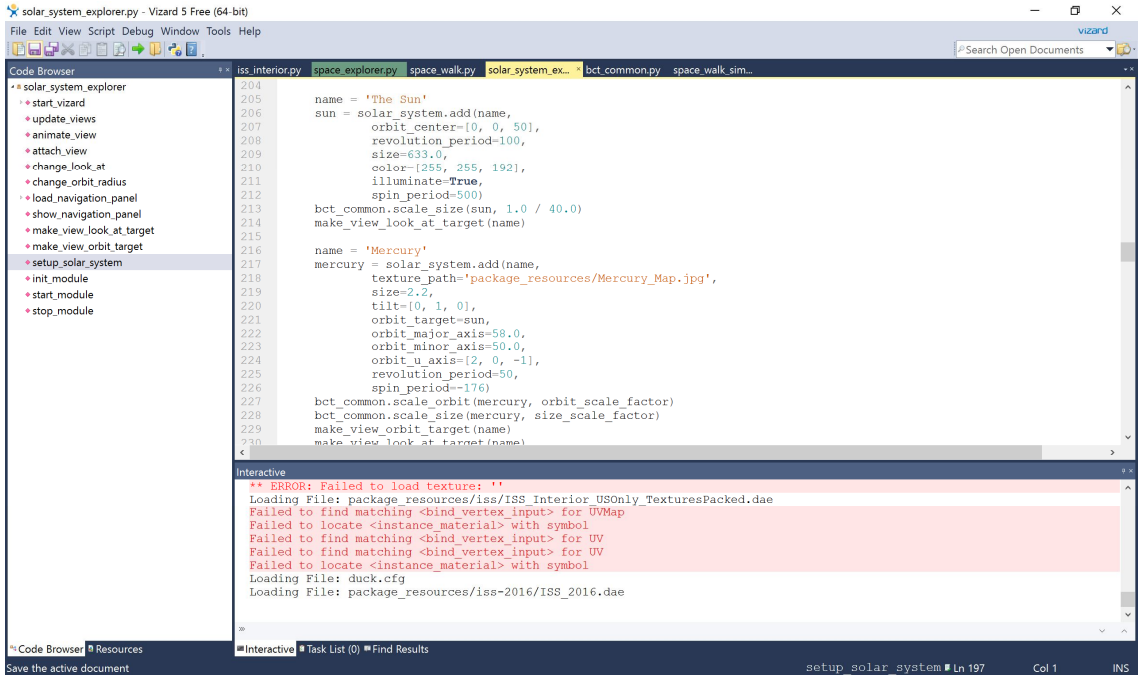
Solar System Exploration – screen captures



Solar System Exploration – screen captures



Solar System Exploration – screen captures



ISS Interior Exploration

- Purpose: Allows users to navigate within the ISS.
- Key features:
  - User task to locate ducks within ISS environment given predefined time limits
  - Keyboard and mouse keys and gestures were utilized to navigate

ISS Interior Exploration

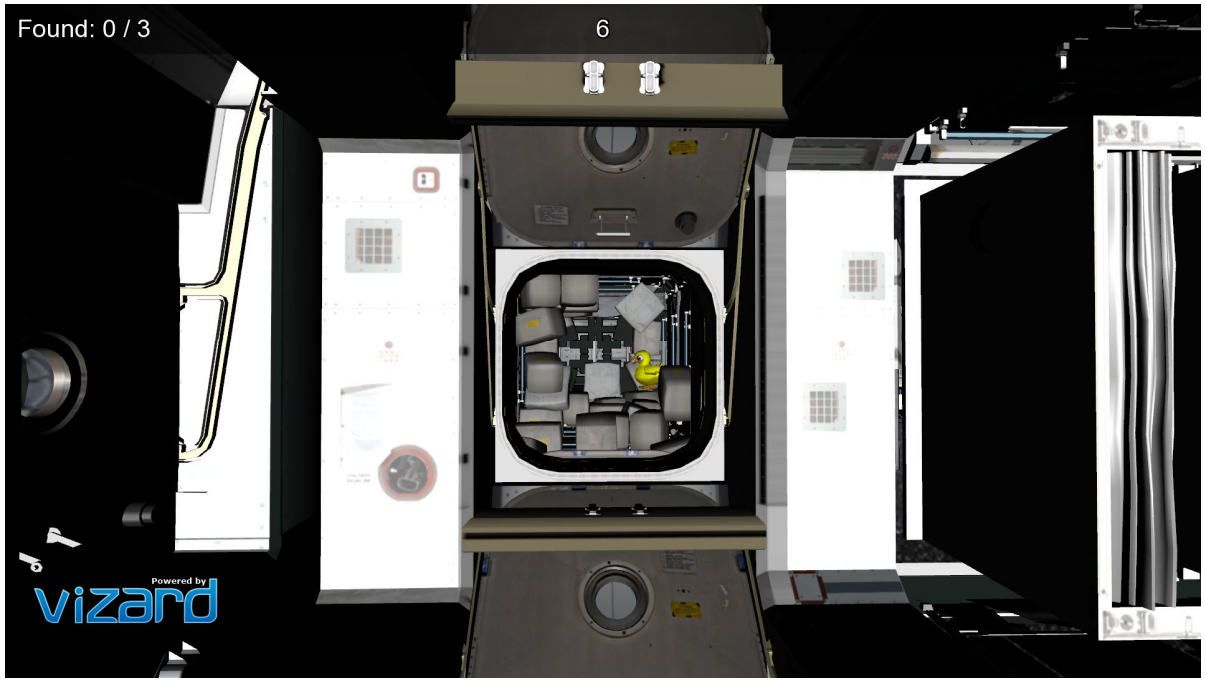
space_explorer.py | space_walk.py | iss_interior.py | solar_system_explorer.py

```
    vizfx.postprocess.addEffect(gray_effect)

def DisplayInstructionsTask():
    """Task that display instructions and waits for keypress to continue"""
    global pigeon
    panel = vizinfo.InfoPanel(INSTRUCTIONS, align=viz.ALIGN_CENTER, fontSize=22, icon=False, key=None)
    pigeonClone = pigeon.clone(scale=[200]*3)
    pigeonClone.addAction(vizact.spin(0, 1, 0, 45))
    pigeonClone.enable(viz.DEPTH_TEST, op=viz.OP_ROOT)
    panel.addItem(pigeonClone, align=viz.ALIGN_CENTER)
    yield viztask.waitKeyDown(' ')
    panel.remove()

def TrialCountDownTask():
    """Task that count downs to time limit for trial"""

    # Action for text fading out
    text_fade = vizact.parallel(
        vizact.fadeTo(0,time=0.8, interpolate=vizact.easeOut),
        vizact.sizeTo([1.5, 1.5, 1.0], time=0.8, interpolate=vizact.easeOut)
    )

    # Reset time text
    global time_text
    time_text.clearActions()
    time_text.alpha(1.0)
    time_text.color(viz.WHITE)
    time_text.setScale([1, 1, 1])
    time_text.message(str(int(TRIAL_DURATION)))

    # Countdown from time limit
    start_time = viz.getFrameTime()
    last_remain = int(TRIAL_DURATION)
    time_played = viz.getFrameTime() - start_time
    #while (viz.getFrameTime() - start_time) < TRIAL_DURATION:
    global game_is_running
    while time_played < TRIAL_DURATION:
        if game_is_running:
```

