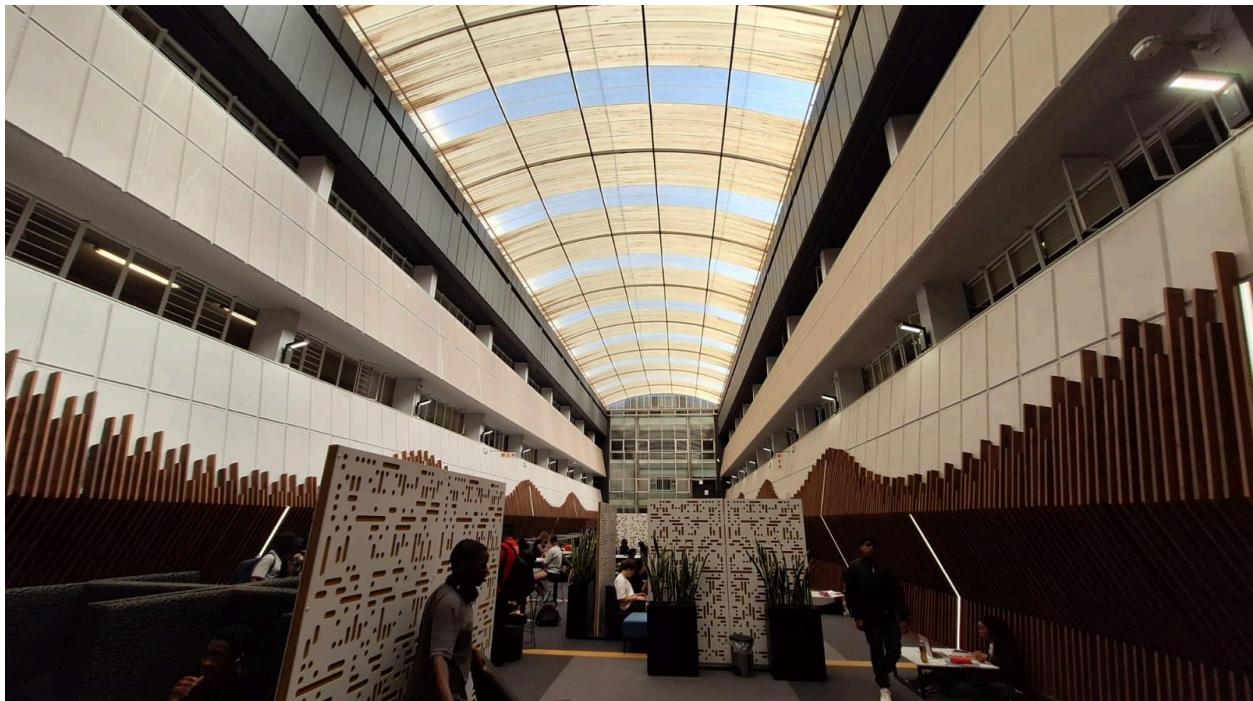


COS 344

CantCode.Com

Homework Assignment Report



Group Members:

Scott Bebington	u21546216
Charlize Hanekom	u22487222
Tim Whitaker	u22744968

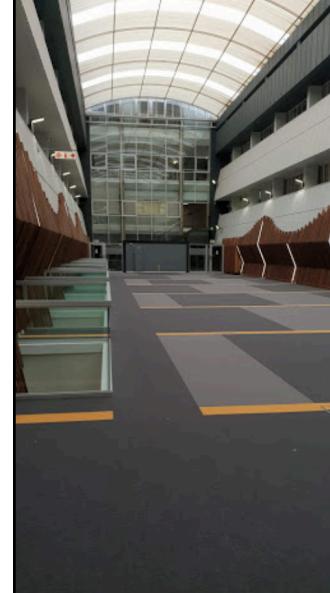
Introduction

Role division

Scott Bebington will be focused on the generation of the objects on the floor of the kiosk such as the couches and tables, the other members will naturally also work on these objects. Scott will also be responsible for the drone movement including the rotations.

Charlize Hanekom will focus on the floor and ceiling generation, such as the textures for the floor, the dome shape of the roof, the glass panels of the roof as well as the light shining through it. Charlize will also focus on the lighting aspects of the kiosk.

