
Immersive Virtual Tour

Software Design Document

CS243: Software Engineering

Group 3

Abhishek Suryavanshi (160101009)

Nitesh Jindal (160101084)

Ameya Daigavane (160101082)

Index

Introduction	3
1.1 Purpose	3
1.2 Scope	3
Use Cases	4
U1: Start Virtual Tour	5
U2: Virtual World Movement	6
U3: Looking Around	7
U4: Interaction	7
U5: Change Settings	8
U6: Exit Tour	8
Data Flow Diagram: Level 0	9
Data Flow Diagram: Level 1	10
Data Dictionary (for DFD Levels 0 and 1)	11
Data Flow Diagram: Level 2	12
Initialize Virtual Tour	12
Change Settings	12
Player State Modifier	13
Sound Generator	14
Interaction Handler	15
Virtual World Generator	16
Process Decomposition Diagram	17
ER Diagram	18

1. Introduction

1.1 Purpose

The purpose of this document is to describe the design and implementation of the Immersive Virtual Tour Android application, that will enable users to take a virtual-reality tour of the Department of Computer Science and Engineering, IIT Guwahati. This document is intended for both the developers and users of the system.

1.2 Scope

The scope of this document is to explain the design of the software system - by identifying the Use Cases, and charting the Data Flow Diagram and the Entity-Relationship Diagram. The expected software system will be a Immersive Virtual Tour system intended for people who wish to use a Virtual Reality Headset to tour the Department.

2. Use Cases

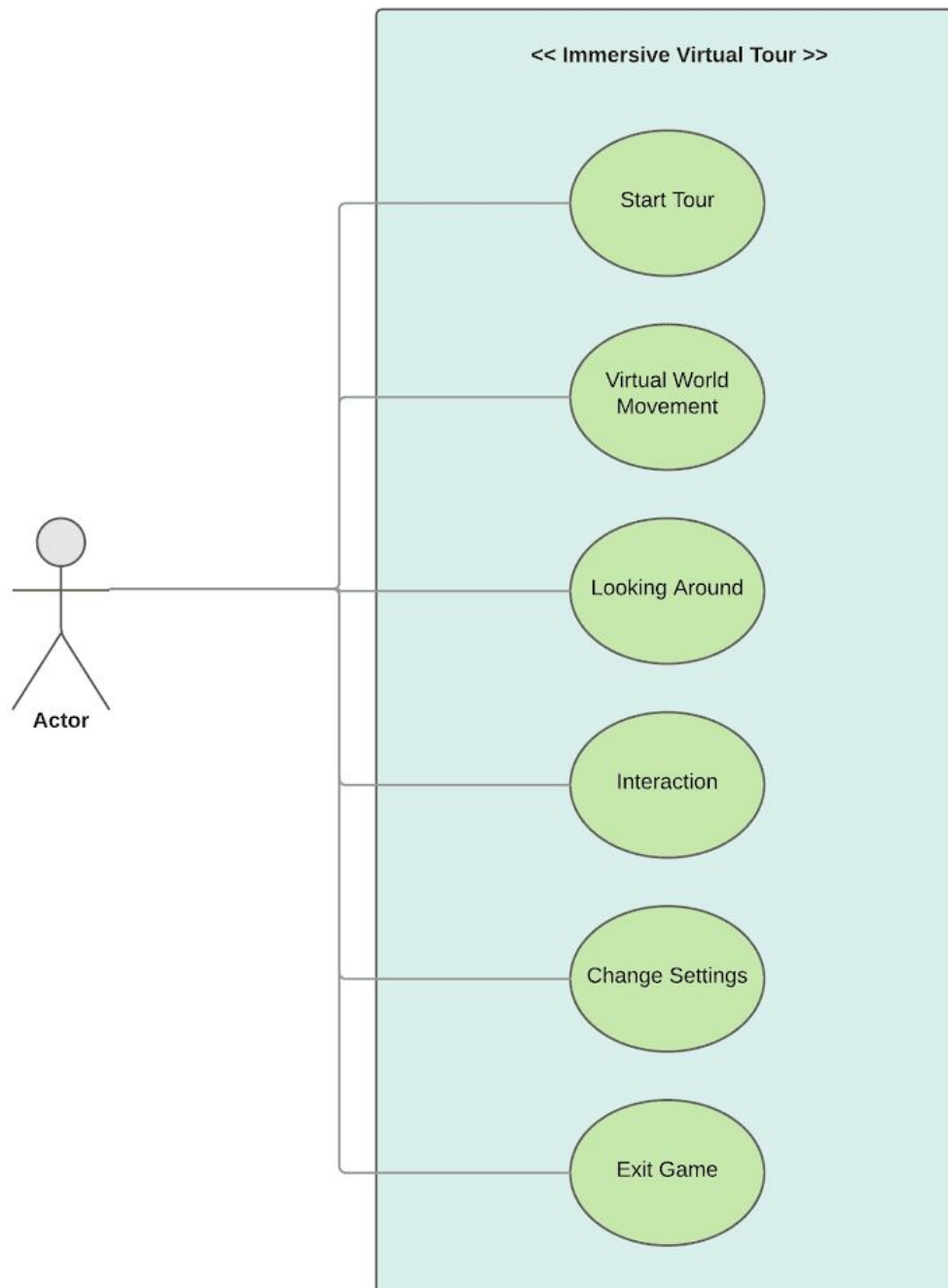


Figure 1: Use Case Diagram

U1: Start Virtual Tour

Description: This use case starts the Virtual Tour and places the user in the 3D Virtual World.

