InternPro Weekly Progress Update

Name	Email	Project Name	NDA/ Non-NDA	InternPro Start Date	OPT
Adharsh Prasad Natesan	anatesan@asu.edu	IT-Core Foundation Suriname	Non-NDA	2024-08-05	Yes

Progress

Include an itemized list of the tasks you completed this week.

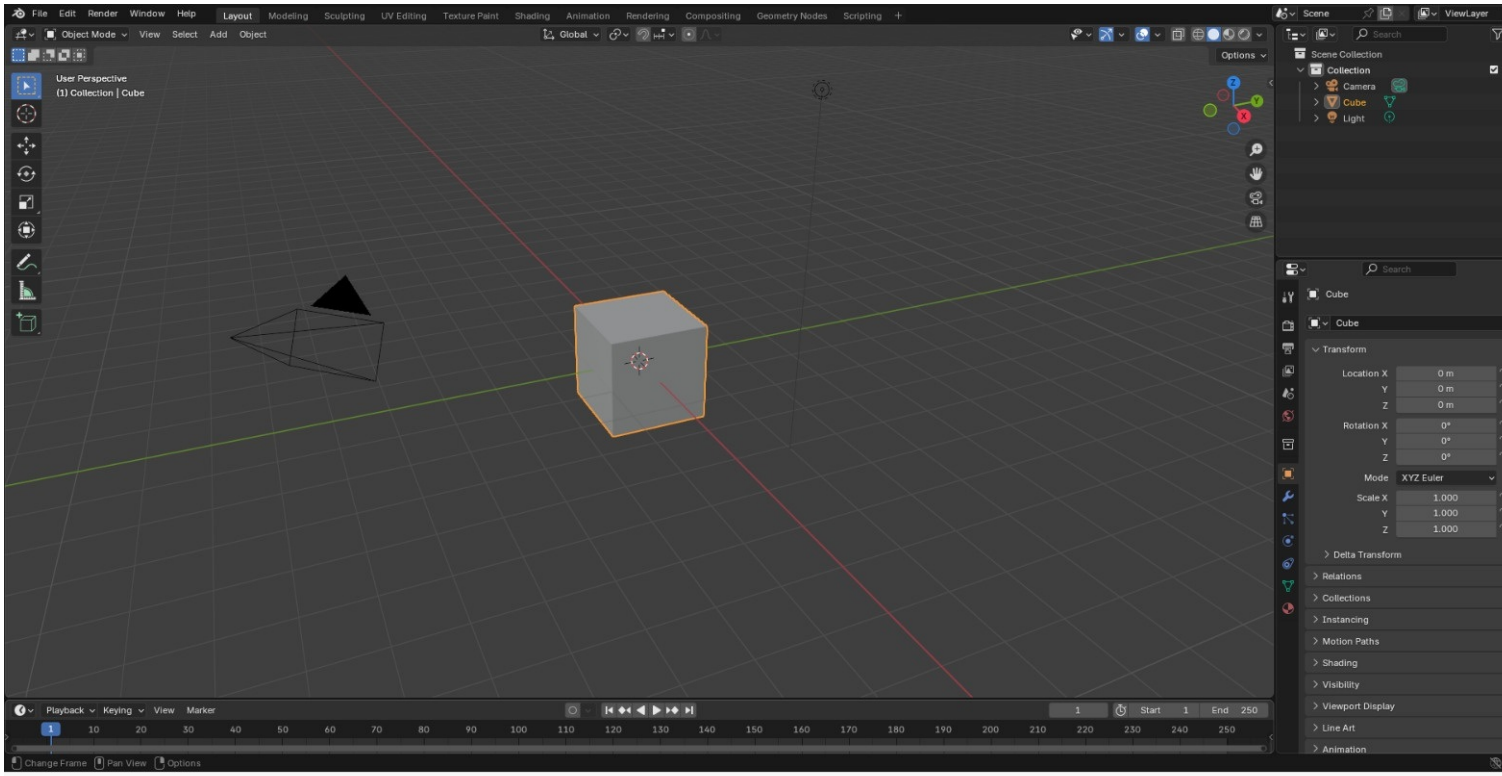
#	Action Item/ Explanation	Total Time This Week (hours)
1	Installing Blender and setting up	3
2	Creating and Sculpting Terrain in Blender	3
3	Finding suitable Soil Texture and Applying in Blender	3
4	Creating Crop rows in Blender	3
5	Procedural Generation and Blender Integration for Farmland Simulation	3
6	Exporting a Blender Map to Gazebo	3
7	Weekly Plan for Next Week	1
8	Report Writing	1
	Total hours for the week:	20

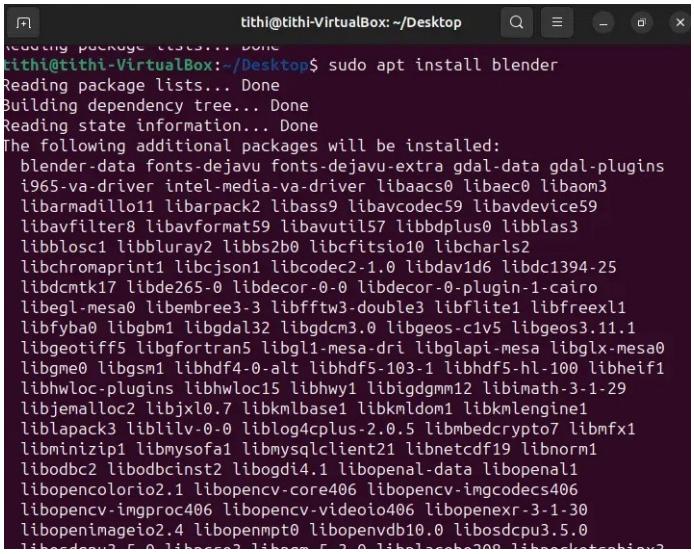
Verification Documentation:

Action Item 1: Installing Blender and setting up – 3 hour(s).

Project Work Summary

- This week, I'm working with Blender to create the simulation map for future integration with the gazebo. By creating the map with Blender it becomes flexibility to create and adjust maps as we needed, making it feasible for designing custom simulation environments.
- We constantly ran into some trouble with Husky navigating the heightmap of the virtual maize map. That's why I decided to make the map without importing and more over the map is not very realistic from the sources online and eventually get more control of the environment by creating a custom map in Blender.
- Steps to Install Blender via Terminal
  - Before installing Blender, made sure my system was up to date with the sudo update. I ran the necessary terminal commands to refresh the package list and upgrade any outdated packages.
  - Since we need the latest version directly from Blender's developers, so i explored the official Blender repository for information.
  - Once the repository was added, I installed Blender using a simple terminal command. The process was smooth, and the installation completed without issues.
  - To confirm everything worked as expected, I launched Blender from the terminal using the command blender. It opened without any issues, so we finally know the installation was successful and ready to use.
- Unlike prebuilt maps, Blender allows to create exactly what we need to tackle Husky's navigation challenges. We can tweak and adapt it for different simulations and scenaros as needed. By designing our own map, we can ensure better control over elevation and obstacles.

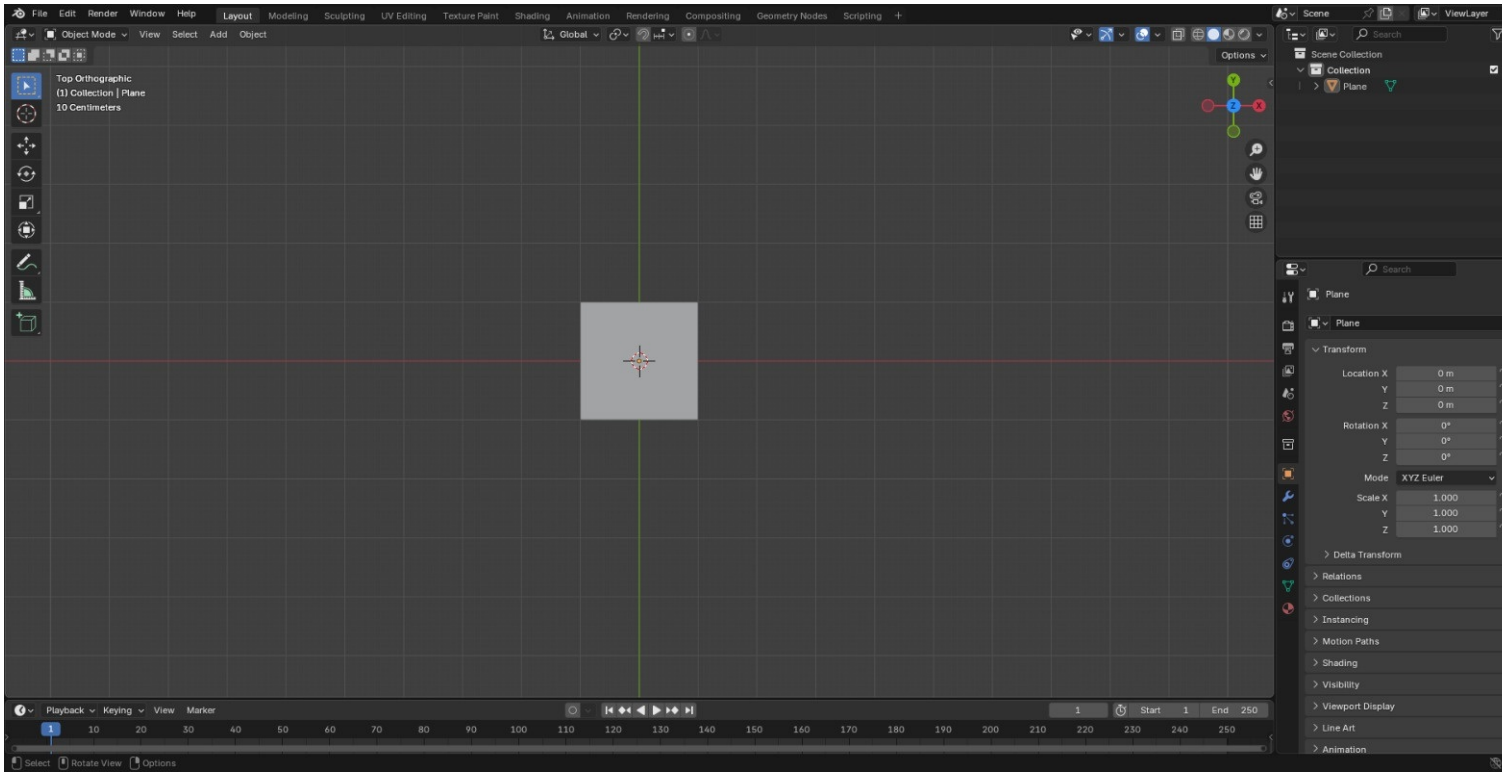


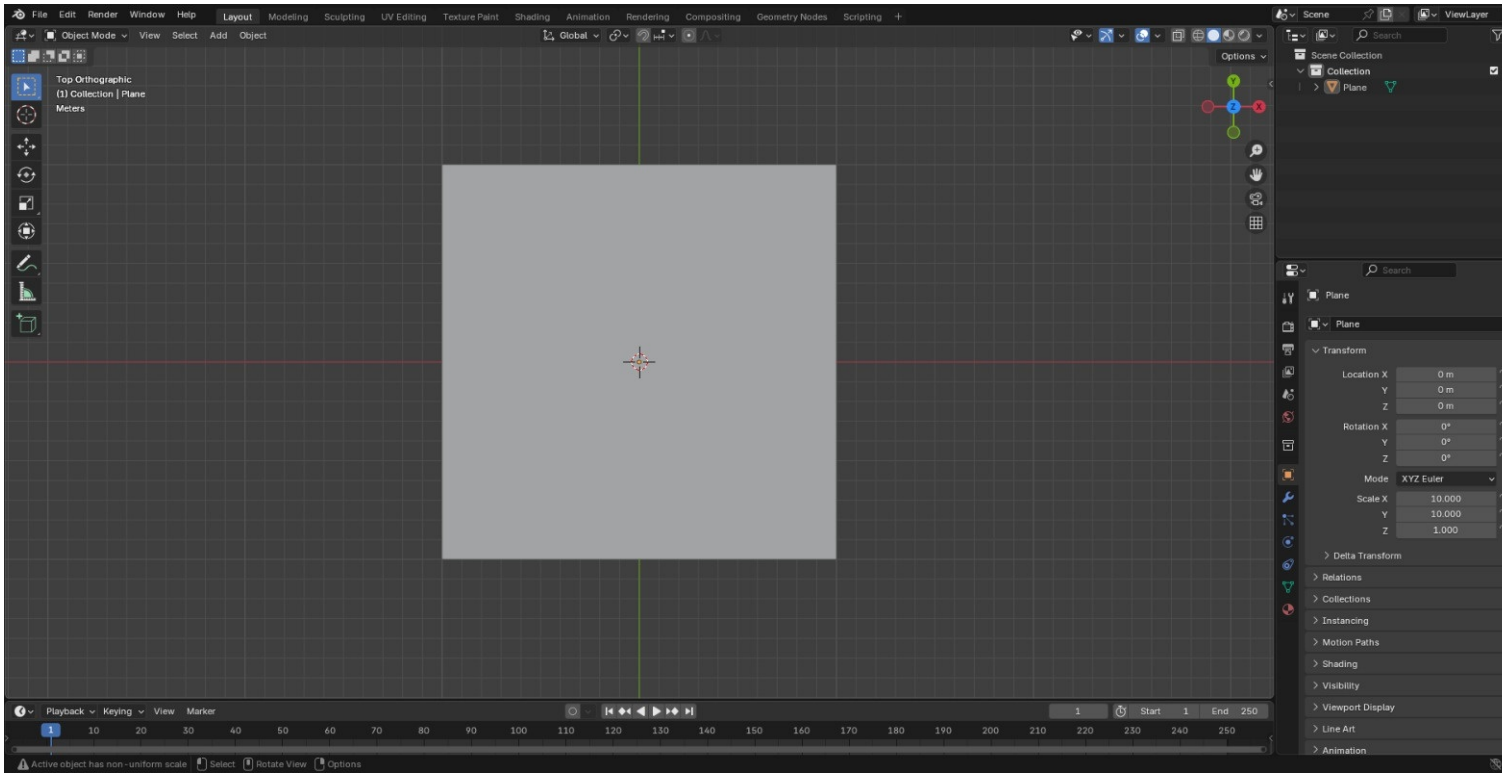


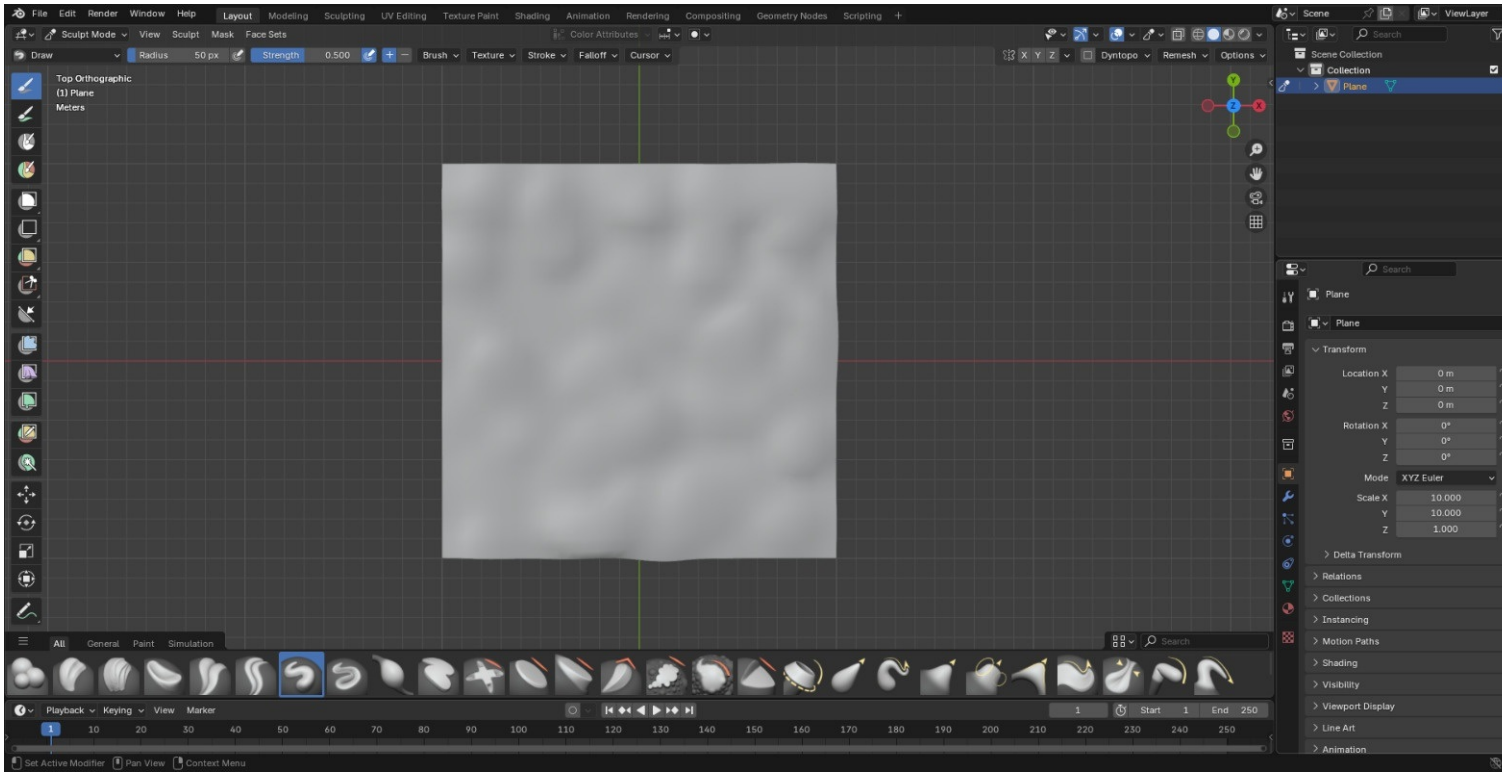
Action Item 2: Creating and Sculpting Terrain in Blender – 3 hour(s).

