

CC3501: Tarea 2C - Snake 3D

Camilo Núñez Barra

9 de diciembre de 2020

Resumen

Una serpiente con hambre infinita de manzanas crece cada vez que come una. La serpiente muere al caer por uno de los bordes del mapa en el cual se mueve o al colisionar consigo misma o con un obstáculo. Se implementa el patrón de arquitectura de software Modelo–Vista–Controlador, permitiendo al usuario manejar a la serpiente tridimensional edición Pokémon mediante la interfaz gráfica de PyOpenGL y GLFW.

1. Solución propuesta

Se utiliza el patrón de arquitectura de software Modelo–Vista–Controlador, de modo que, el usuario, al interactuar con la interfaz de GLFW, usa el controlador mediante ciertas teclas, que manipulan los modelos de PyOpenGL, actualizando la vista del usuario y generando así un bucle de interacción que permite la ejecución del juego. El diagrama de este patrón se muestra en la figura 1.

El cuerpo de la serpiente se modela con la estructura de datos **deque**. La serpiente tiene una cabeza cuya posición (x,y,z) , en coordenadas cartesianas, se actualiza en cada fotograma según un vector velocidad (ρ, ϕ) , en coordenadas polares, cuya magnitud ρ es constante y cuyo ángulo ϕ es modificado según las teclas de movimiento que presione el usuario. El movimiento de la serpiente duplica el elemento en la cabeza, insertándolo por delante y, para mantener constante el largo de la serpiente, elimina el elemento en la cola, a continuación actualiza la posición de cada elemento. El crecimiento solamente duplica el elemento de la cabeza, insertándolo por delante, y actualizando su posición.

Las partes de la serpiente están diseñadas con un modelo tridimensional esférico **.obj** cuya textura **.png** (figura 17) corresponde a una Poké Ball. Las manzanas están diseñadas según modelos tridimensionales **.obj** correspondientes a los Pokémon Charmander, Bulbasaur y Squirtle, con sus colores respectivos, rojo, verde y azul. Cada manzana una tiene una probabilidad 0.3 de aparición. Existe una manzana dorada, con probabilidad 0.1 de aparición, que está diseñada con un modelo tridimensional **.obj** del Pokémon Pikachu cuyo color es amarillo, además, se distingue por tener una iluminación propia muy fuerte, simulando el ataque impactrueno. Ver figuras 2-5.

Si la posición de la cabeza de la serpiente coincide dentro de un radio con la posición de la manzana, entonces se aumenta el largo en uno y la manzana reaparece en una posición aleatoria válida que no se superponga con una parte de la serpiente ni se escape de la plataforma.

Si la cabeza está fuera de la plataforma, si atraviesa su cuerpo o si choca con un obstáculo aleatorio generado al inicio del programa, entonces la serpiente muere.

2. Instrucciones de ejecución

1. Descargar Python 3.8.5 de <https://www.python.org/downloads/release/python-385/> e instalar, añadiendo a PATH.
2. Descargar la carpeta **tarea2c** de <https://github.com/camilo-nb/CC3501-tareas>. La estructura de esta carpeta se muestra en la figura 6.
3. Abrir el intérprete de comandos en la carpeta **tarea2c**. Por ejemplo, en Microsoft Windows se puede usar **cmd.exe**.
4. Instalar las librerías: **glfw==1.12.0**, **PyOpenGL==3.1.5** y **numpy==1.19.1**, mediante el comando **pip install -r requirements.txt**.
5. Ejecutar el código mediante el comando **python main.py**.
6. Usar las teclas de movimiento: **A** izquierda y **D** derecha (con respecto a la visión en primera persona), con el objetivo de que las Poké Balls capturen a los Pokémon. Se alterna la vista mediante: **E** cámara superior, **R** vista desde cabeza y **T** cámara diagonal. Se pierde al caer por el borde de la plataforma, al chocar con un obstáculo o al atraparse a sí misma, apareciendo la escena final animada, que activa la tecla **ENTER**: reiniciar el juego. La tecla **ESC** termina el programa en cualquier momento.

3. Resultados

La figura 7 muestra la ejecución de la instrucción 5, que permite crear una ventana mostrando el juego, tal como, por ejemplo, la ventana que aparece en la figura 8, que tiene una serpiente con muchas Poké Balls pues ha capturado muchos Pokémon, además de tener la cámara con una vista en primera persona detrás de la cabeza. La figura 9 muestra la visión superior del juego. La figura 10 muestra la visión diagonal del juego. La figura 11 muestra a una Poké Ball junto a Pikachu durante el impacto, que ilumina fuertemente la zona. El usuario eventualmente pierde el juego y se despliega una escena final que consta de cuatro fotogramas para animarla, la figura 12 es el primer fotograma de la animación.

El mapa tiene un suelo sobre el que avanzan o descansan las Poké Balls, los Pokémon y el obstáculo, y su textura se muestra en la figura 13. Hay agua alrededor del suelo, cuya textura se muestra en la figura 14. Una pared gigante rodea la zona de juego, y su textura se muestra en la figura 15. Hay un obstáculo generado aleatoriamente al inicio del juego que tiene la textura mostrada en la figura 16. Estas cuatro texturas se asignan a cubos correspondiente transformados.