Actor: User

Scenario 1: Mainline Sequence

1. User: Requests to start virtual tour.
2. System: Displays the virtual world of the CSE Department, and shows the instructions to move/look around/interact.

U2: Virtual World Movement

Description: This use case allows the user to move in a direction according to his physical head movements.

Actor: User

Scenario 1: Mainline Sequence

1. System: Shows instructions for user to move.
2. User: Tilts his/her head in the forward direction.
3. System: Moves the user forward in the virtual world.

Scenario 2: At step 2 of mainline sequence.

2. User: Tilts his/her head in the backward direction.
3. System: Moves the user backward in the virtual world.

Scenario 3: At step 2 of mainline sequence.

2. User: Tilts his/her head towards the left.
3. System: Moves the user left in the virtual world.

Scenario 4: At step 2 of mainline sequence.

2. User: Tilts his/her head towards the right.
3. System: Moves the user right in the virtual world.

U3: Looking Around

Description: This use case allows the user to turn around in the virtual world, without moving.

Actor: User

Scenario 1: Mainline Sequence

1. System: Shows instructions for user to move.
2. User: Rotates his/her head towards the left.
3. System: Changes the direction the user is facing towards the left.

Scenario 2: At step 2 of mainline sequence.

2. User: Rotates his/her head towards the right.
3. System: Changes the direction the user is facing towards the right.

U4: Interaction

Description: This use case allows the user to interact with items in the Virtual World.

Actor: User

Scenario 1: Mainline Sequence

1. System: Displays a prompt requesting the user to press a button to interact with the object.
2. User: Presses the button while facing the object.
3. System: Displays text describing the object, plays sound associated with the object, and displays a short animation of the object.

Scenario 2: At step 2 of mainline sequence

2. User: Moves/turns away from the object.
3. System: Closes request prompt.

U5: Change Settings

Description: This use case allows the user to change settings, such as audio volume and brightness, while touring.

Actor: User

Scenario 1: Mainline Sequence

1. User : Requests the 'Settings' option.
2. System: Opens sub-menu with the following options:
 - a. Volume: To allow the user to change the in-tour audio volume.
 - b. VR Sensitivity: To allow the user to change the sensitivity of the VR Headset.
 - c. Brightness: To allow the user to change the screen brightness.
 - d. Font size: To allow the user to change the font size of displayed text in the tour.
 - e. Instructions to Tour: To understand how to play.
 - f. Return: To return to the tour.
3. User : Selects one of the six options displayed.
4. System : If the user selects,
 - a. Volume: displays a slider bar to increase / decrease the audio volume - range 0 to 100.
 - b. VR Sensitivity: displays a slider bar to increase / decrease the VR Sensitivity - range 0 to 10.
 - c. Brightness: displays a slider bar to increase / decrease the brightness - range 0 to 100.
 - d. Font size: displays 3 options of small, medium, large font sizes.
 - e. Instructions to Tour: shows the user a set of images showing how to move, turn and interact.
 - f. Return: returns the user to the tour.

U6: Exit Tour

Description: This use case allows the user to exit the Virtual Tour.