Tim Whitaker will be focusing on the rendering of the walls of the kiosk, this includes the general shape of the walls, the wooden slats on the North and South walls, the concrete textures of the walls, the glass panes with offices behind them as well as the lights that are on the second floor. Tim will also take point on the texture mapping.



For all the objects in the IT Kiosk we will use object order rendering since it is less processing intensive and time consuming than image order rendering.

North and South Walls

North:

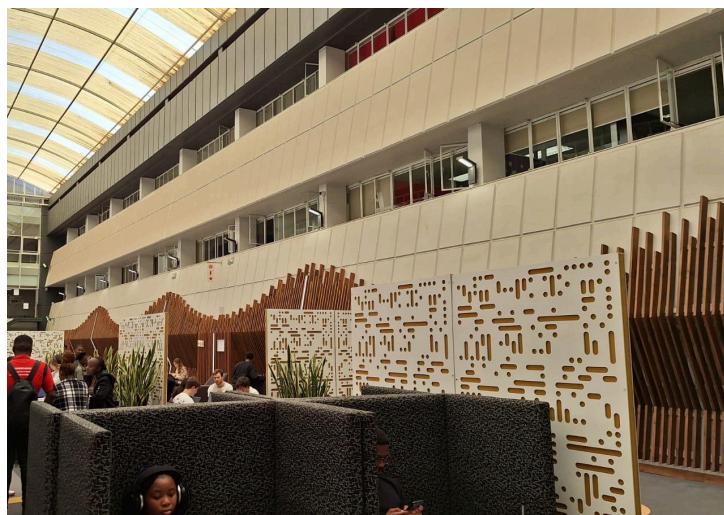
For the north wall panels, we propose using a single large 3D rectangle with the texture mapped onto it. This approach will create a realistic depiction of the wall, enhancing its visual authenticity. To optimise memory usage, backface culling will be implemented to avoid rendering surfaces that are not visible.

For the columns between the panels and the window panes, a similar 3D rectangle approach will be utilised with the same texture mapping technique. Backface culling will also be applied to these elements to maintain efficiency.

For the point light sources along the wall, their light strength is low and the room is in daylight. As a result, they will not provide much light, and no shadows will need to be calculated since the lights do not create any.

For the windows, we will use a 3D rectangle with enough transparency to see through to the offices behind. The offices will be shown with a texture applied to a simple block.

Bonus marks: We plan to implement the wooden slats on the side of the wall using a wooden texture layered over the wall texture. A displacement map will be used to create the depth of the slats.



South:

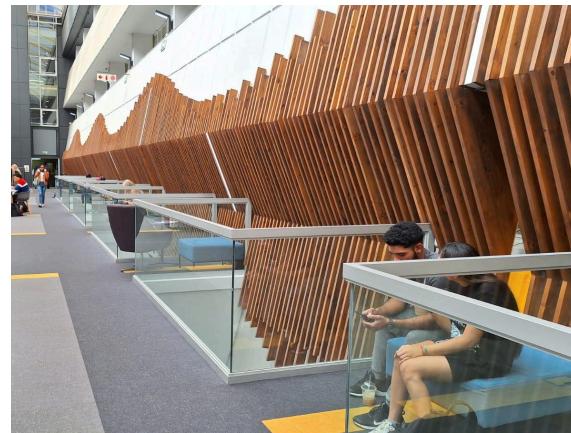
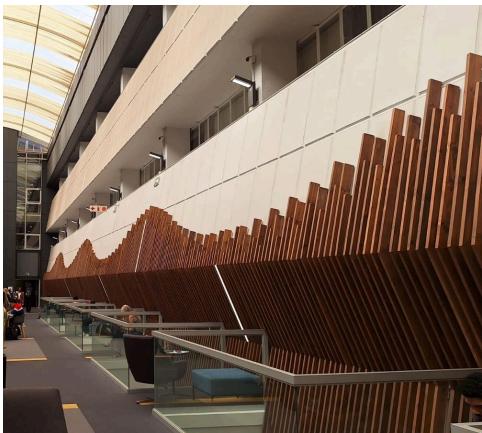
For the south wall panels are almost identical to the north wall and will therefore use the same approach, indicated by the red arrow, we propose using a single large 3D rectangle with the texture mapped onto it. This approach will create a realistic depiction of the wall, enhancing its visual authenticity. To optimise memory usage, backface culling will be implemented to avoid rendering surfaces that are not visible.