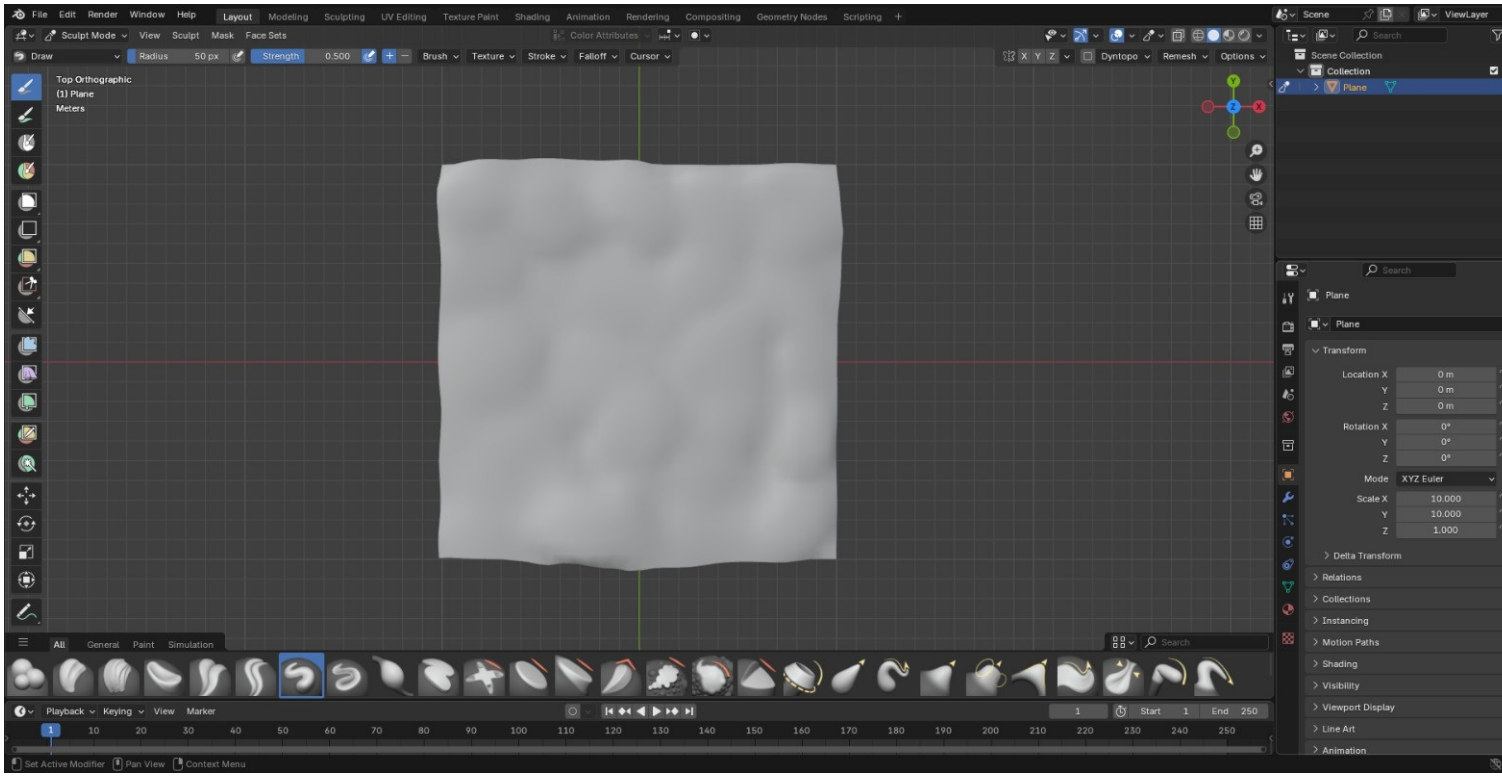
Project Work Summary

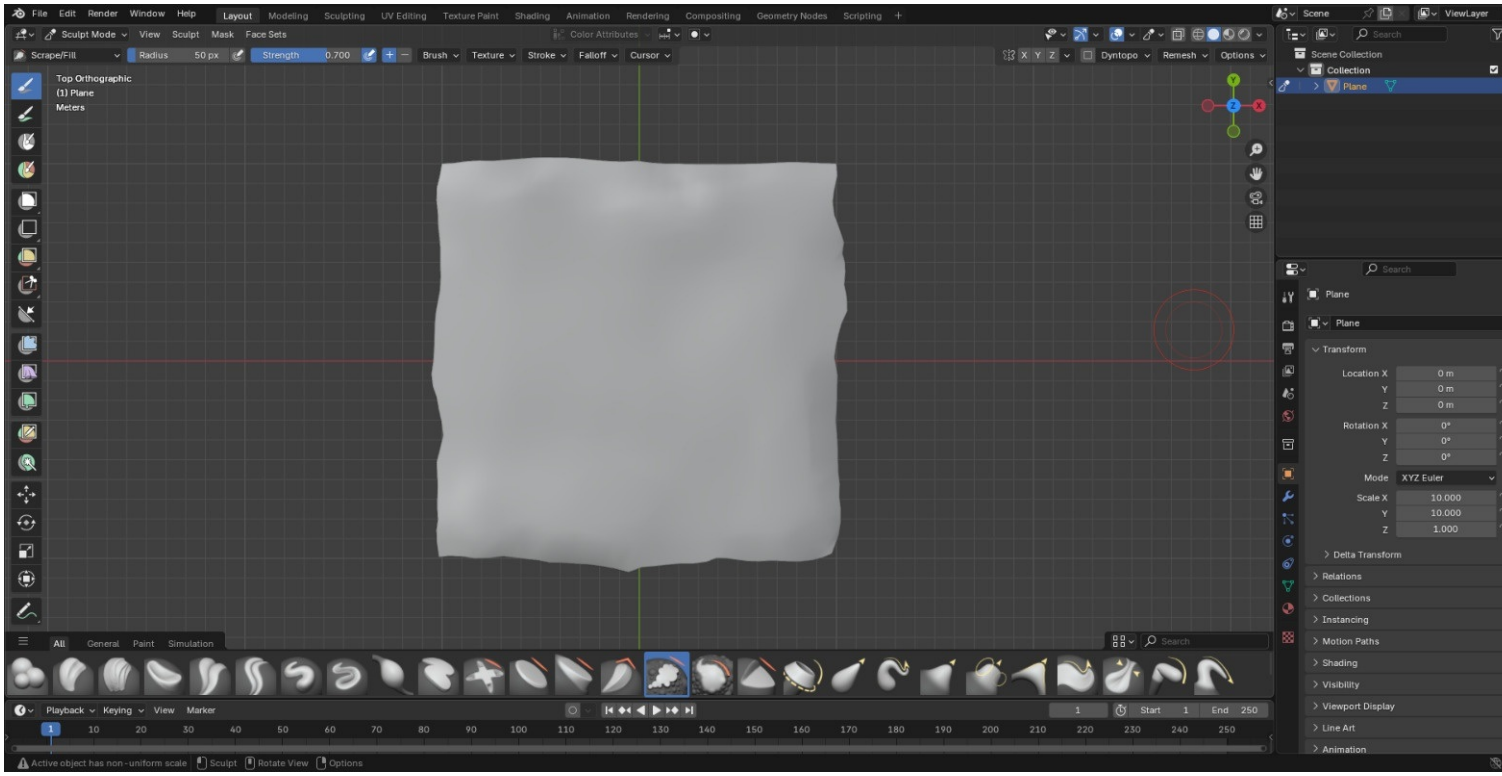
- Now we need to start work on to building the custom terrain in Blender to add our husky to the map and the gazebo. By designing the terrain from scratch, we can have a better control over the map and ensure that we can meets Husky's requirements while also making it adaptable for future applications. Blender gives that flexibility to sculpt realistic environments, making it a great tool for this task.
- Steps to Create the Terrain
  - Setting Up the Scene
    - Started by preparing the workspace in Blender:
    - Removed the default cube to clear the scene (Shift + A → Delete).
    - Added a new plane to serve as the base for the farmland (Mesh → Plane).
    - Scaled up the plane (S → 15) to give myself enough space to design a large, realistic farmland.
  - Subdividing the Plane
    - To make sure the terrain had enough detail for sculpting:
      - Entered Edit Mode (Tab) and selected the plane.
      - Right-clicked and chose "Subdivide" to increase the number of cuts.
      - Added many the cuts until the number of subdivides are 200, ensuring there was enough geometry to sculpt fine details later.
- Sculpting the Farmland
  - With the subdivided plane ready, then we switched to Sculpt Mode (Ctrl + Tab → Sculpt Mode).
  - Shaping the Terrain, I used various sculpting tools to shape the farmland:
    - Draw Brush to created hills and elevated areas.
    - Flatten Brush to smoothen out certain sections to form flat, farmable land.
    - Dyntopo (Dynamic Topology) to add finer details, ensuring small bumps and indentations were accurately represented.
    - Scrape Brush to added natural imperfections and subtle variations in elevation to make the terrain look more realistic.

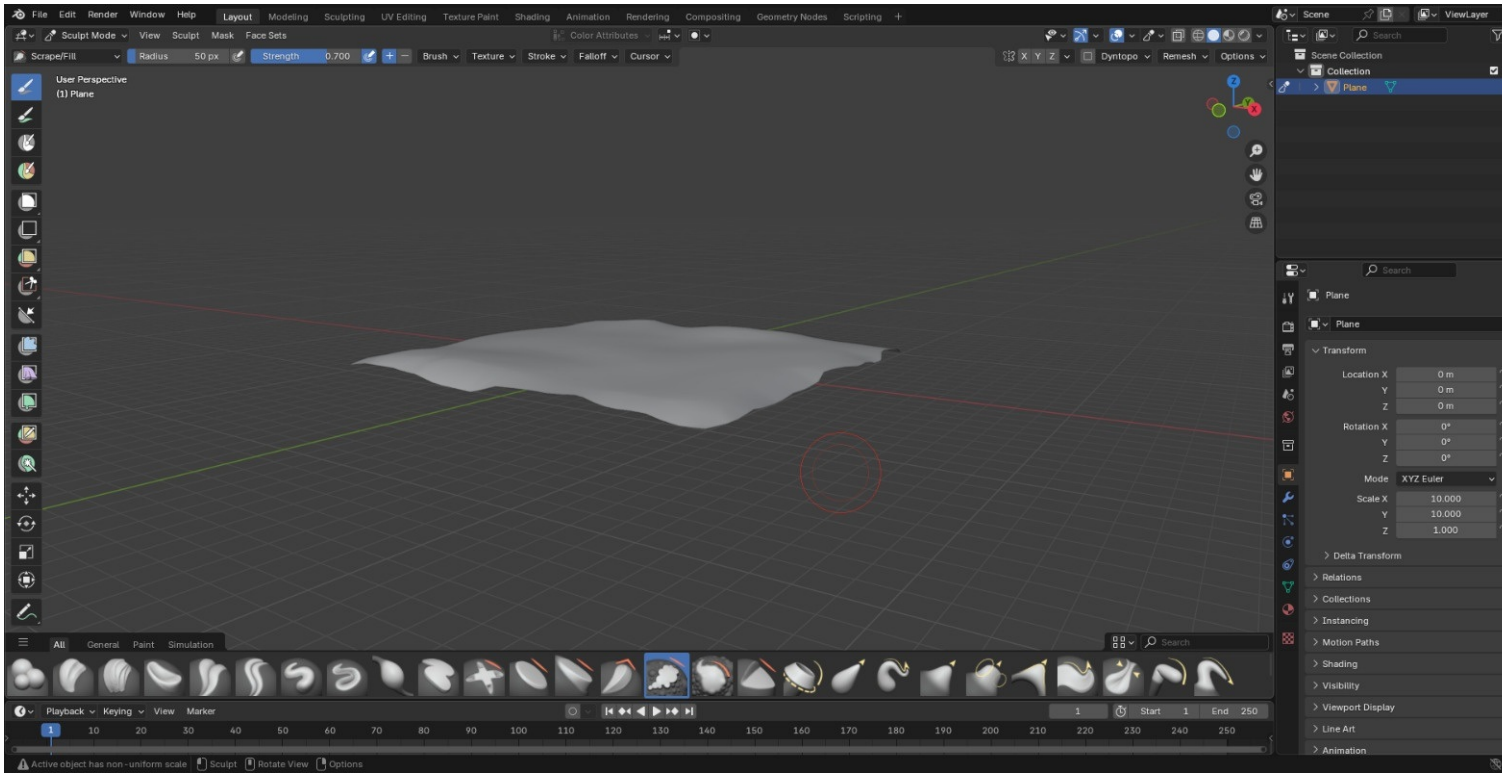

Top Orthographic
(1) Collection | Plane
10 Centimeters
Location X 0 m
Rotation X 0°
Scale X 1.000


Top Orthographic
(1) Collection | Plane
Meters
Location X 0 m
Rotation X 0°
Scale X 10.000
Y 10.000


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Sculpt Mode View Sculpt Mask Face Sets
Draw Radius 50 px Strength 0.500 Brush Texture Stroke Falloff Cursor
Top Orthographic
(1) Plane
Meters
Scene
ViewLayer
Scene Collection
Collection
Plane
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 10.000
Y 10.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
All General Paint Simulation
Playback Keying View Marker
Start 1 End 250
Set Active Modifier Pan View Context Menu


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Sculpt Mode View Sculpt Mask Face Sets
Draw Radius 50 px Strength 0.500 Brush Texture Stroke Falloff Cursor
Top Orthographic
(1) Plane
Meters
Scene
ViewLayer
Scene Collection
Collection
Plane
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 10.000
Y 10.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
All General Paint Simulation
Playback Keying View Marker
Start 1 End 250
Set Active Modifier Pan View Context Menu