Actor: User

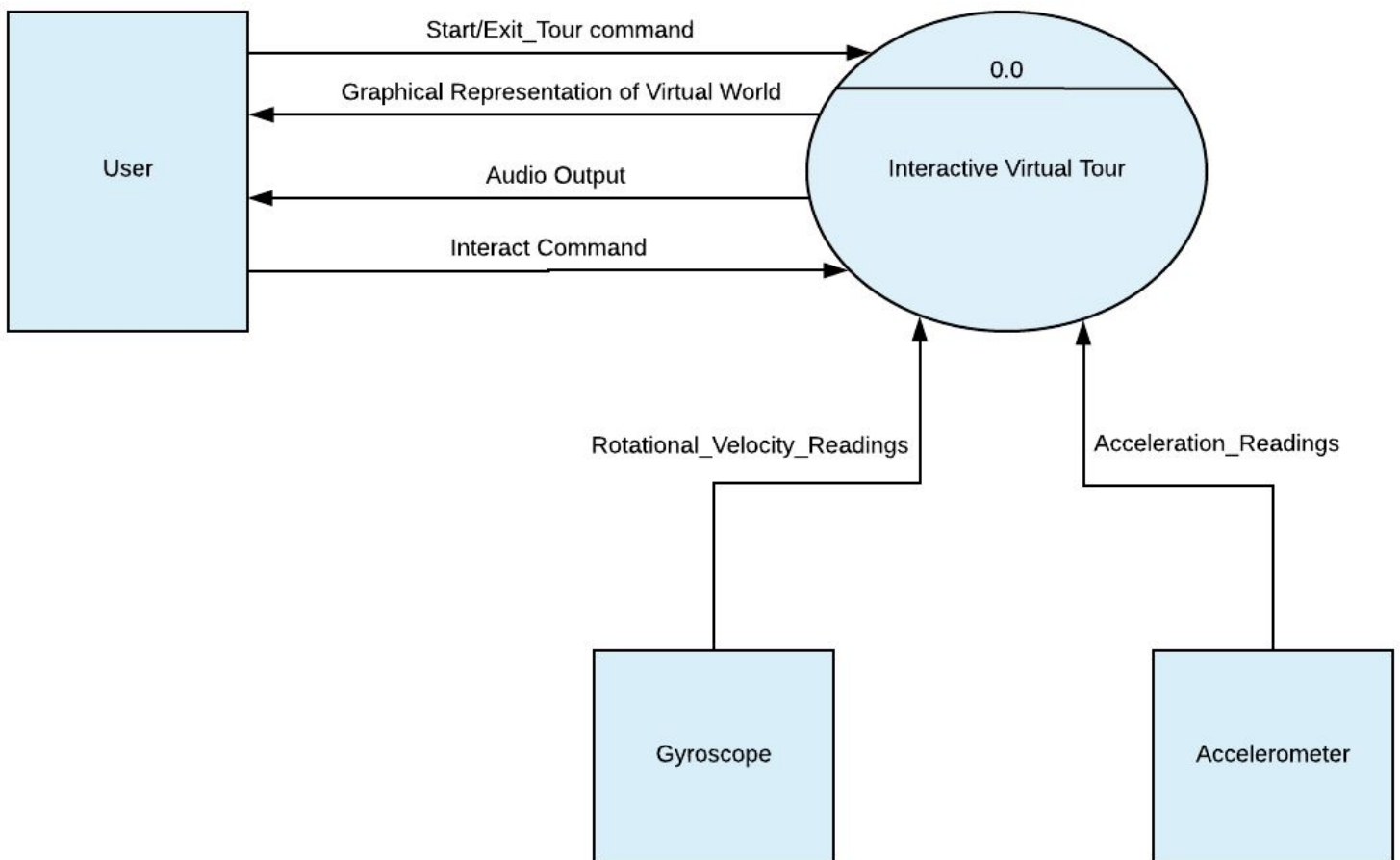
Scenario 1: Mainline Sequence

1. User: Selects 'Exit Tour' option from the menu.
2. System: Asks the user if they are sure about exiting the application.
3. Player: Selects the 'Yes' option.
4. System: Closes the virtual tour application.