For the columns between the panels and the window panes, a similar 3D rectangle approach will be utilised with the same texture mapping technique. Backface culling will also be applied to these elements to maintain efficiency.

For the point light sources along the wall, their light strength is low and the room is in daylight. As a result, they will not provide much light, and no shadows will need to be calculated since the lights do not create any.

For the windows, we will use a 3D rectangle with enough transparency to see through to the offices behind. The offices will be shown with a texture applied to a simple block.

Bonus marks: We plan to implement the wooden slats on the side of the wall using a wooden texture layered over the wall texture. A displacement map will be used to create the depth of the slats.



East and West walls

Graphical Data Structure:

- Rectangular planes will be used for both the main wall and the glass panels.
- The doors will simply be rectangular objects that will be closed, they will be a highly opaque glass to avoid having to include details of the outside environment.
- The stairs will be created using a combination of different trapezoids.
- The back of the coffee shop will just be a simple floor with bounds checking preventing the drone from looking too far behind the coffee shop.

Windows and Offices:

- We won't render the actual office geometry behind the glass.
- Instead, a separate rectangular plane with an "office texture" will be positioned slightly behind the glass plane.
- This technique creates the illusion of depth without the rendering overhead of complex office models.

Transparency (Glass Panels):

- We'll use a single rectangular plane for each glass panel with a low alpha value in the texture to achieve transparency.
- This allows light and some background details to show through.

Reflections (Directional Light):

- We'll implement environment mapping using a cube map texture.
- This captures the scene (including the directional light) from all directions and reflects it off the glass surface, creating a realistic illusion without rendering the entire scene behind the glass.

Optimization:

- We'll combine the main wall and glass plane into a single rendered object. This reduces the number of draw calls needed, improving performance.
- We'll potentially use backface culling to avoid rendering the back faces of the walls, as they won't be visible from the camera's perspective.

West Wall Specifics

- The gap for the stairs will be created by adjusting the UV mapping of the main wall texture, effectively "cutting out" that section in the rendered image.
- Alternatively, we could use multiple rectangular planes: one for the main wall and another for the remaining section next to the stairs.



Overall Approach:

By using simple geometry, clever texturing techniques, and efficient rendering methods, we can create a visually appealing coffee shop environment without overloading the graphics card.

Roof

Graphical Data Structure:

- The roof will consist of multiple curved glass panels to achieve the dome shape.
- Behind the glass there will be a blue rectangular shape to give the impression of the sky.

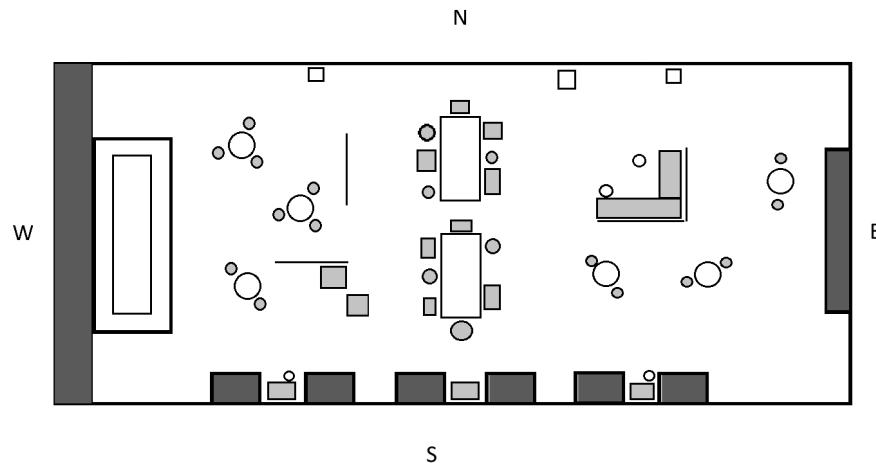
Glass Panels:

- We will use a single curved rectangular plane for each glass panel with a low alpha value in the texture to achieve transparency.
- This allows light from the sun as well as the sky to show through.
- While some of the panels will almost be completely transparent, others will be yellow coloured with a slightly higher alpha value in order to create the striped pattern that the roof of the IT kiosk has, as seen below.



Floor Plan

A general floor plan can be seen below to illustrate the possible layout of the IT kiosk. In this image the dark grey areas indicate areas that are outside of the main area such as the cutouts leading to the bottom floor on the south wall, and the area behind the coffee shop. The light grey areas are chairs and couches and the white spaces are tables and the floor.