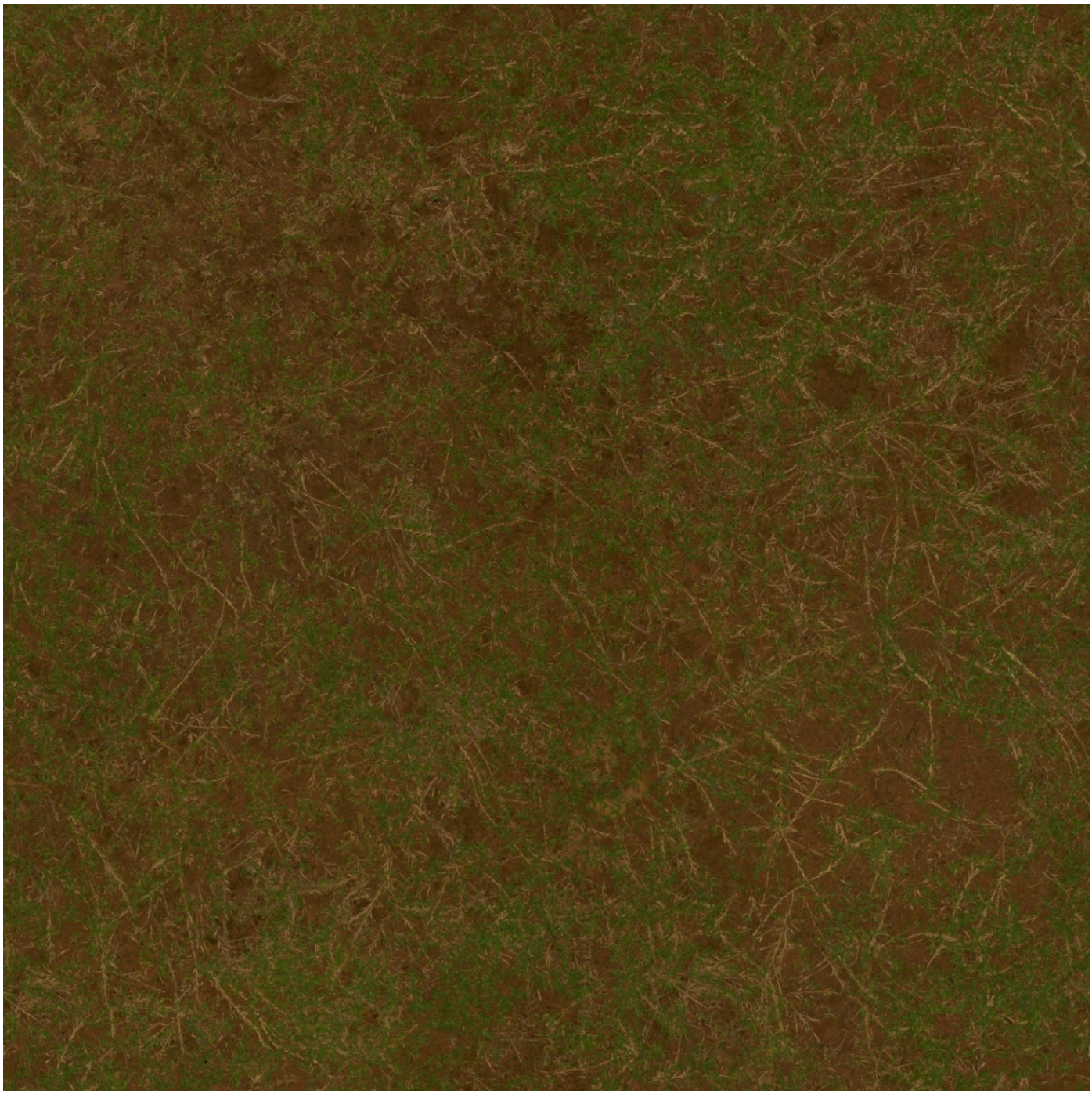


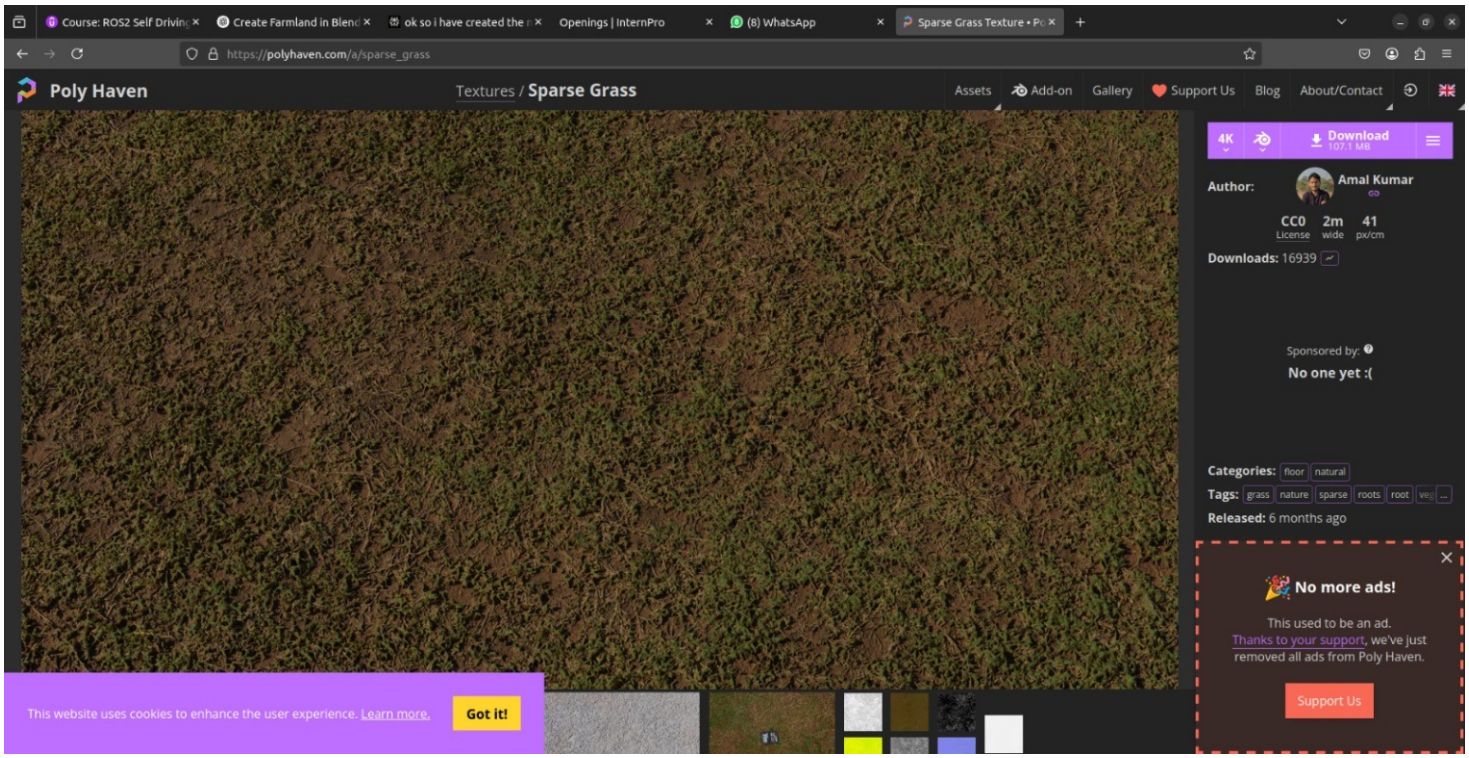


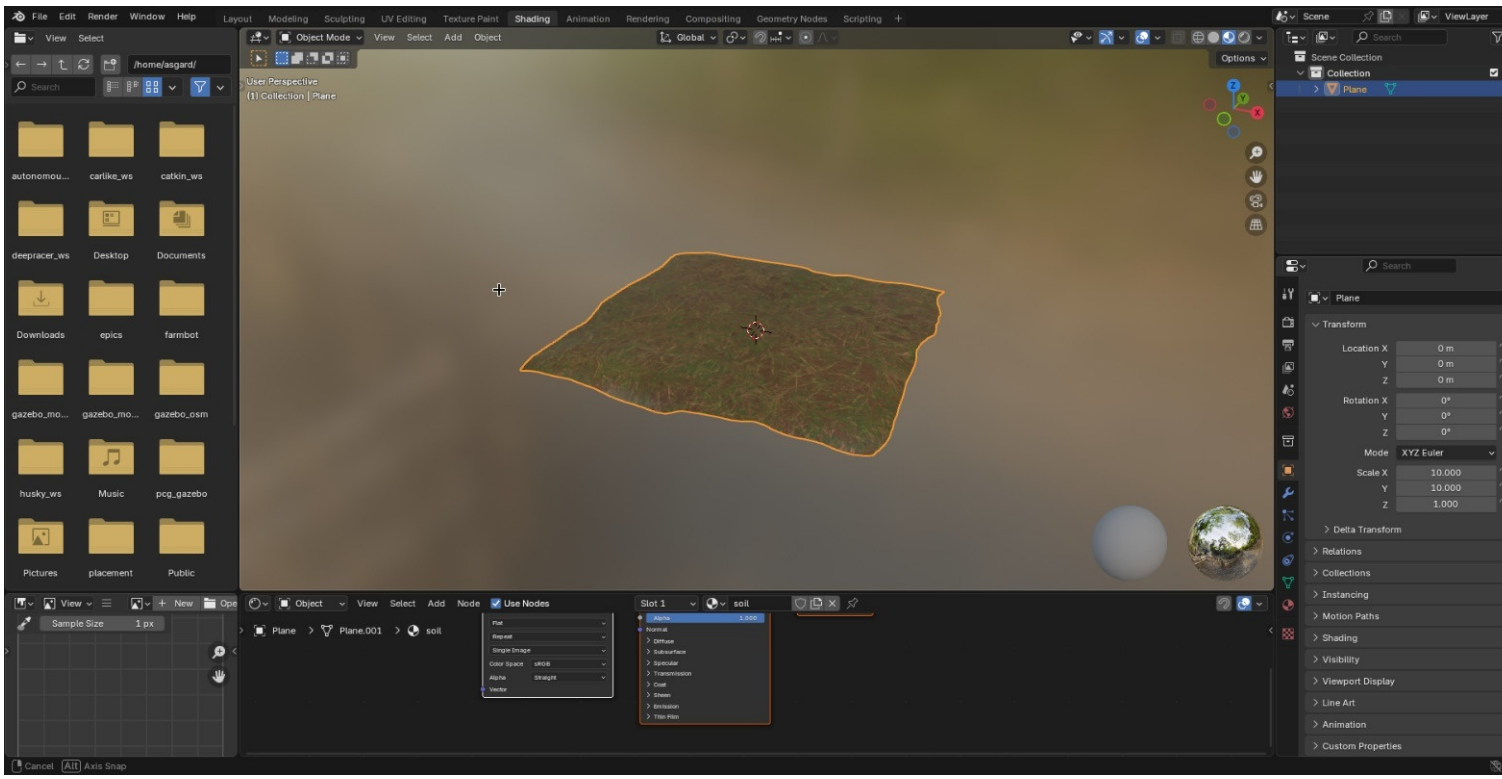
Action Item 3: Finding suitable Soil Texture and Applying in Blender – 3 hour(s).