Scenario 2 : At step 3 of mainline sequence

3. Player: Selects the 'No' option.
4. System: Resumes the tour.

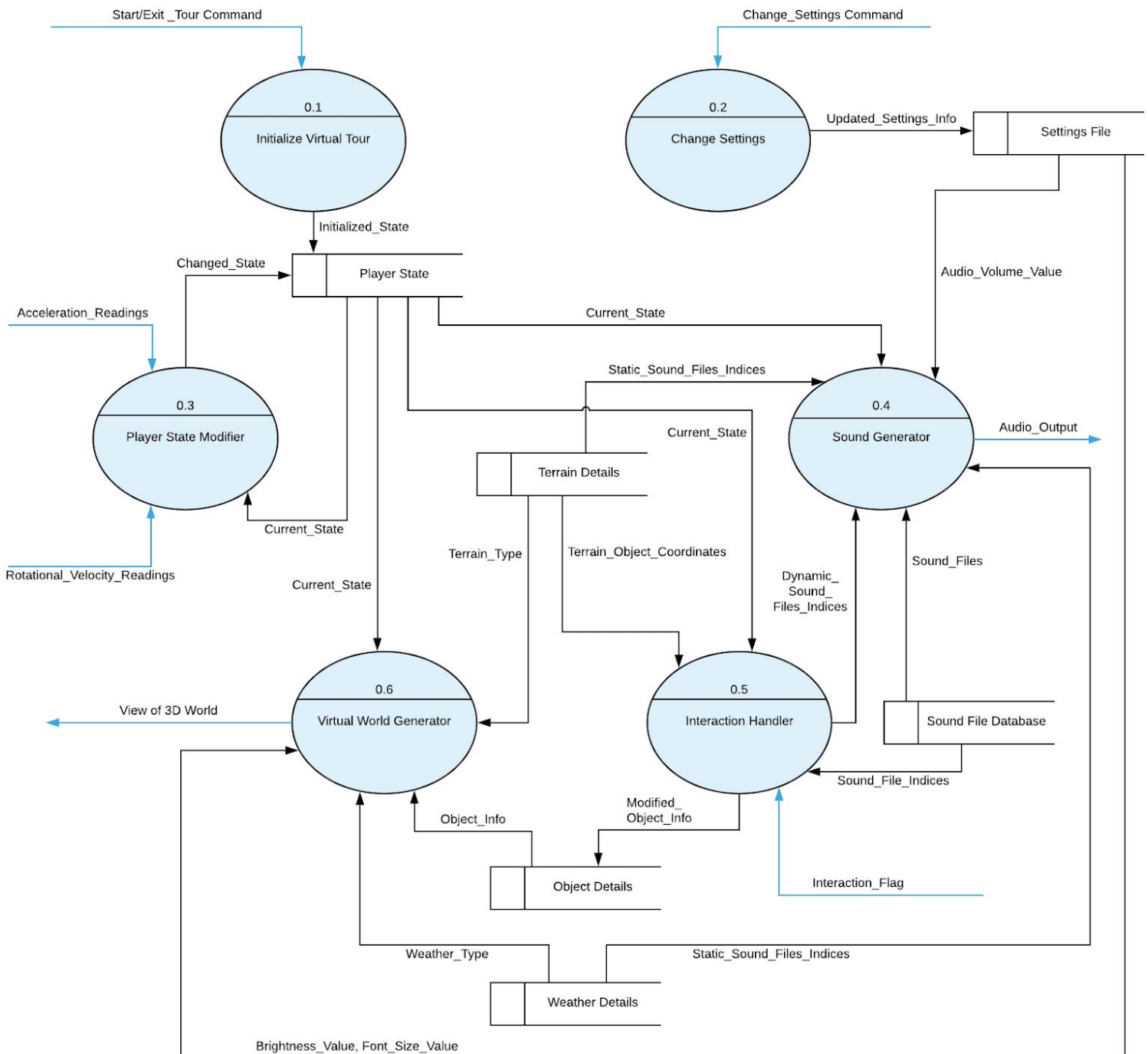
3. Data Flow Diagram: Level 0



- Blue Arrow** : For data flow to other DFD levels
Black Arrow : For data flow within the same DFD level

Note : Refer to the **Data Dictionary** for any queries about the data.

4. Data Flow Diagram: Level 1



Data Dictionary (for DFD Levels 0 and 1)

(Alphabetically ordered)

- **Acceleration_Readings** [double-3-tuple] Acceleration along the x, y, z coordinate axes of the VR Headset
- **Audio_Output** [.wav file] Audio playback for speakers.
- **Audio_Volume_Value** [integer] Value of volume setting - range 0 to 100.
- **Brightness_Value** [integer] Value of brightness setting - range 0 to 100.
- **Current_State** [double-5-tuple] x, y, z coordinates of the user in the world, as well as the azimuthal and elevation angles of the user's line-of-sight.
- **Changed_State** [double-5-tuple] Modified x, y, z coordinates of the user in the world, as well as the modified azimuthal and elevation angles of the user's line-of-sight.
- **Change_Settings Command** [boolean] true if the user has requested to change settings, false otherwise.
- **Dynamic_Sound_Files_Indices** [integer-vector] Indices of sound files to be played - corresponding to interactions between user and world.
- **Font_Size_Value** [integer] Value of font size setting - range 1 to 3 - indicating small, medium and large font sizing.
- **Interaction_Flag** [boolean] true when the user chooses to interact with the object in front.
- **Initialized_State** [double-5-tuple] Initial value of user's x, y, z coordinates and resets the azimuthal and elevation angles of the user's line-of-sight.
- **Object_ID** [integer] unique integer assigned to each object.
- **Object_Info** [structure] pre-defined data-type containing object ID, 3D RGB Matrix representing the object model, coordinates and orientation of the object.

- **Modified_Object_Info** [structure] pre-defined data-type containing the object ID, changed 3D RGB Matrix representing the object model, changed coordinates and orientation of the object.
- **Rotational_Velocity_Readings** [double] Values of rate of change of the azimuthal and elevation angles.
- **Sound_File** [.wav file] Sound file sent as an encoded bitstream.
- **Sound_File_Indices** [integer-vector] Indices of sound files.
- **Start/Exit_Tour Command** [boolean] Indicates whether the user initiates and terminates the tour.
- **Static_Sounds_Files_Indices** [integer-vector] Indices of sound files to be played - for static sound sources.
- **Terrain_Display_Matrix** [integer-3-tuple-3D-matrix] 3D RGB Matrix corresponding to the terrain.
- **Terrain_Object_Coordinates** [(double, double, double, integer)-4-tuple-list] List of x, y, z coordinates with the Object_ID of the object whose centre is placed at these coordinates.
- **Updated_Settings_List** [(string, integer)-vector] List of updated settings with updated values.
- **Weather_Display_Matrix** [integer-3-tuple-3D-matrix] 3D RGB Matrix corresponding to the weather.