The following components are objects that can be found in the IT kiosk:

2x Big tables - 6 seater

The tabletop will be a simple 3D rectangle with a consistent texture mapped around all six sides. The legs will also be simple 3D rectangles, coloured solid brown, as they aren't the main focus and won't be very visible.



Round tables - 4 seater

Similar to the 6-seater tables, the tabletop will feature a white cylindrical shape. The table leg will be a long, thin black cylinder, while the base will consist of a nearly flat black cylinder.

Dividers

The dividers, as depicted in the image below, will consist of tall, upright 3D rectangle. The two main faces will be mapped with an image representing the divider, while the other four less prominent faces will be



coloured black. This design will be implemented as an object to allow for easy replication across the coffee shop floor.

Cubicles

The cubicle will be constructed using five 3D rectangles arranged to resemble its structure, with each side of the rectangle mapped with the texture of the cubicle. This will also be designed as an object for easy replication across the coffee shop floor.



High tables

These objects follow the exact same design principle as the [round tables](#) but the main legs of the table are longer.

Couches

The couches will be two 3D rectangles each to form the seat and back of each couch. The couches will also have four short spherical legs. The fabric texture will be used for the seat and back of the couches and the legs will simply be black. These couches will be placed next to each other to form a larger couch.



Ottoman couches



These couches will come in different shapes (namely square and circular) as well as in different sizes (small, medium and large). The square shaped couches will simply be cubes and the circular couches will be spherical. They will be scaled to different sizes when placed in the IT kiosk scene. The ottomans will also have short black spherical legs and will have various colours.

Plastic chairs

These chairs will have two curved 3D rectangles of various colours to form the seat and back of the chair with brown cylinders as the legs.





Swirling chairs

The swirling chairs will be made of 4 3D rectangles to shape the seat, back and arm rests of these chairs. The fabric texture will also be used for these areas. Furthermore, the chairs will have one main 3D rectangle as a leg, splitting into four different legs. The legs of this chair will be a silver, metallic colour.

High chairs

These chairs will use a single curved 3D rectangle that will be grey. The legs of these chairs will be thin, black and spherical.

Coffee counter

The coffee counter for the coffee shop will consist of three black 3D rectangles placed in a U shape with the middle rectangle being wider than the other two. The counter will have another 3D rectangle on each of the others to form the granite slab, this will be mapped with the granite texture.

Materials

We will map textures to the objects for the following materials:

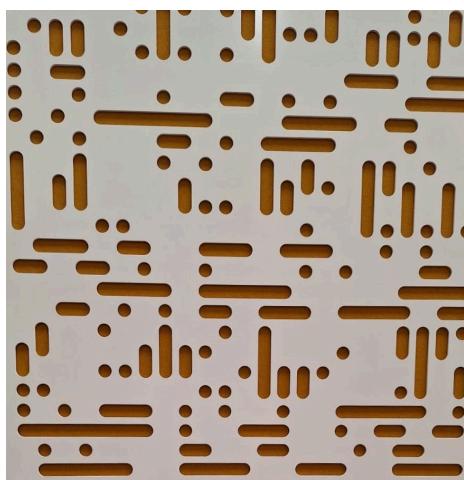
Carpet

The floor will use a repeating carpet texture (*right*) that will be mapped onto it. A diffuse map will be used and we hope to use a bump map to add some roughness. The colour is grey to match the carpet of the kiosk.



Wood

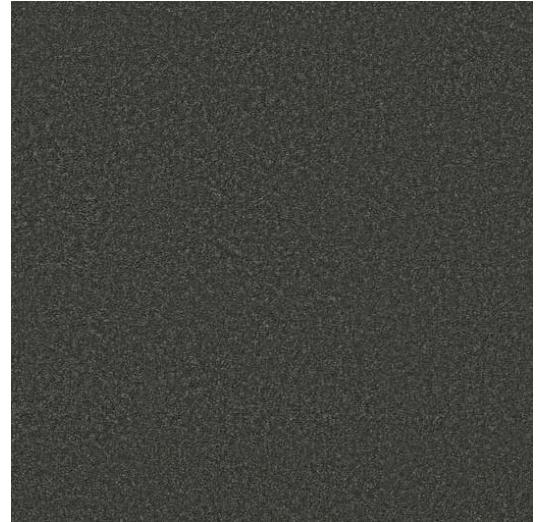
For the side panelling we will use a texture like this (*right*). We will use a diffuse map and hope to use a displacement map with the texture to create the effect of the slats coming out.