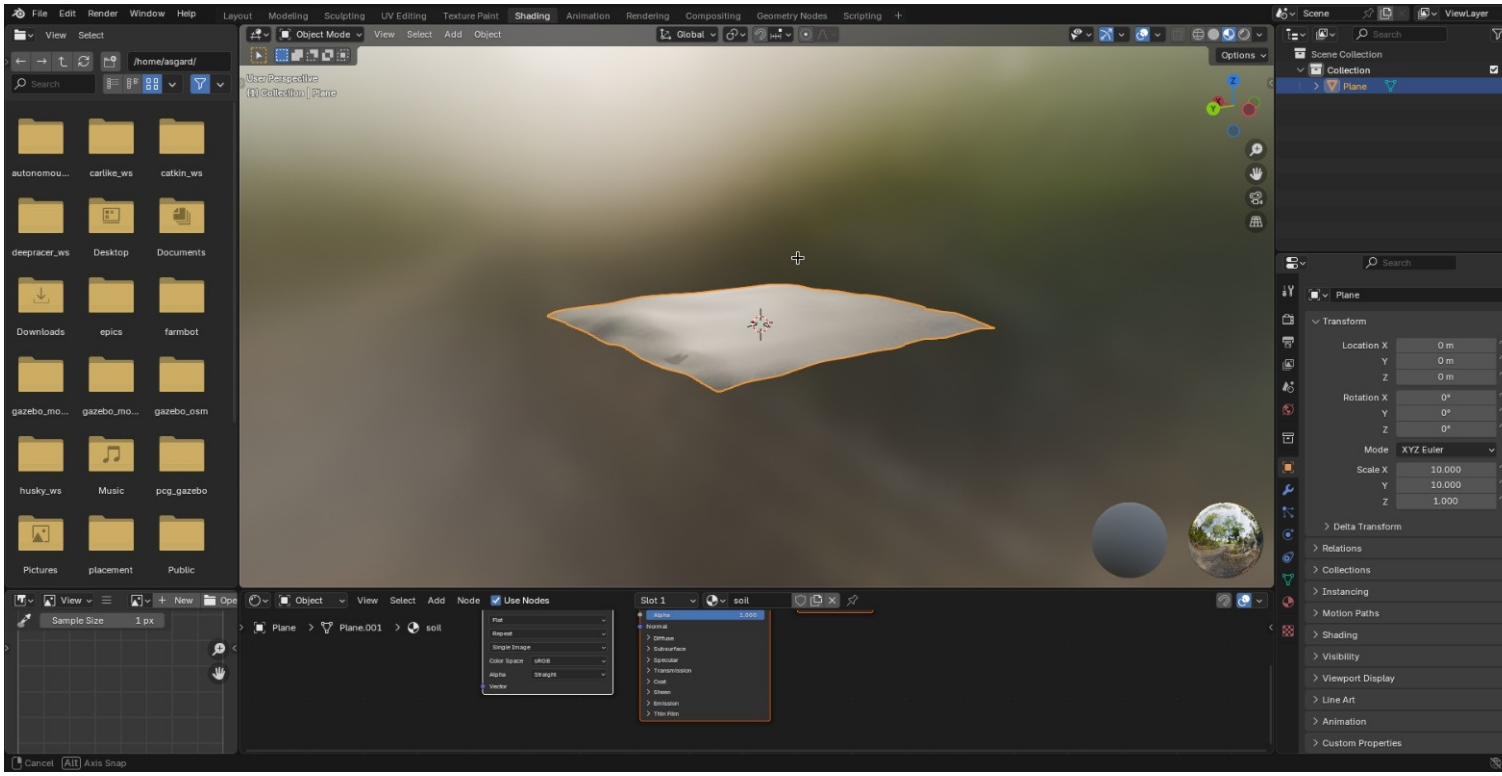
Project Work Summary

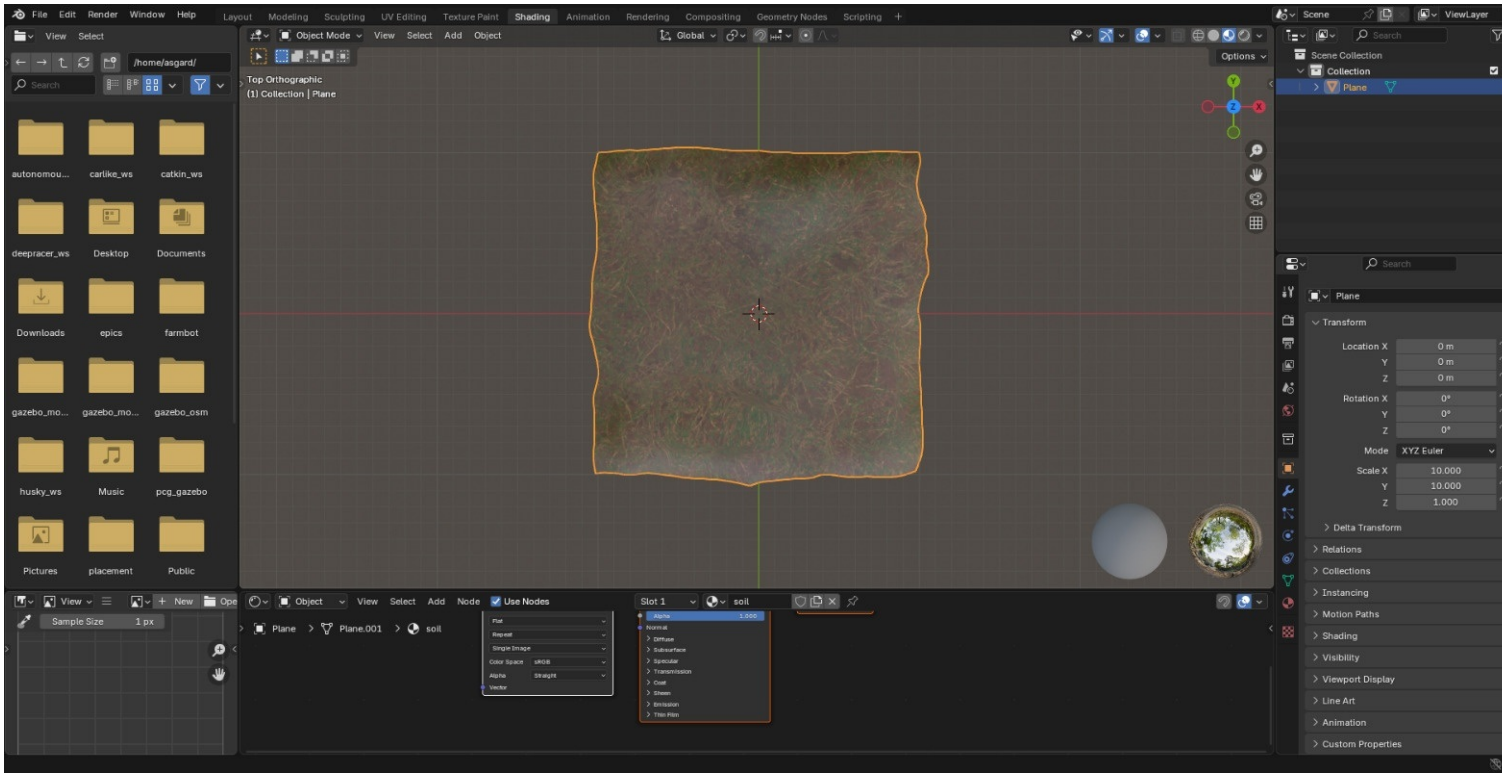
- After sculpting the terrain for the custom map, the next step was to add a realistic soil texture. This was important to both make visual appeal and also to make the environment feel more immersive in the Gazebo simulation. A well-textured landscape enhances realism, making it easier to test Husky's navigation effectively.
- Finding the Right Soil Texture
  - Researching Texture Sources
    - Started by looking for high-quality PBR (Physically Based Rendering) textures that would make the terrain look as realistic as possible. After exploring different options, found the Poly Haven that contains a lot of resource that are free with high-quality textures designed for 3D environments.
  - Choosing the Best Texture
    - On Poly Haven's website, I browsed through their terrain textures, found the soil and ground textures. I was looking for something that closely resembled a real farmland surface that contains a mix of soil with sparse patches of grass for a natural look.
  - Downloading the Texture
    - SetOnce I found the perfect soil texture, I downloaded the full texture set, which included:
      - Base Color Map to define the surface's main color and appearance.
      - Normal Map to add fine surface details like bumps and grooves.
      - Roughness Map to control how shiny or matte the surface appears.
      - Displacement Map to create additional depth by adjusting the surface elevation.
  - Organizing Files

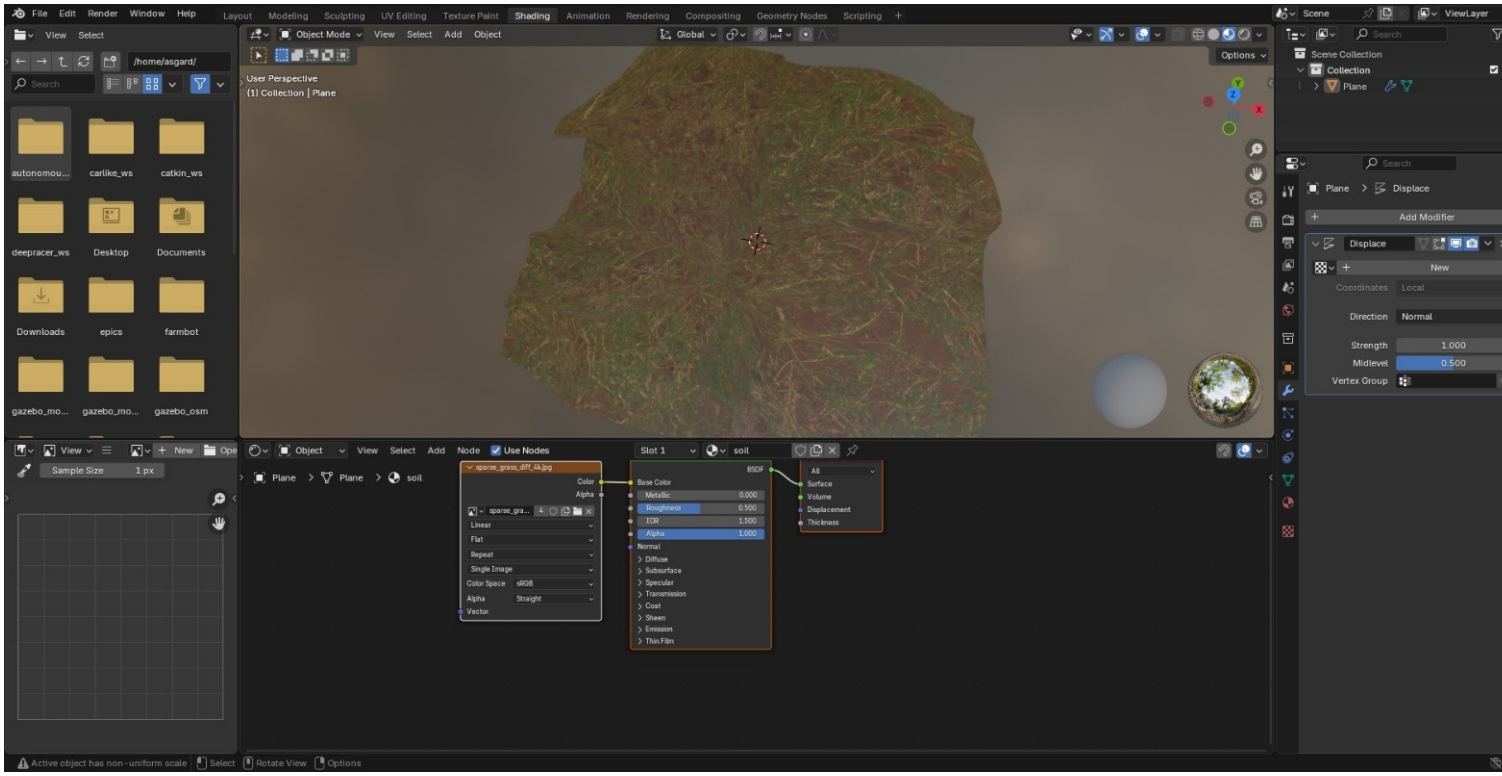
        - Saved all the textures in a dedicated folder on my farmland directory. A clean workflow helps when working with multiple assets in Blender.
- Applying the Texture in Blender
    - Switching to the Shading Workspace
        - With the terrain ready, I moved to Blender’s Shading tab to start applying the textures.
    - Creating a New Material
        - Selected the terrain mesh and created a new material in the Material Properties panel. This would serve as the base for my soil texture.
    - Setting Up the Node Network
        - Using Blender’s Shader Editor, I built a material network to apply the texture properly:
            - Base Color Texture → Connected to the Base Color input of the Principled BSDF shader.
            - Normal Map → Linked through a Normal Map node to add realistic surface bumps.
            - Roughness Map → Connected to the Roughness input to adjust surface reflectivity.
    - Adding Noise for More Realism
        - To make the soil look more natural and less uniform, I added some variation:
            - Inserted a Noise Texture node (Shift + A → Texture → Noise Texture).
            - Mixed it with the Base Color using a MixRGB node to create subtle variations in soil color.
            - Adjusted the scale and detail parameters to enhance the effect.
- 

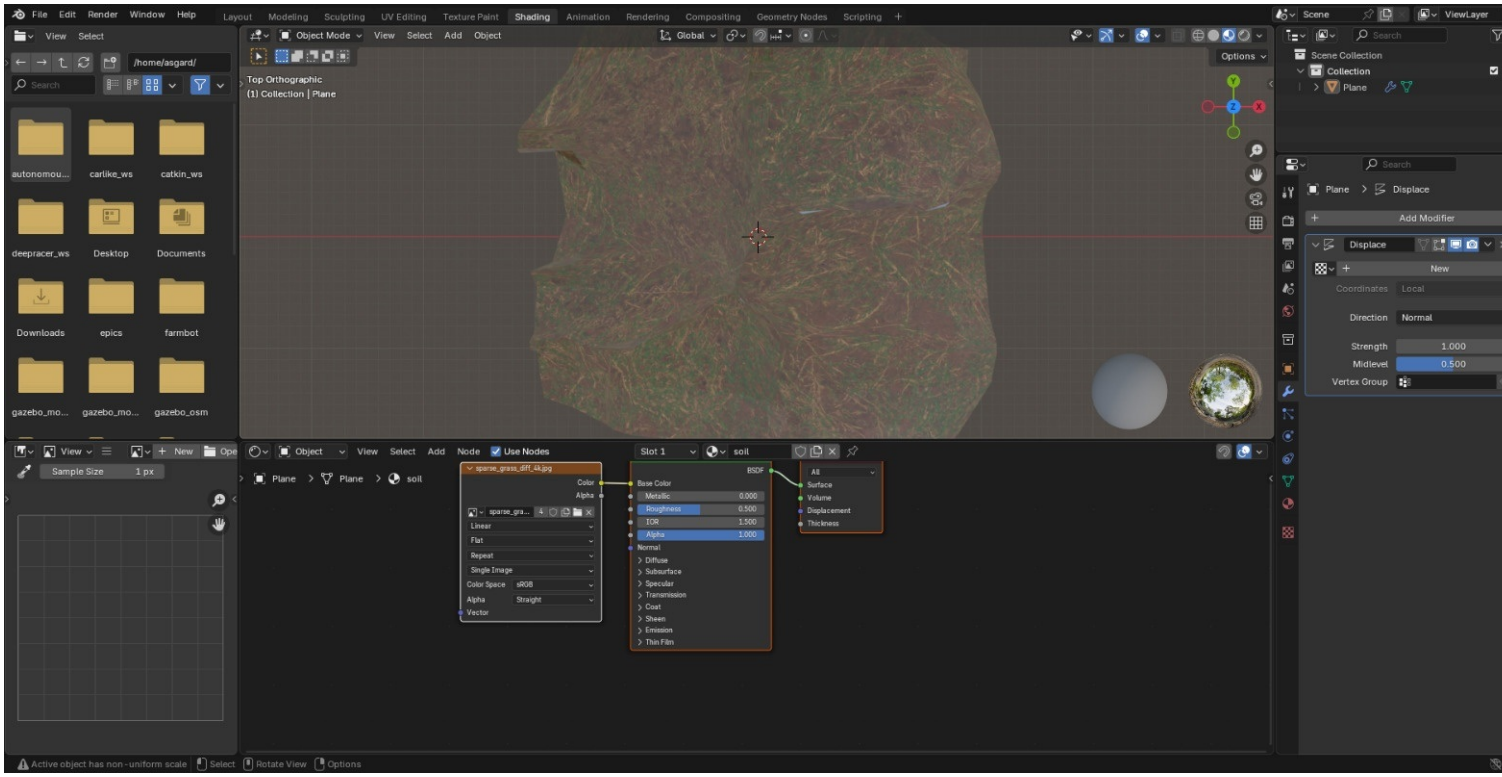

Poly Haven
Textures / Sparse Grass
Assets
Add-on
Gallery
Support Us
Blog
About/Contact
4K
Download
Author:
Amal Kumar
CC0
License
2m
wide
41
px/cm
Downloads: 16939
Sponsored by:
No one yet :(
Categories: floor natural
Tags: grass nature sparse roots root
Released: 6 months ago
No more ads!
This used to be an ad.
Thanks to your support, we've just removed all ads from Poly Haven.
Support Us
This website uses cookies to enhance the user experience. Learn more.
Got it!


Layout
Modeling
Sculpting
UV Editing
Texture Paint
Shading
Animation
Rendering
Compositing
Geometry Nodes
Scripting
Object Mode
View
Select
Add
Object
Global
Options
User Perspective
(1) Collection | Plane
Scene
ViewLayer
Scene Collection
Collection
Plane
Transform
Location X
Rotation X
Mode
XYZ Euler
Scale X
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
Custom Properties
Use Nodes
Plane.001
soil


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
View Select
/home/asgard/
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | Plane
autonomou...
carlike_ws
catkin_ws
deepracer_ws
Desktop
Documents
Downloads
epics
farmbot
gazebo_mo...
gazebo_mo...
gazebo_osm
husky_ws
Music
pcg_gazebo
Pictures
placement
Public
Scene
ViewLayer
Scene Collection
Collection
Plane
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 10.000
Y 10.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
Custom Properties
Sample Size 1 px
Object View Select Add Node Use Nodes
Slot 1
soil
Plane Plane.001 soil
Cancel Alt Axis Snap


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
View Select
/home/asgard/
Object Mode View Select Add Object
Global
Options
Top Orthographic
(1) Collection | Plane
autonomou...
carlike_ws
catkin_ws
deepracer_ws
Desktop
Documents
Downloads
epics
farmbot
gazebo_mo...
gazebo_mo...
gazebo_osm
husky_ws
Music
pcg_gazebo
Pictures
placement
Public
Scene
ViewLayer
Scene Collection
Collection
Plane
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 10.000
Y 10.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
Custom Properties
Sample Size 1 px
Object View Select Add Node Use Nodes
Slot 1
soil
Plane Plane.001 soil