5. Data Flow Diagram: Level 2

1. Initialize Virtual Tour

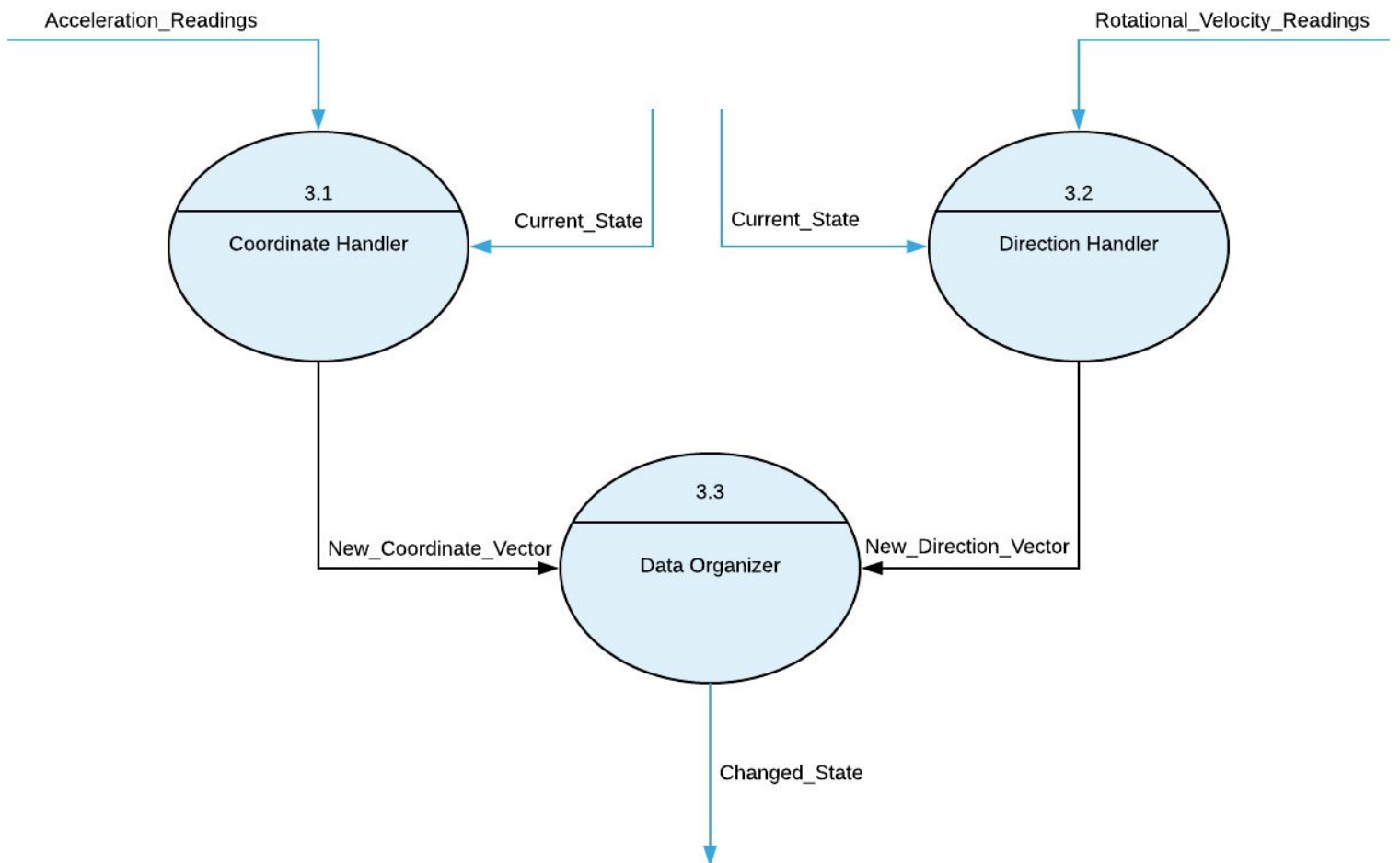
This process starts the tour and places the user in the Virtual World at a predefined location, and has no sub-processes in this DFD.

2. Change Settings

This process modifies the settings as per the user's input, and has no sub-processes in this DFD.

3. Player State Modifier

This process modifies the user's coordinates and direction he/she is facing, depending on the gyroscope and accelerometer readings.