For the divider panels we will use a texture like this (*left*) using a diffuse map and hope to add a displacement map to handle the slats protruding out from the wall.

Textile/fibre

The couches and textile chairs will use this texture (*right*) for the cushioning using a diffuse map. The colour matches that of the kiosk couch colours. We would also like to use a bump map to add roughness to the surface.



Concrete

The walls will have the following textures (*below*) and a diffuse map. With the grey texture being stretched to create the needed rectangular tiles.



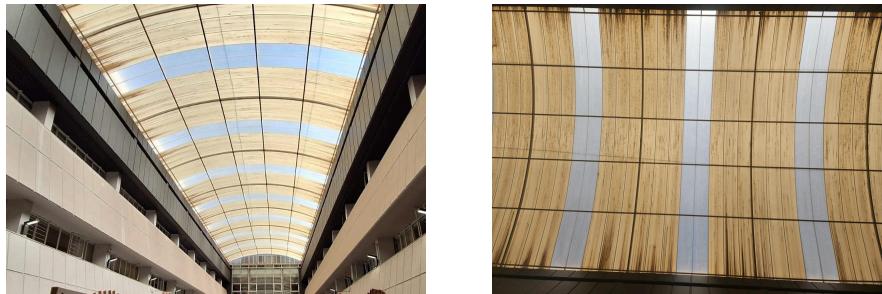
Granite

The shop's table will use the granite texture(*right*) for its surface; a diffuse map will add a glossiness to the surface.



Lighting

The transparent glass in the roof will cause the sunlight to shine into the IT kiosk, this will be treated as a directional light illuminating the entire room. This will be a yellowish light as it is the colour of the sunlight.



The strip lights in between the wooden slats will consist of multiple point lights, one for each level of the curve in the wall. The effects of these lights are not very noticeable, especially in the daytime, as such there will not be any noticeable shadows as an effect of these lights. The colour of these lights will be a bright white.



The lights that are on the second level of the room will be of rectangular shape, they will also be point lights as well as a very bright white colour.



Code design

Strategy: We'll use the strategy design pattern to handle the creation of different shapes, allowing for a flexible approach to choosing and executing shape creation methods based on specific needs.

Facade: The facade design pattern streamlines user interactions by consolidating multiple operations into a single key press, eliminating the need for users to perform multiple steps to accomplish their desired outcome.

Object-Oriented Design: Our code will be separated into classes representing different types of objects (e.g., Cube, Sphere) and use inheritance and polymorphism to manage their behaviour such as rotating and transforming the objects.

Factory Pattern: We will use factory methods or classes to create objects of different types, allowing for easy instantiation and management of objects.

Drone

Move forward, left, back and right with the WASD keys and up and down with SPACE and SHIFT keys respectively.

We will keep the drone at the origin (0,0,0) and translate the surroundings relative to the drone. This approach reduces the need for calculations to move objects to the origin and back, using this approach would speed up the program.

The drone will be able to rotate around the y-axis with the J and L keys to turn left and right as well as around the x-axis to look up and down with the I and K keys.

This would be the same approach as movement where the drone would be at the origin and rotations for the surroundings would happen around the drone, eliminating the need for additional translations to and from the origin.

Bonus marks

- Add a “sensor” to the drone that detects collisions with other objects and avoids them.
- Adding the wood siding on the walls.
- Adding extra details behind the glass on each wall such as offices.
- Changing the LED lights to be a set of smaller spotlights.
- Adding the coffee shop to the IT Kiosk, including snacks and a coffee machine.
- Adding 5 extra unique objects to the tables such as laptops, textbooks, coffee cups, bags, headphones or water bottles.
- Enabling the drone to have a night vision camera.
- Zooming in on the camera of the drone.
- Adding different lights onto the drone that will interact with objects in the IT Kiosk.
- Adding a minimap to the drone camera that will indicate the drones current position in the IT
- Giving the drone collision detection with objects in the IT Kiosk.
- Adding different camera projections to the drone camera such as fisheye projections.
- Add a colour filter to the camera of the drone, i.e. grey-scale, inverted or monochromatic.