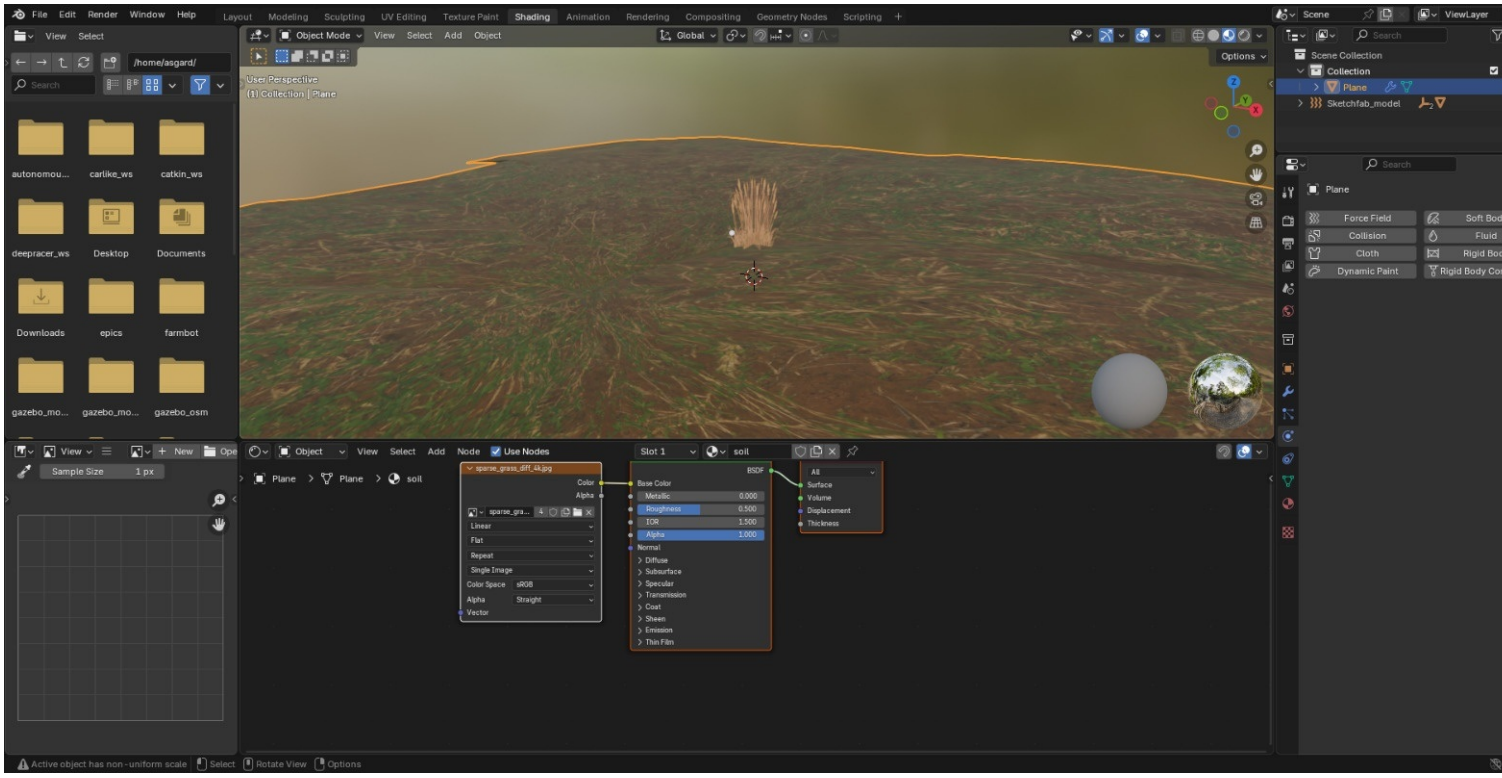


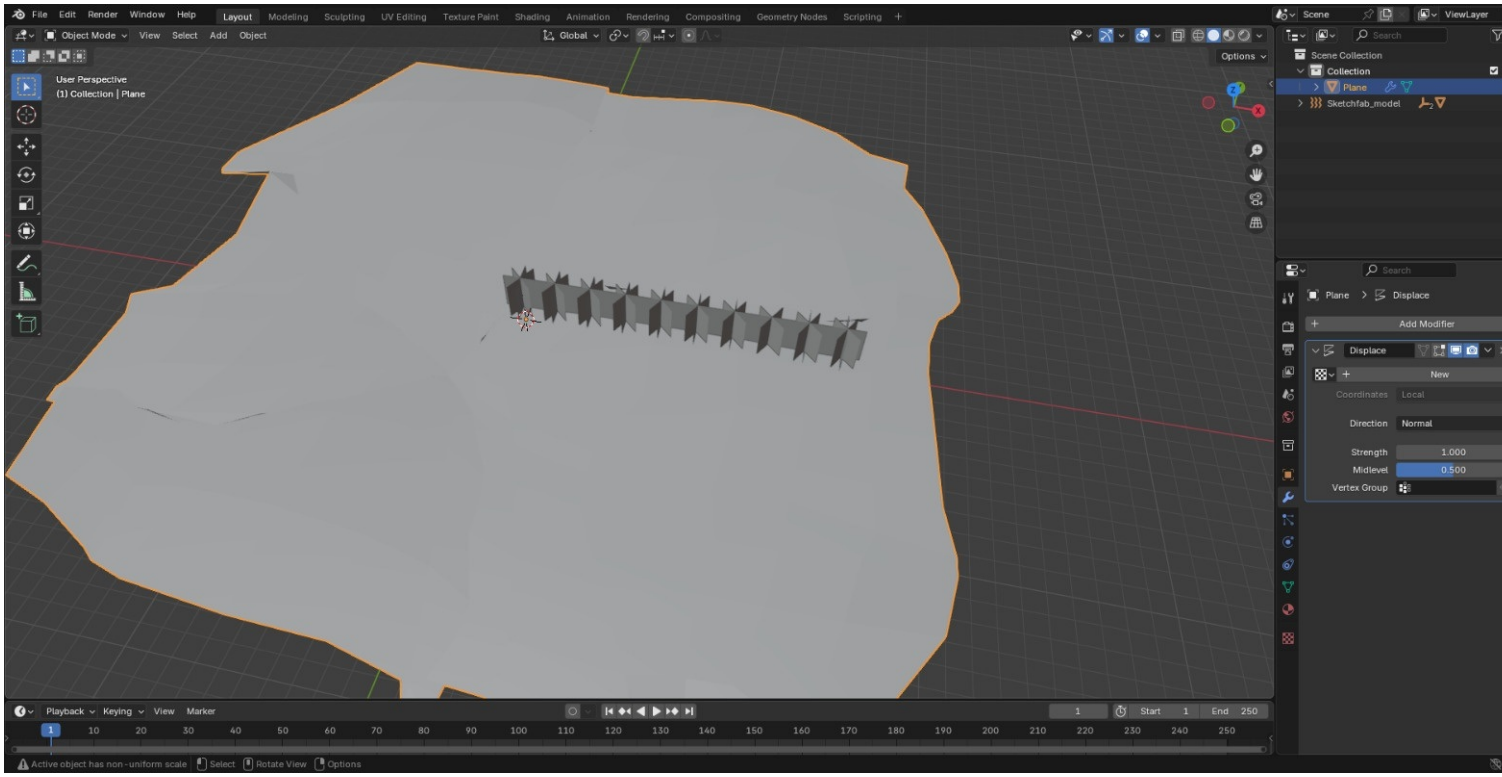
Action Item 4: Creating Crop rows in Blender – 3 hour(s).

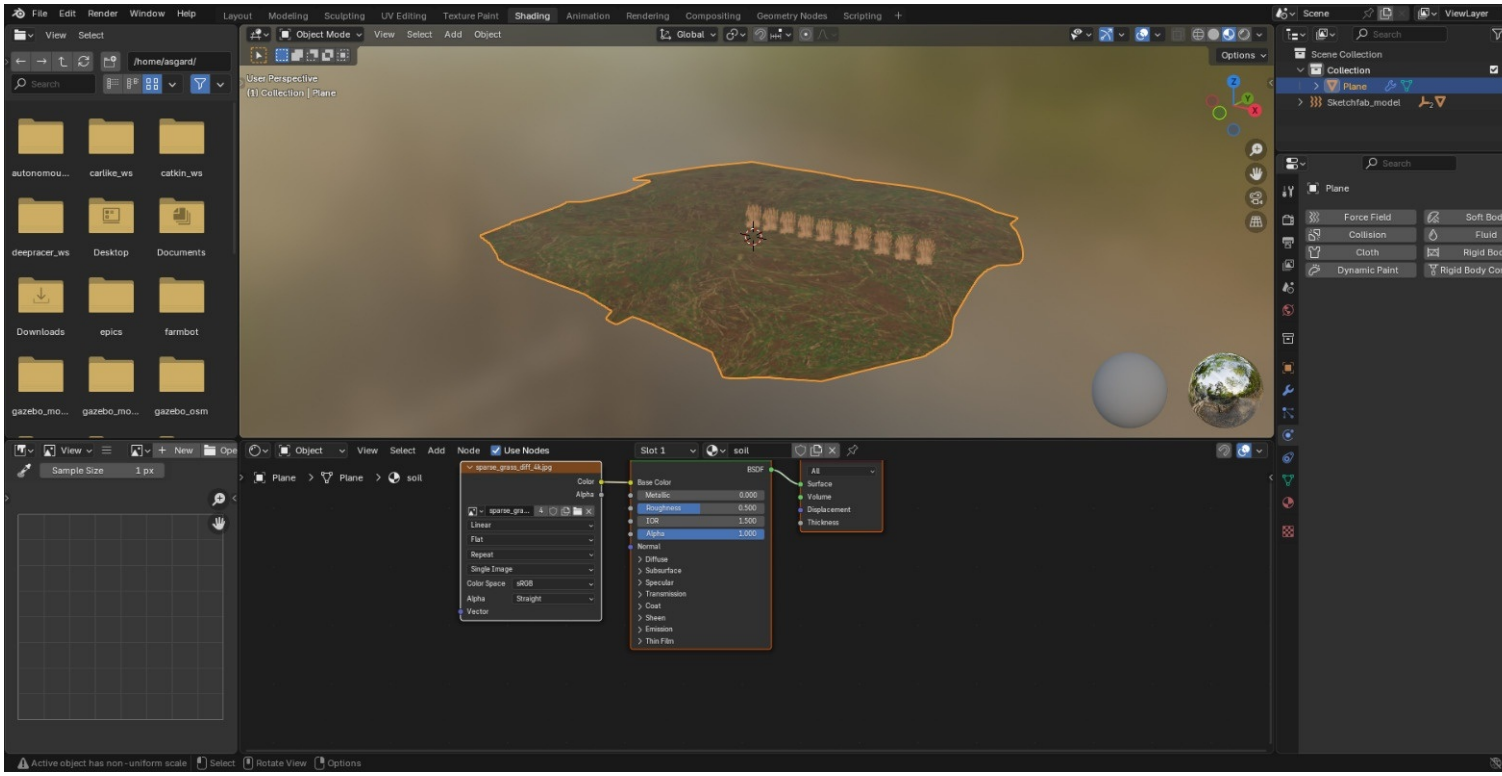
Project Work Summary

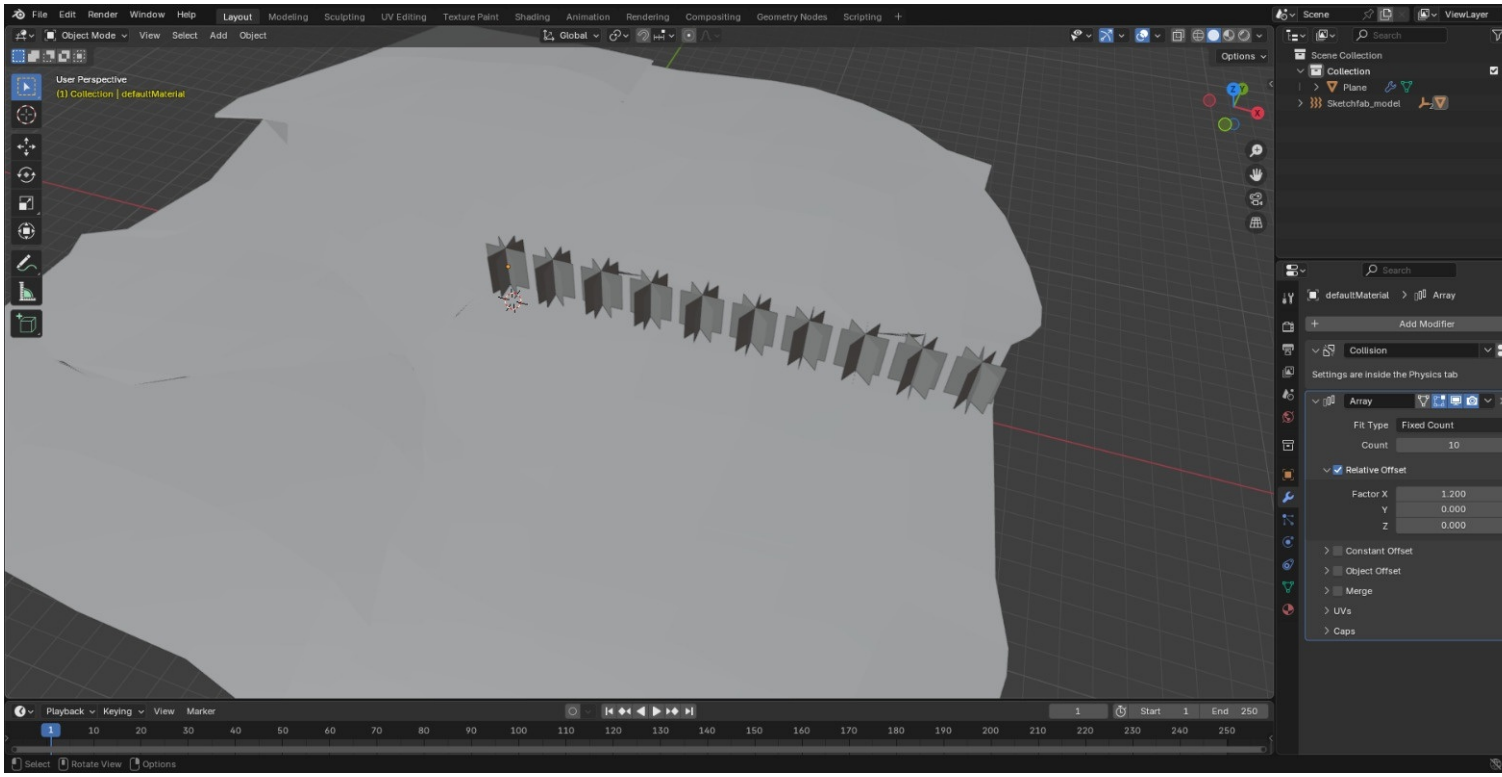
- To make the farmland look more realistic for Gazebo simulations, I worked on adding furrows (plowed fields) and crops (wheat) in Blender. This step not only improves the visual accuracy of the environment but also ensures that Husky's navigation is tested in a setting that mimics real-world conditions.
- Creating Furrows (Plowed Fields)
  - Preparing the Terrain
    - I started by selecting the sculpted farmland terrain and applied a Displacement Modifier to add the furrow details.
  - Generating the Furrow Pattern
    - To create the characteristic ridges and grooves of a plowed field, I used one of two approaches:
      - Custom Height Map: I designed a black-and-white height map with wavy lines—white areas representing raised ridges and black areas forming the grooves.
      - Pre-made Textures: As an alternative, I searched for seamless furrow textures from resources like Poly Haven or OpenTopography.
- Applying the Displacement Modifier
  - Loaded it as a texture in the Displacement Modifier.
  - Adjusted the Strength to control the depth of the furrows and fine-tuned the Midlevel to set the base elevation.
  - Increased subdivisions to ensure smooth and realistic furrow shapes.
- This gave the farmland a properly plowed appearance, making it more suited for an agricultural simulation.
- Adding Crops (Wheat)
  - Finding the Right Wheat Model