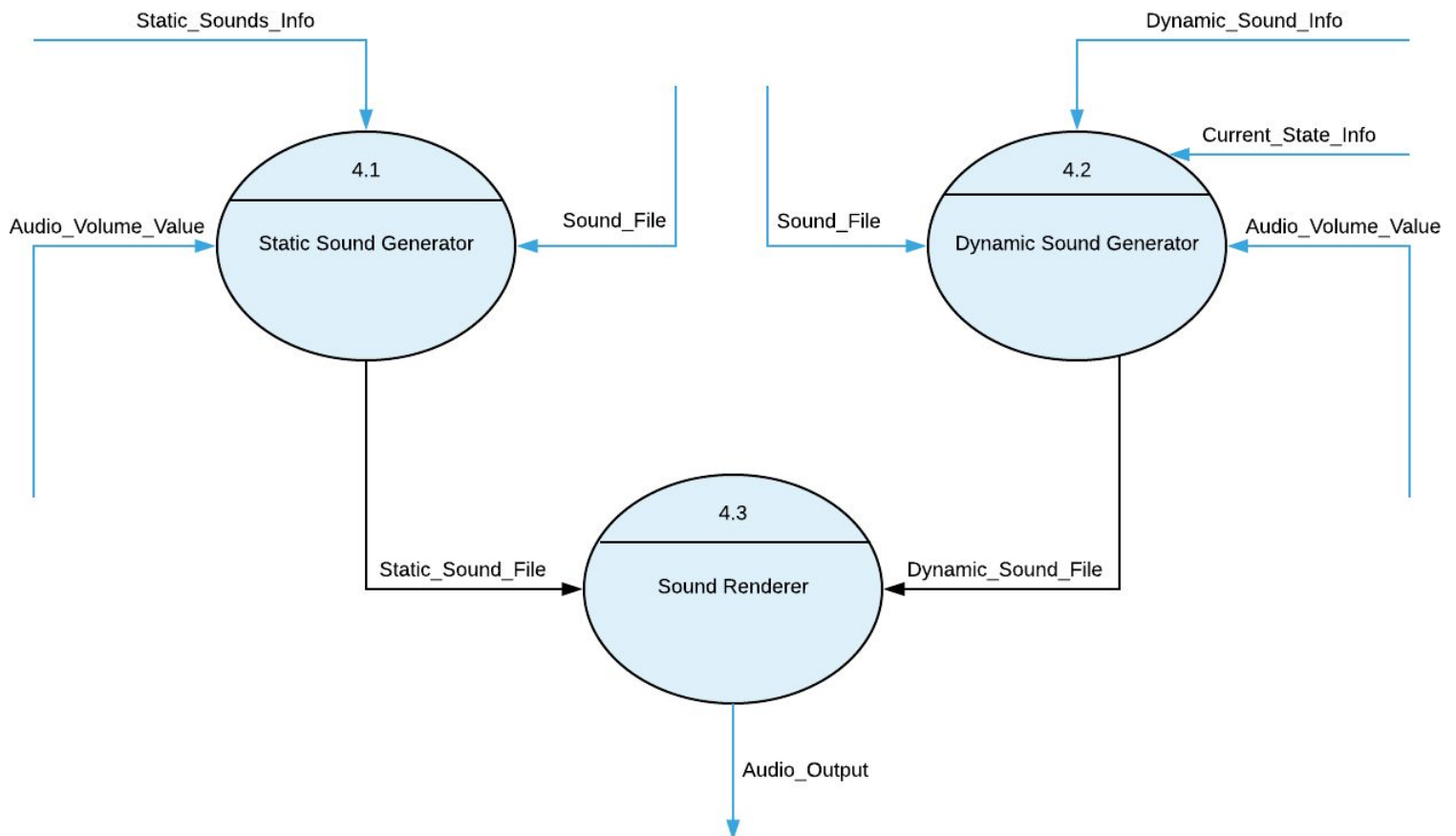


Additions to Data Dictionary:

- **New_Coordinate_Vector** [double-3-tuple] the new x, y, z coordinates of the user, after calculations.
- **New_Direction_Vector** [double-2-tuple] the new azimuthal and elevation coordinates, after calculations.

4. Sound Generator

This process generates the audio output, arising from objects and the interactions between the user and the Virtual World.