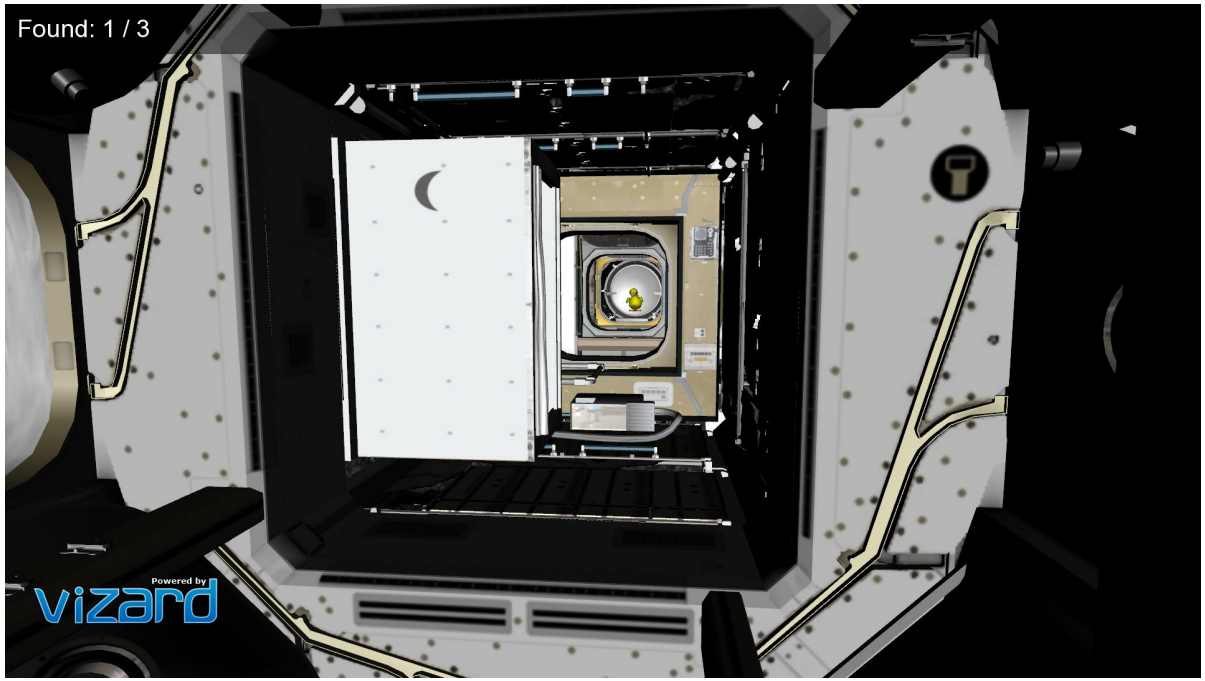
Interactive

ISS Interior Exploration – screen captures

ISS Interior Exploration – screen captures



ISS Interior Exploration – screen captures



ISS Interior Exploration – screen captures