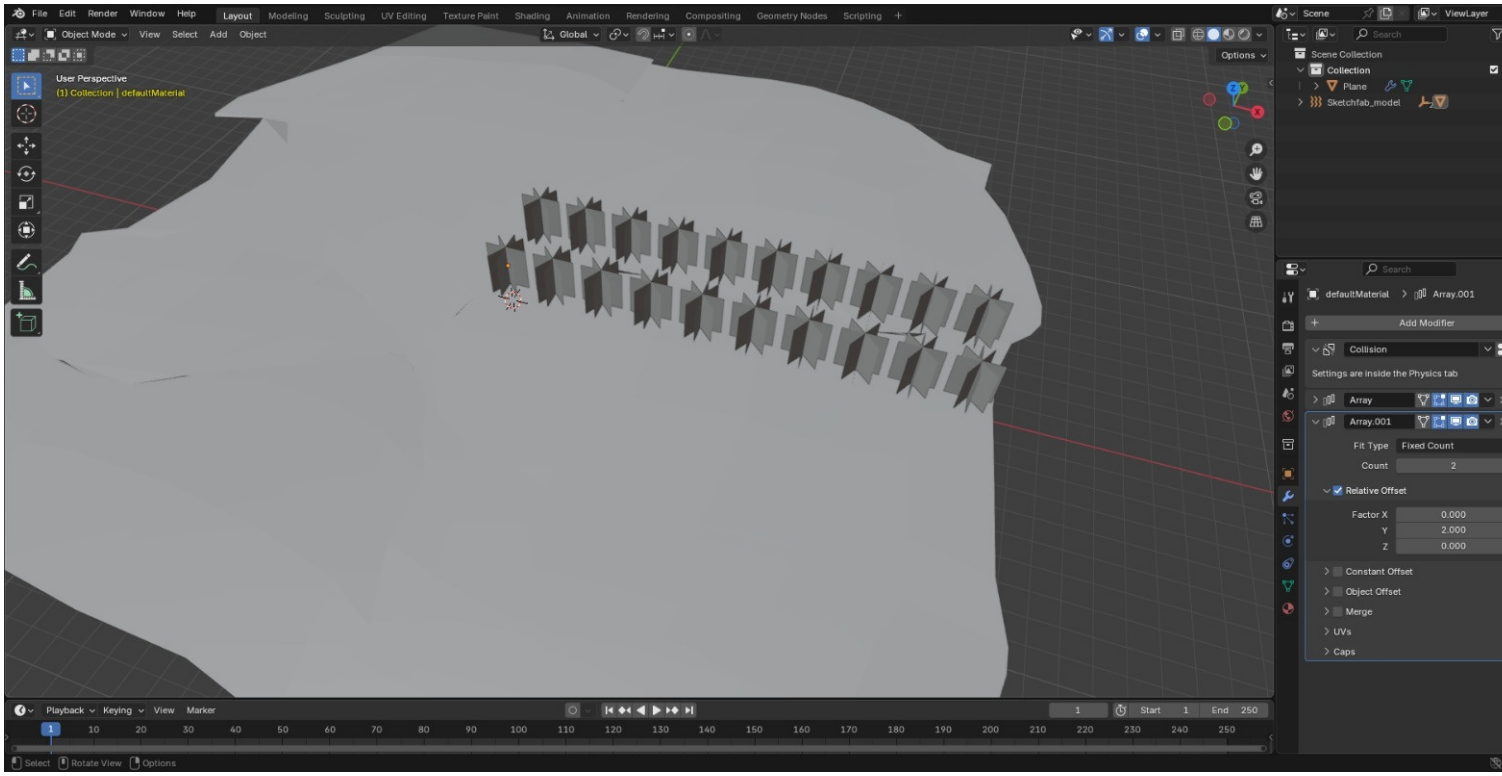
        - To add wheat crops, I searched for low-poly wheat models on Sketchfab, ensuring compatibility with Gazebo for real-time simulation. I found the "Wheat Field" model by Buncic, which featured both ripe and green wheat variations. I downloaded the model in FBX/OBJ format and saved it locally.
    - Importing and Adjusting the Wheat Model
        - Imported the wheat model into Blender (File → Import → FBX/OBJ).
        - Scaled it to match the terrain's dimensions.
        - Optimized the geometry if needed, reducing complexity for better performance in Gazebo.
    - Distributing the Wheat Using a Particle System
        - To spread the wheat across the farmland, I used Blender's Particle System:
        - Selected the terrain and added a new Particle System.
        - Assigned the wheat model as the particle object.
        - Tweaked the settings to make it look more natural:
            - Density: Controlled how closely crops were spaced (e.g., 0.5 for sparser fields).
            - Scale Randomness: Added slight variations (0.2–0.5) to make the wheat look less uniform.
            - Rotation: Enabled Random and Dynamic rotation to mimic the effects of wind.

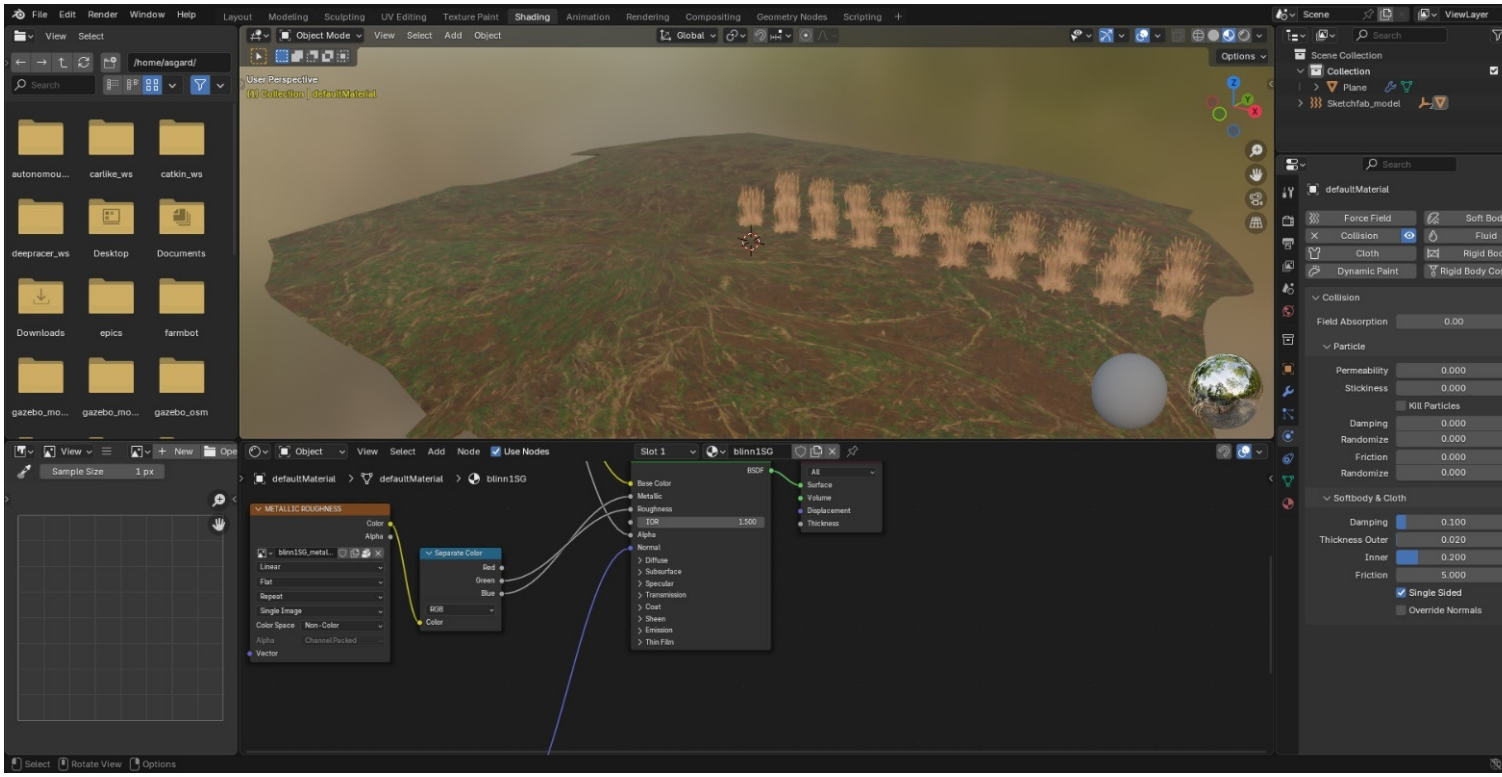


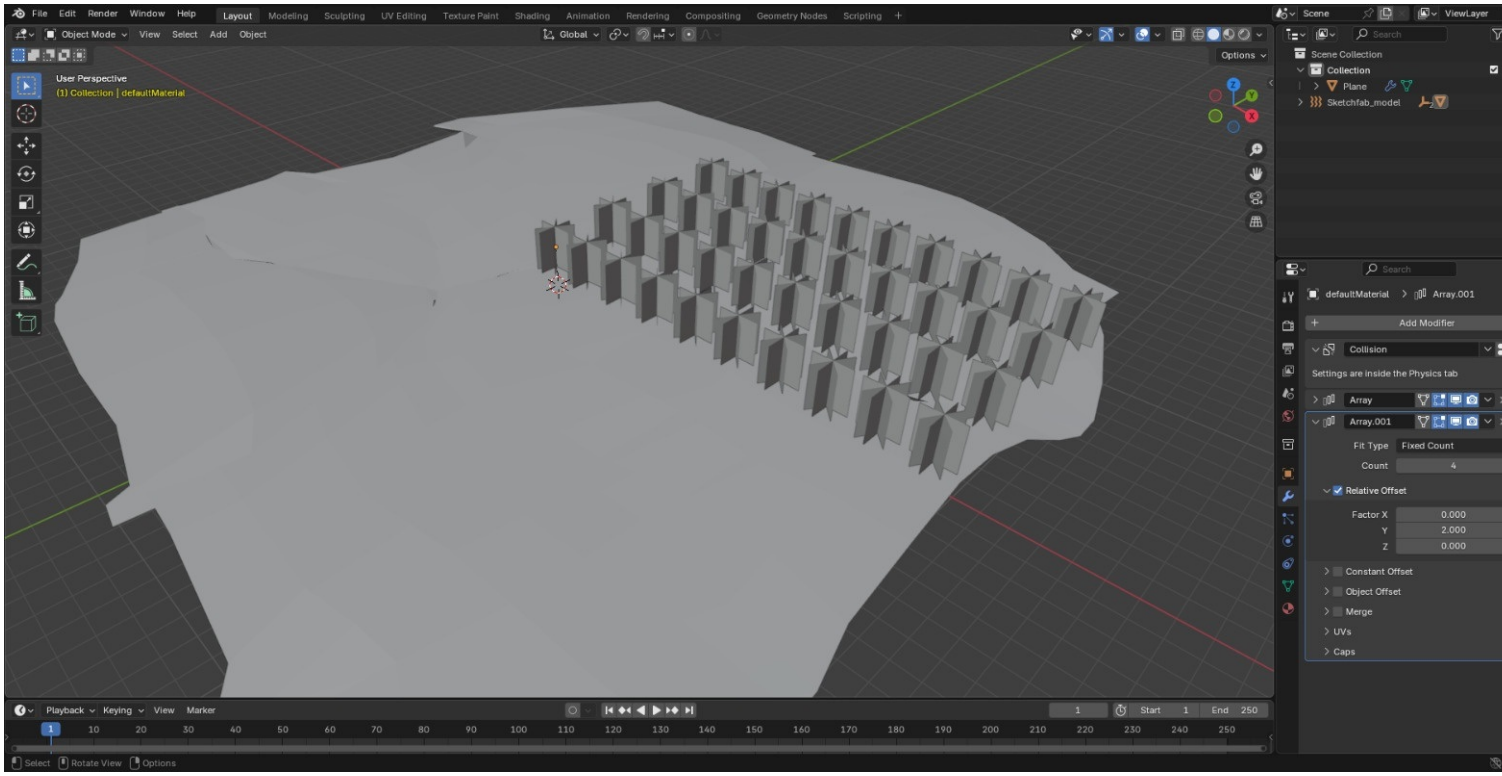


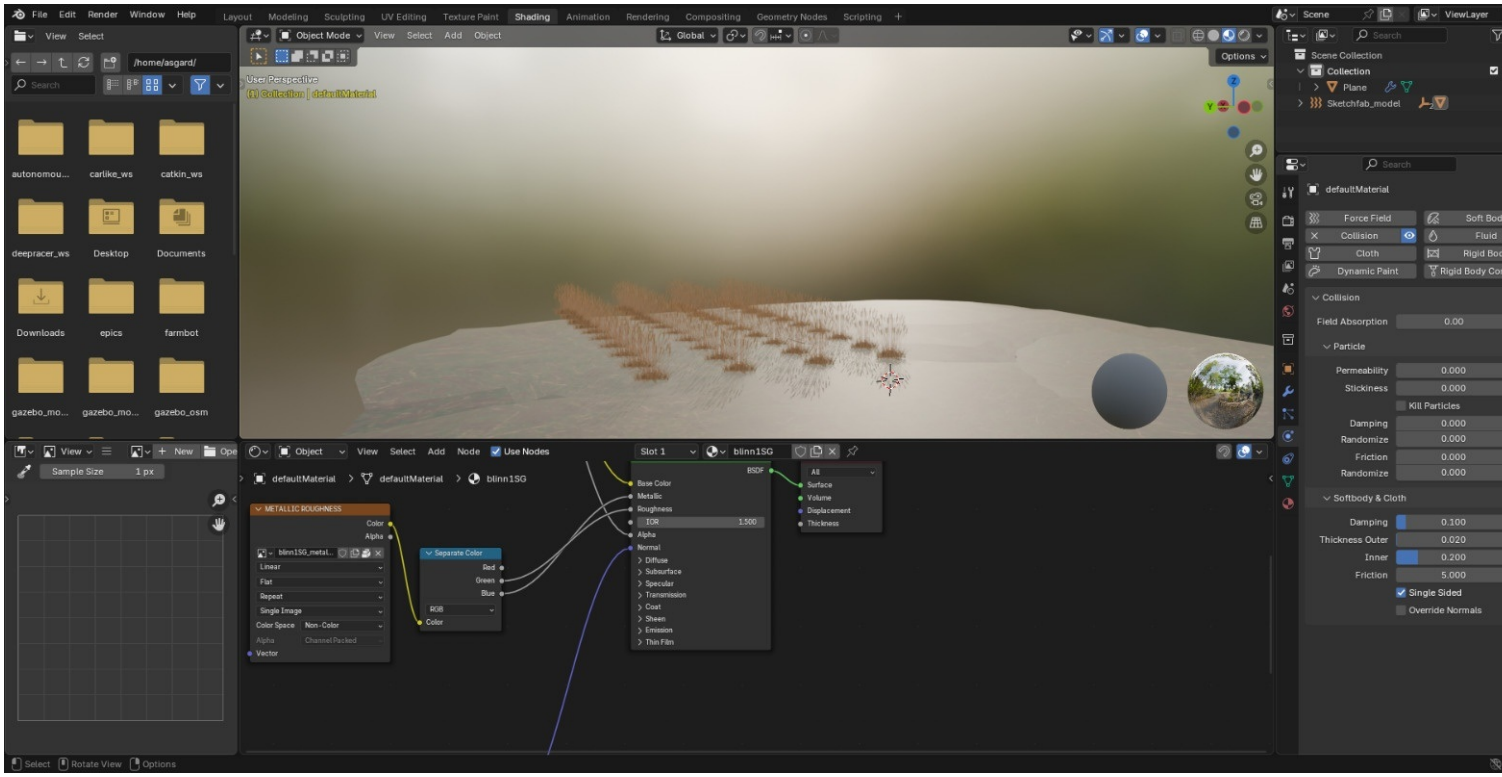

File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | Plane
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
Force Field
Collision
Cloth
Dynamic Paint
Rigid Body
autonomou...
carlike_ws
catkin_ws
deepracer_ws
Desktop
Documents
Downloads
epics
farmbot
gazebo_mo...
gazebo_mo...
gazebo_osm
Object View Select Add Node Use Nodes
Slot 1
soil
Plane
Base Color
Metallic
Roughness
IOR
Alpha
Normal
Surface
Volume
Displacement
Thickness
Sample Size
1 px
Active object has non-uniform scale
Select
Rotate View
Options