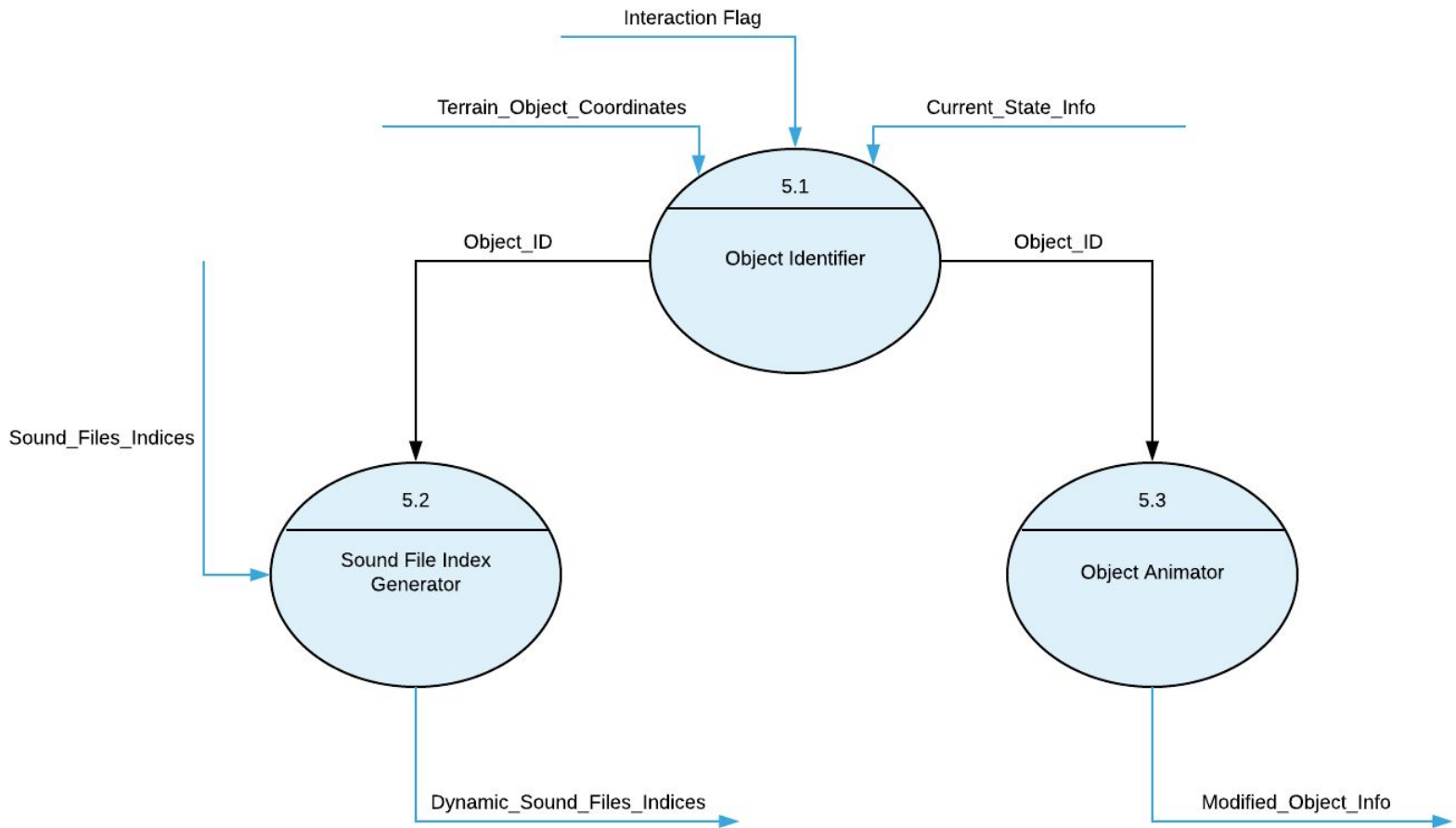


Additions to Data Dictionary:

- **Dynamic_Sound_File** [.wav file] audio file associated with dynamic sounds - from user interactions with the virtual world.
- **Static_Sound_File** [.wav file] audio file associated with static sounds.

5. Interaction Handler

This process handles interactions between the user and objects in the Virtual World.

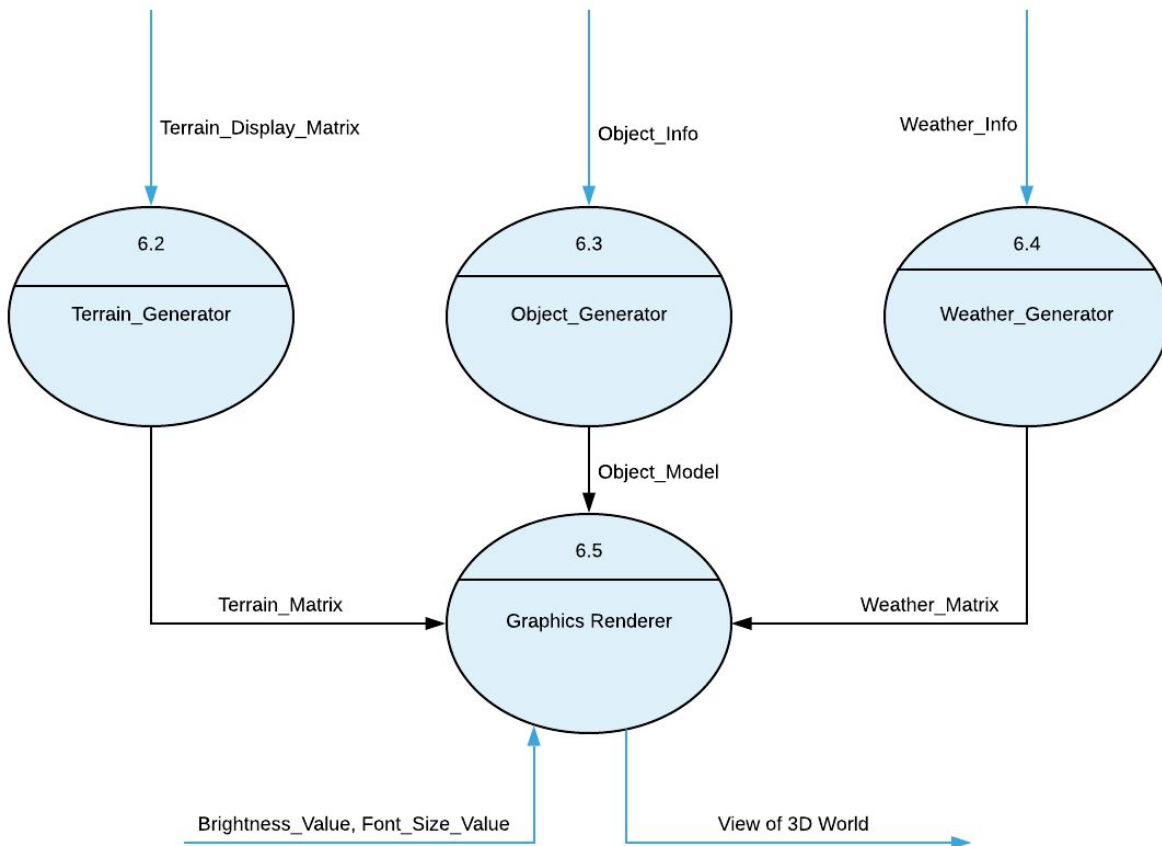


Additions to Data Dictionary:

- **Object_ID** [integer] unique integer assigned to each object when modified.

6. Virtual World Generator

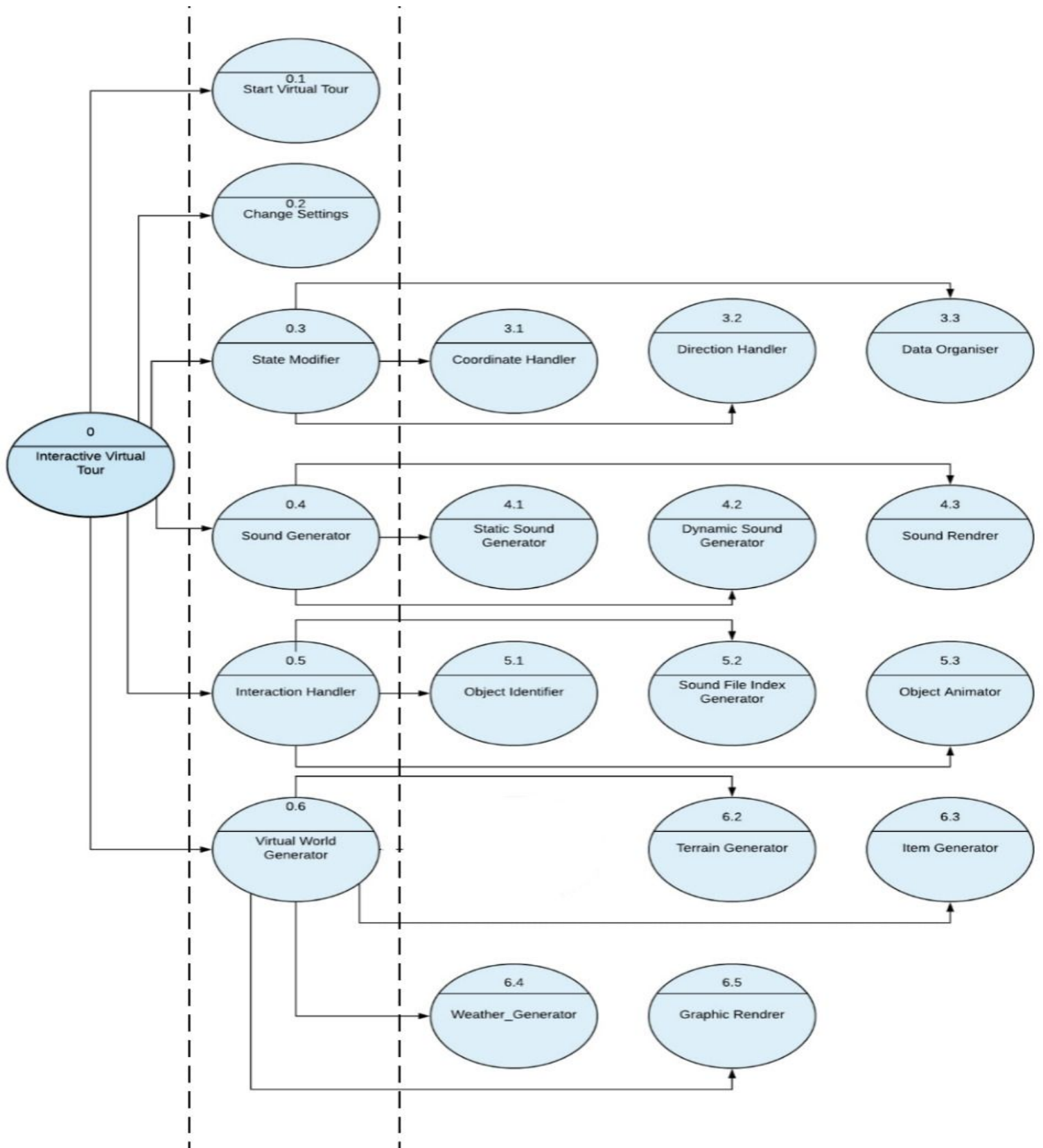
This process generates the graphics of the Virtual World that will be displayed to the user.



Additions to Data Dictionary:

- **Terrain_Display_Matrix** [integer-3-tuple-3D-Matrix] 3D matrix graphics representation of the terrain.
- **Terrain_Type** [structure] predefined data-type containing a 2D-matrix of height of the surface at each x, y coordinate, and surface type - soil, tile, ground.
- **Object_Data** [structure] predefined data-type containing a 2D-matrix of height of the surface at each x, y coordinate, and surface type - soil, tile, ground.
- **Object_Model** [integer-3-tuple-3D-Matrix] 3D matrix graphics representation of the object.
- **Weather_Display_Matrix** [integer-3-tuple-3D-matrix] 3D RGB Matrix corresponding to the weather.
- **Weather_Type** [integer] Value representing the weather type - sunny, rainy, cloudy.

6. Process Decomposition Diagram



7. ER Diagram