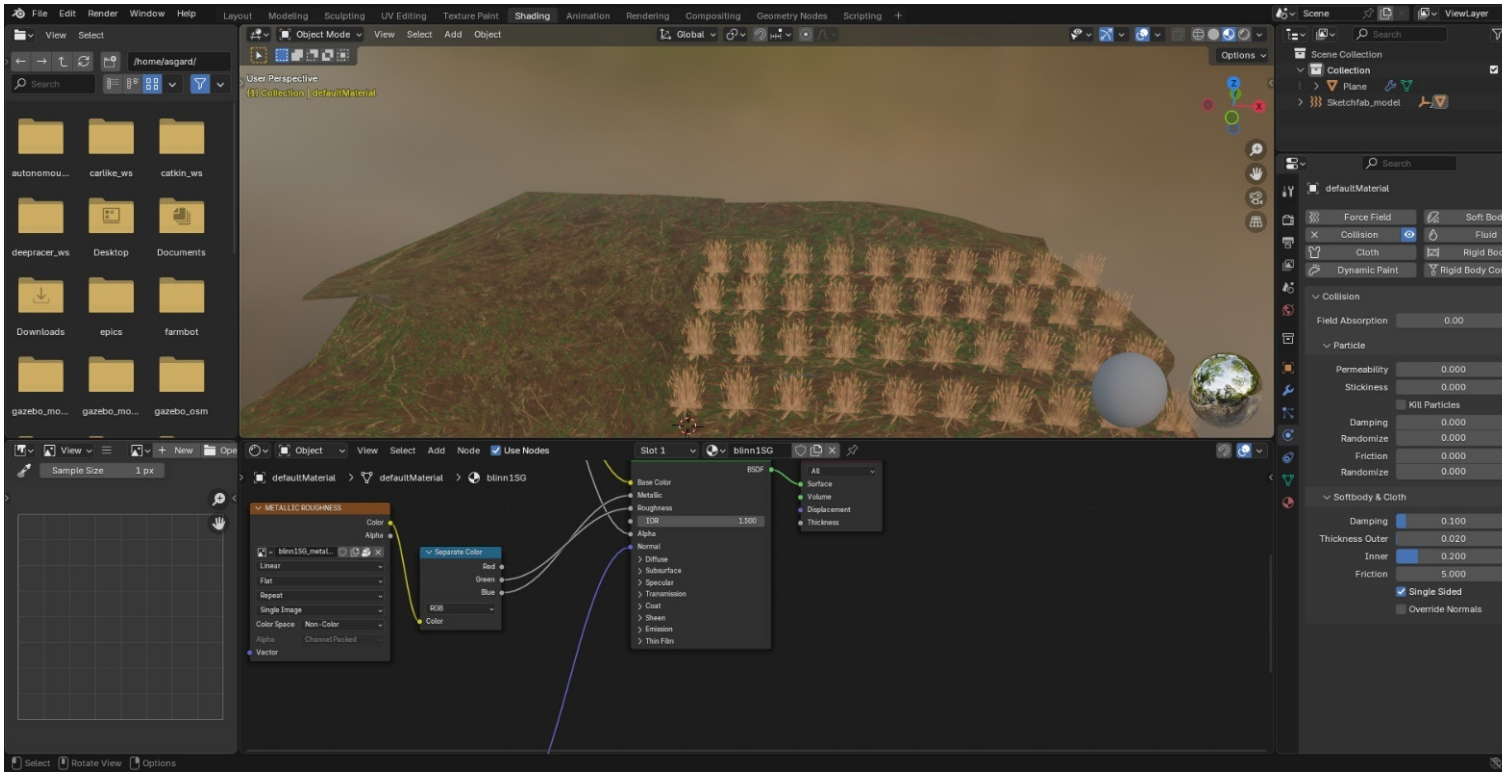

File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | defaultMaterial
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
defaultMaterial
Array
Add Modifier
Collision
Settings are inside the Physics tab
Fit Type
Fixed Count
Count
10
Relative Offset
Factor X
1.200
Y
0.000
Z
0.000
Constant Offset
Object Offset
Merge
UVs
Caps
Playback Keying View Marker
Start
1
End
250
Select
Rotate View
Options


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | defaultMaterial
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
defaultMaterial
Array.001
Add Modifier
Collision
Settings are inside the Physics tab
Array
Array.001
Fit Type
Fixed Count
Count
2
Relative Offset
Factor X
0.000
Y
2.000
Z
0.000
Constant Offset
Object Offset
Merge
UVs
Caps
Playback Keying View Marker
Start
1
End
250
Select
Rotate View
Options


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
View Select
/home/asgard/
autonomou...
carlike_ws
catkin_ws
deepracer_ws
Desktop
Documents
Downloads
epics
farmbot
gazebo_mo...
gazebo_mo...
gazebo_osm
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | defaultMaterial
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
defaultMaterial
Force Field
Collision
Cloth
Dynamic Paint
Collision
Field Absorption
0.00
Particle
Permeability
0.000
Stickiness
0.000
Kill Particles
Damping
0.000
Randomize
0.000
Friction
0.000
Randomize
0.000
Softbody & Cloth
Damping
0.100
Thickness Outer
0.020
Inner
0.200
Friction
5.000
Single Sided
Override Normals
View New
Sample Size
1 px
Object View Select Add Node Use Nodes
Slot 1
blinn1SG
defaultMaterial
defaultMaterial
blinn1SG
METALLIC ROUGHNESS
Color
Alpha
Linear
Flat
Repeat
Single Image
Color Space
Non-Color
Vector
Separate Color
Red
Green
Blue
RGB
Color
BSDF
Base Color
Metallic
Roughness
IOR
1.500
Alpha
Normal
Diffuse
Subsurface
Specular
Transmission
Coat
Sheen
Emission
Thin Film
All
Surface
Volume
Displacement
Thickness
Select
Rotate View
Options


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | defaultMaterial
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
defaultMaterial
Array.001
Add Modifier
Collision
Settings are inside the Physics tab
Array
Array.001
Fit Type
Fixed Count
Count
4
Relative Offset
Factor X
0.000
Y
2.000
Z
0.000
Constant Offset
Object Offset
Merge
UVs
Caps
Playback Keying View Marker
Start 1
End 250
Select
Rotate View
Options

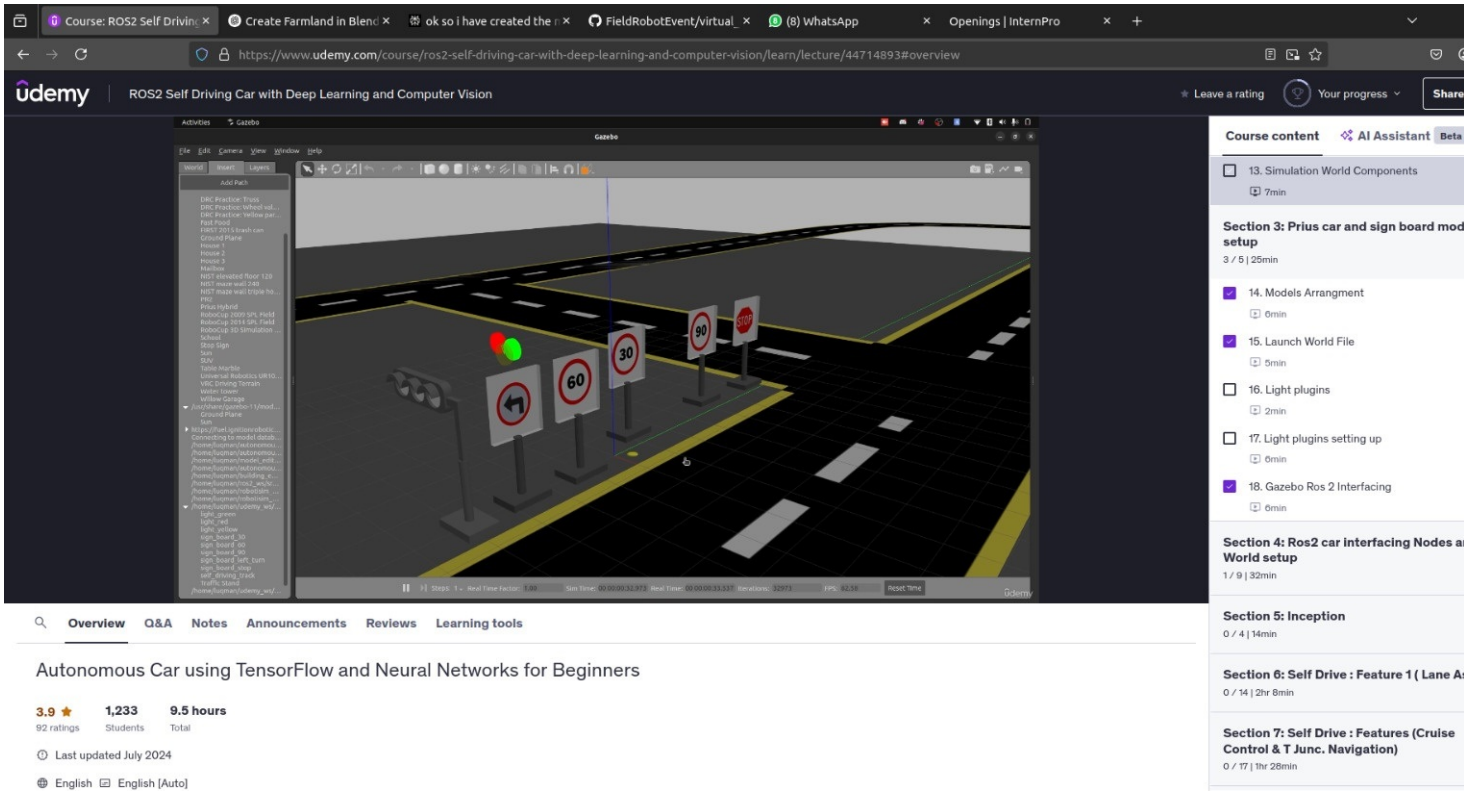

File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
View Select
/home/asgard/
autonomou...
carlike_ws
catkin_ws
deepracer_ws
Desktop
Documents
Downloads
epics
farmbot
gazebo_mo...
gazebo_mo...
gazebo_osm
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | defaultMaterial
Scene
ViewLayer
Scene Collection
Collection
Plane
Sketchfab_model
defaultMaterial
Force Field
Collision
Cloth
Dynamic Paint
Fluid
Collision
Field Absorption
0.00
Particle
Permeability
0.000
Stickiness
0.000
Kill Particles
Damping
0.000
Randomize
0.000
Friction
0.000
Randomize
0.000
Softbody & Cloth
Damping
0.100
Thickness Outer
0.020
Inner
0.200
Friction
5.000
Single Sided
Override Normals
View New
Sample Size
1 px
Object View Select Add Node Use Nodes
Slot 1
blinn1SG
defaultMaterial
defaultMaterial
blinn1SG
METALLIC ROUGHNESS
Color
Alpha
Linear
Flat
Repeat
Single Image
Color Space
Non-Color
Vector
Separate Color
Red
Green
Blue
RGB
Color
BSDF
Base Color
Metallic
Roughness
IOR
1.500
Alpha
Normal
Diffuse
Subsurface
Specular
Transmission
Coat
Sheen
Emission
Thin Film
All
Surface
Volume
Displacement
Thickness
Select
Rotate View
Options



Action Item 5: Procedural Generation and Blender Integration for Farmland Simulation – 3 hour(s).

Research

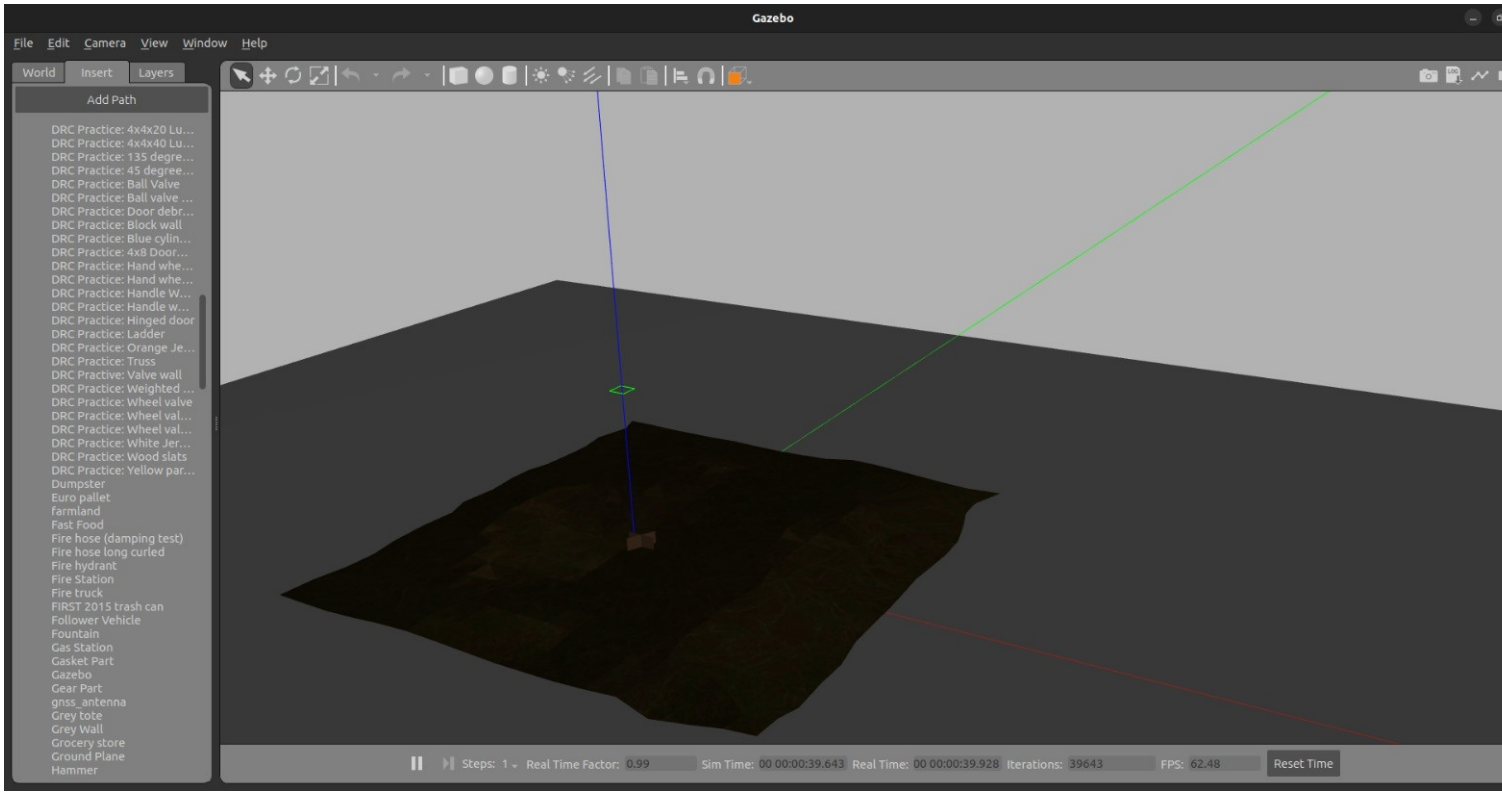
- https://github.com/FieldRobotEvent/virtual_maize_field
- Procedural Generation and Blender Integration for Farmland Simulation
- Summary of Report
  - The generate_world.py script automates farmland creation by defining parameters like: Row length (default: 12m), Plant spacing (0.13–0.19m) and Ground elevation (up to 0.2m). To add realism, the script introduces randomized plant placement errors (±0.02m) and holes in crop rows, mimicking real-world inconsistencies. This removes the need for manual design, making robotic simulation setup much more efficient.
  - The tool offers granular control over the environment: Row Design: Rows can be straight, curved, or sinusoidal, with curvature adjustable up to 1.57 rad. Ground Features: Users can set ditch depths (0.3m), headland size (2m), and ground resolution (0.02m/pixel) to create irrigation channels and field boundaries. Vegetation & Obstacles: Plants can have randomized heights (0.3–0.6m), and extra elements like weeds or litter (e.g., nettles, coke cans) can be added to test navigation in cluttered fields.
  - The generated worlds are fully compatible with ROS 2 Humble, supporting both Gazebo Classic and Ignition Gazebo. The package includes pre-configured launch files (e.g., jackal_simulation.launch), making it easy to integrate Clearpath Jackal and other agricultural robots for navigation testing.
- Relation to Project
  - Previously, designing a virtual farmland required manually sculpting terrain and placing crops in Blender, which was time-consuming. With this tool, I could automate base map generation, creating randomized rows, holes, and elevation changes instantly. For example: Using fre21_task_2, I generated straight rows with holes. The fre22_task_navigation preset added curved rows and obstacles, perfect for testing Husky's path-planning algorithms.
  - While the procedural tool created a solid base, I imported the .world file into Blender to refine details like: Adding furrows using displacement modifiers for a more natural plowed look. Manually adjusting crop placements to test specific navigation scenarios.
  - This package's native ROS 2 support solved previous integration challenges I faced with ROS 1-based worlds. I could directly spawn the Husky robot into the generated world using robot_spawner.launch, streamlining navigation algorithm testing.
- Motivation for Research
  - Creating farmland in Blender manually takes hours—this tool cuts setup time by 90%. I could iterate quickly to test different layouts, such as: Dense maize fields (--plant_spacing_max 0.19, --rows_count 8) generated in under 10 seconds. Sparse crop environments to evaluate Husky's pathfinding in open fields.
  - By tweaking parameters like: ground_ditch_depth 0.3, I simulated challenging terrains with deeper ditches, testing Husky's slope-handling capabilities. weed_types dandelion, I introduced random obstacles, helping refine Husky's obstacle detection accuracy.
  - With the terrain automatically generated, I could spend more time improving navigation algorithms rather than manually modeling environments. Predefined configurations like fre22_task_mapping_mini served as benchmark test worlds, while Blender allowed fine-tuning for more complex features like textured furrows.

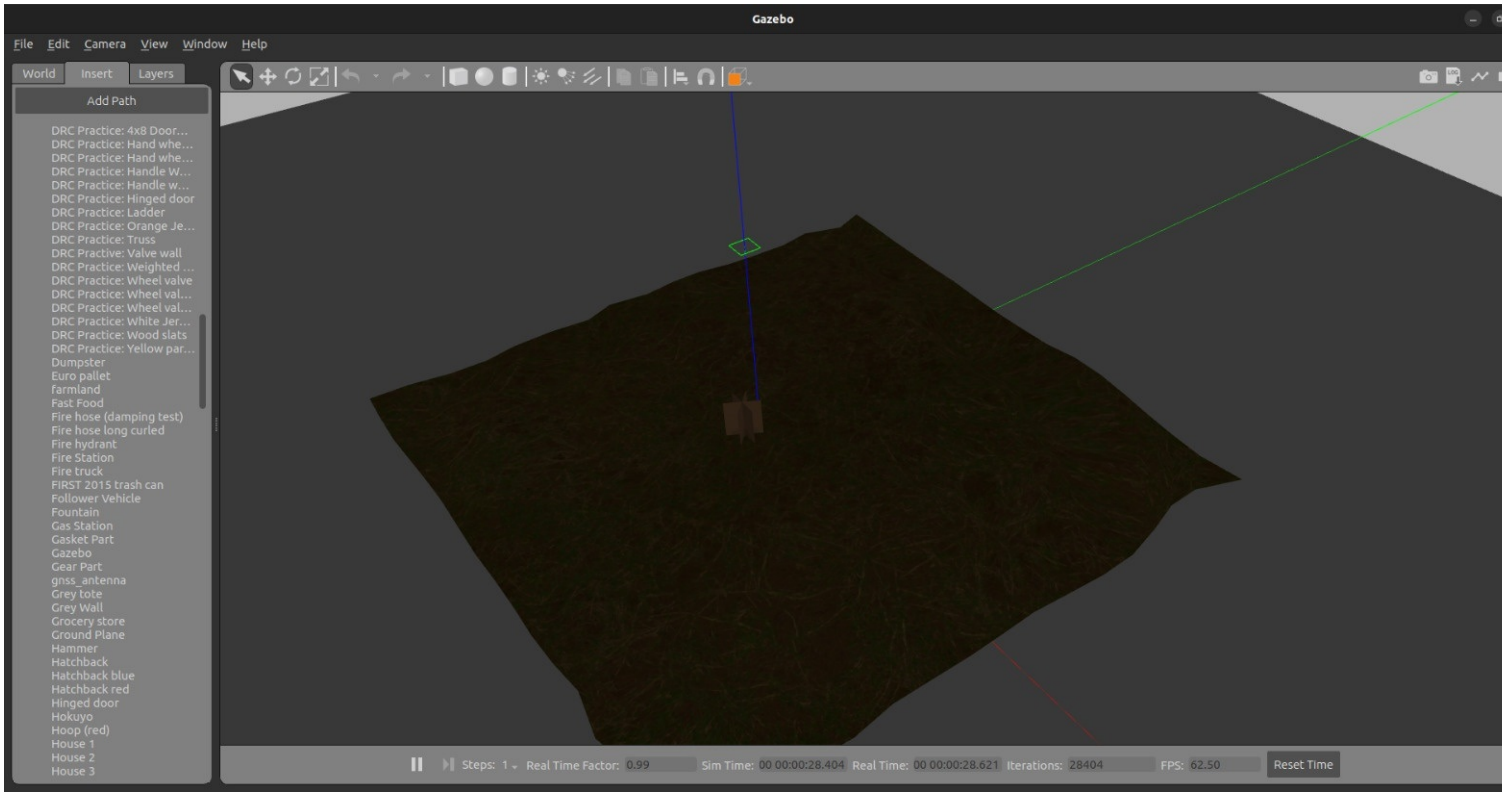


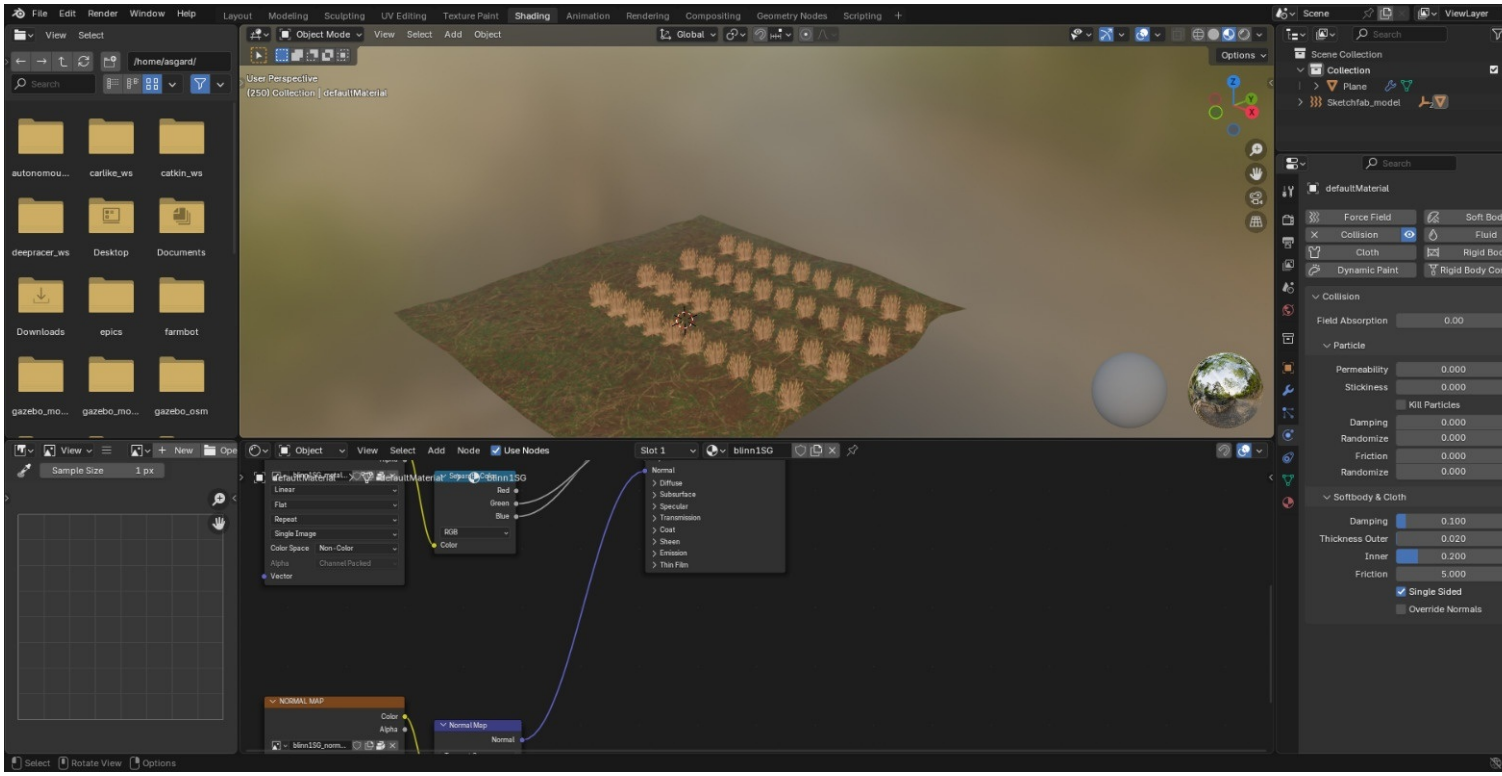
Action Item 6: Exporting a Blender Map to Gazebo – 3 hour(s).

Project Work Summary

- After finalizing the farmland terrain in Blender, the next step was to export it to Gazebo while ensuring all dependencies—such as wheat, grass, and textures—were properly integrated. This setup allows Husky to navigate the map in simulation without rendering or pathing issues.
- Steps to Export and Import the Map
  - Apply Transforms in Blender
    - Before exporting, I selected the entire terrain model (press A) and applied all transformations (Ctrl + A → All Transforms). This step resets the model's scale, rotation, and location to default values, preventing unexpected behavior in Gazebo.
  - Export as Collada (.dae)
    - Used File → Export → Collada (.dae).
    - Enabled Include UV Textures and Include Material Textures to ensure all texture paths were embedded.
    - Saved the .dae file in a dedicated folder along with all texture files (e.g., soil, grass, wheat).
  - Create Model Files
    - To make the terrain compatible with Gazebo, I created two key files:
    - Model Config File (model.config) – Includes metadata like the map name, version, author, and description, helping Gazebo recognize the model.
    - SDF File (model.sdf) – Defines the terrain's geometry, textures, and dependencies (such as wheat and grass models). The <mesh> tags reference the .dae file, while <pose> values ensure proper alignment in the simulation.


Gazebo
File Edit Camera View Window Help
World Insert Layers
Add Path
DRC Practice: 4x4x20 Lu...
DRC Practice: 4x4x40 Lu...
DRC Practice: 135 degre...
DRC Practice: 45 degree...
DRC Practice: Ball Valve
DRC Practice: Ball valve ...
DRC Practice: Door debr...
DRC Practice: Block wall
DRC Practice: Blue cylin...
DRC Practice: 4x8 Door...
DRC Practice: Hand whe...
DRC Practice: Hand whe...
DRC Practice: Handle W...
DRC Practice: Handle w...
DRC Practice: Hinged door
DRC Practice: Ladder
DRC Practice: Orange Je...
DRC Practice: Truss
DRC Practive: Valve wall
DRC Practice: Weighted ...
DRC Practice: Wheel valve
DRC Practice: Wheel val...
DRC Practice: Wheel val...
DRC Practice: White Jer...
DRC Practice: Wood slats
DRC Practice: Yellow par...
Dumpster
Euro pallet
farmland
Fast Food
Fire hose (damping test)
Fire hose long curled
Fire hydrant
Fire Station
Fire truck
FIRST 2015 trash can
Follower Vehicle
Fountain
Gas Station
Gasket Part
Gazebo
Gear Part
gnss_antenna
Grey tote
Grey Wall
Grocery store
Ground Plane
Hammer
Steps: 1 Real Time Factor: 0.99 Sim Time: 00 00:00:39.643 Real Time: 00 00:00:39.928 Iterations: 39643 FPS: 62.48
Reset Time

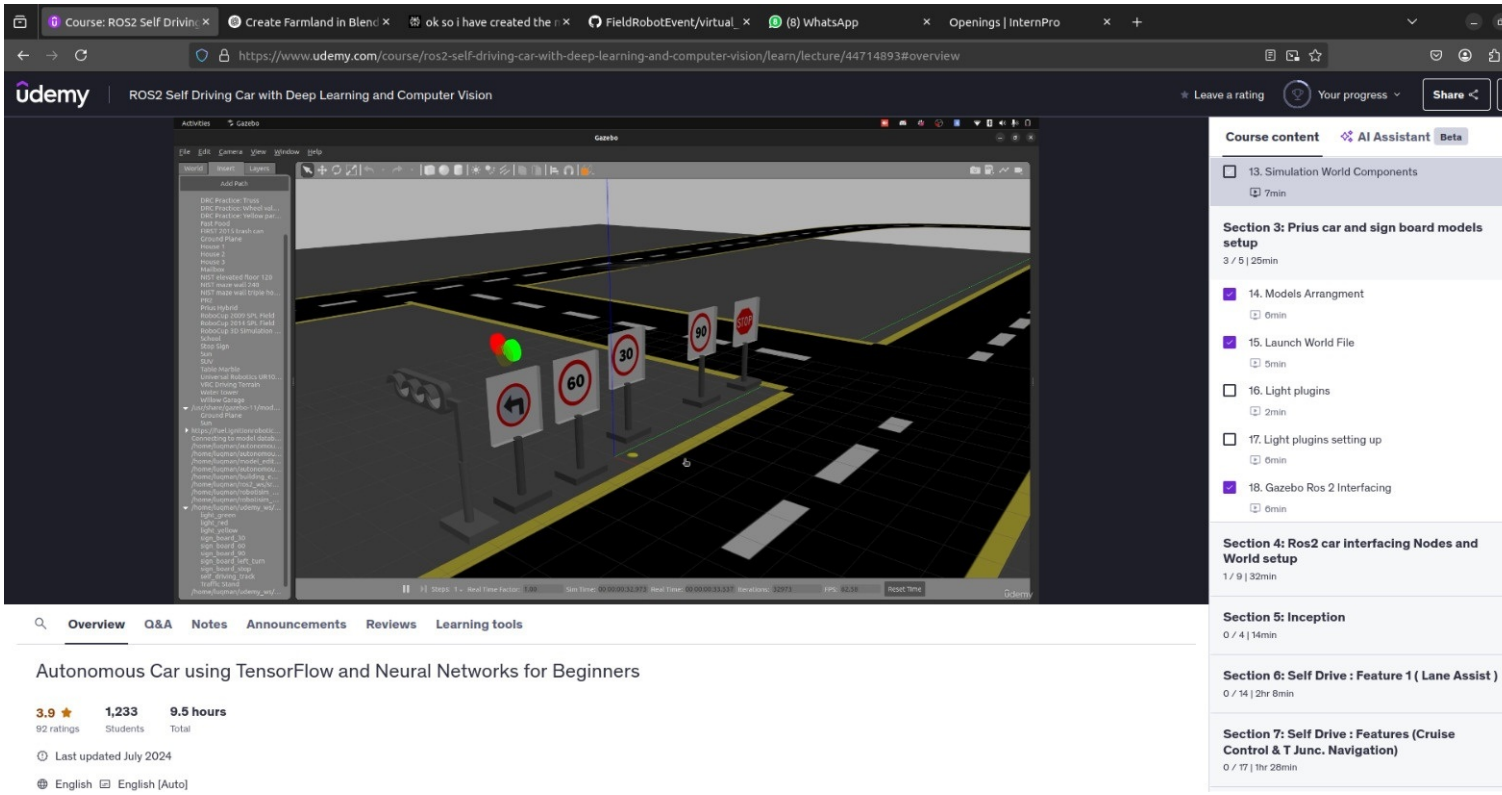

Gazebo
File Edit Camera View Window Help
World Insert Layers
Add Path
DRC Practice: 4x8 Door...
DRC Practice: Hand whe...
DRC Practice: Hand whe...
DRC Practice: Handle W...
DRC Practice: Handle w...
DRC Practice: Hinged door
DRC Practice: Ladder
DRC Practice: Orange Je...
DRC Practice: Truss
DRC Practive: Valve wall
DRC Practice: Weighted ...
DRC Practice: Wheel valve
DRC Practice: Wheel val...
DRC Practice: Wheel val...
DRC Practice: White Jer...
DRC Practice: Wood slats
DRC Practice: Yellow par...
Dumpster
Euro pallet
farmland
Fast Food
Fire hose (damping test)
Fire hose long curled
Fire hydrant
Fire Station
Fire truck
FIRST 2015 trash can
Follower Vehicle
Fountain
Gas Station
Gasket Part
Gazebo
Gear Part
gnss_antenna
Grey tote
Grey Wall
Grocery store
Ground Plane
Hammer
Hatchback
Hatchback blue
Hatchback red
Hinged door
Hokuyo
Hoop (red)
House 1
House 2
House 3
Steps: 1 Real Time Factor: 0.99 Sim Time: 00 00:00:28.404 Real Time: 00 00:00:28.621 Iterations: 28404 FPS: 62.50
Reset Time



Action Item 7: Weekly Plan for Next Week – 1 hour(s).

Project Work Summary

- Update the map further to make sure the crops created in the blender translates properly to the gazebo map.
- Refine the husky_mover.py script in the husky_control package to improve navigation efficiency in the FarmWithCropRow.world.
- Update Husky's SDF model and crop collision properties to prevent unrealistic interactions.
- Test Husky's ability to traverse corner-to-corner in procedurally generated maps.
- Configure Nav2 and test SLAM-based navigation using the husky_cartographer_navigation package.



Action Item 8: Report Writing – 1 hour(s).

Project Work Summary

- Created word document layout to write contents of the weekly progress.

- Created relevant subsections in the epicspro website and documented 20 hours of weekly progress.
- Collected relevant documents research papers, relevant links and company's objective from their portal.