El video de demostración se puede ver en <https://youtu.be/1BTPIp3PK7A>.

Anexo

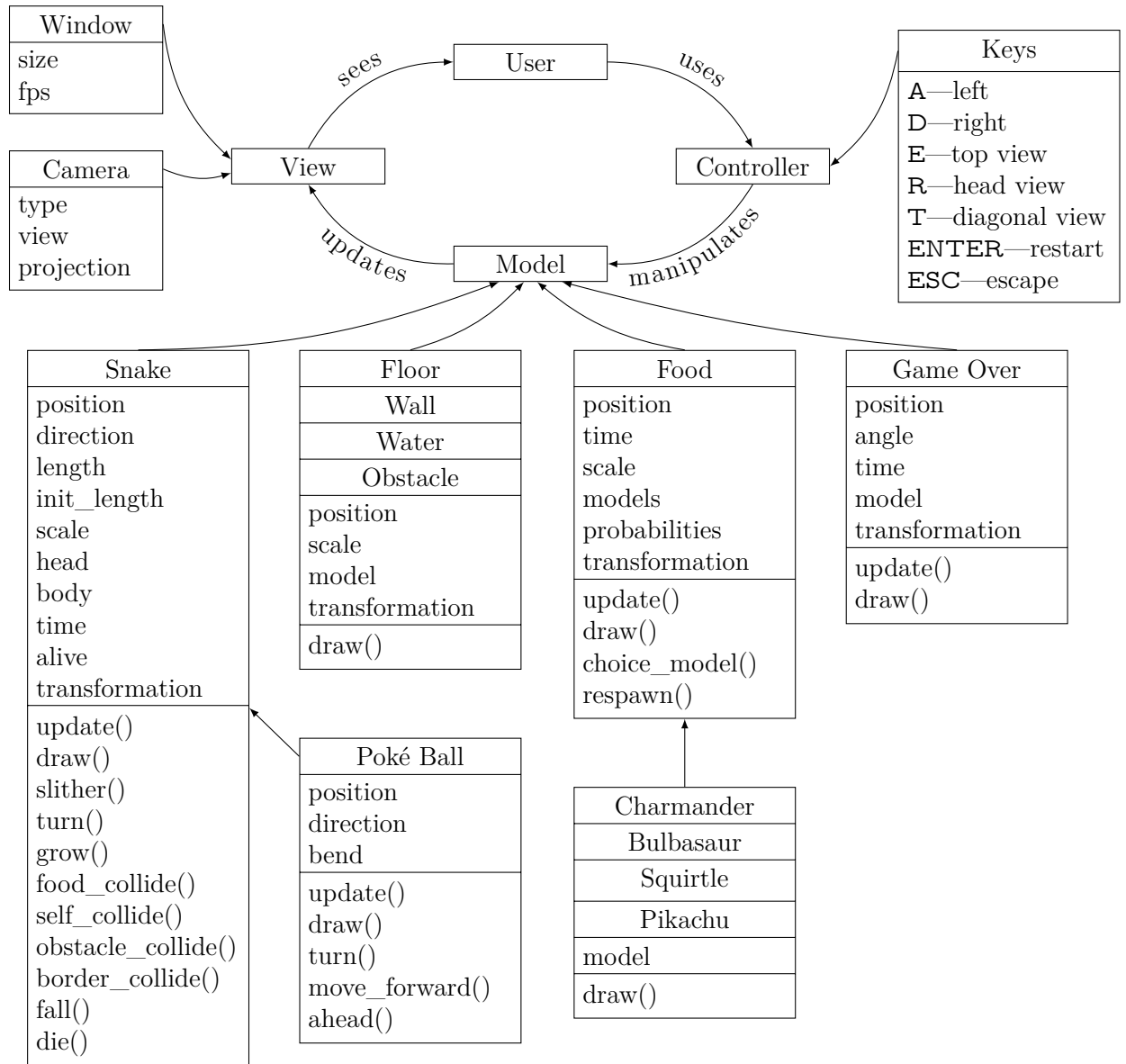


Figura 1: Patrón Modelo–Vista–Controlador

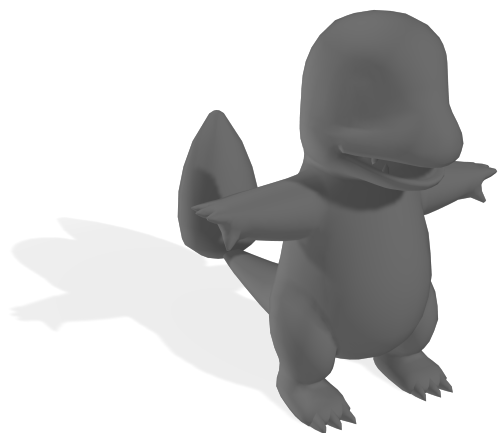


Figura 2: Modelo 3D de Charmander

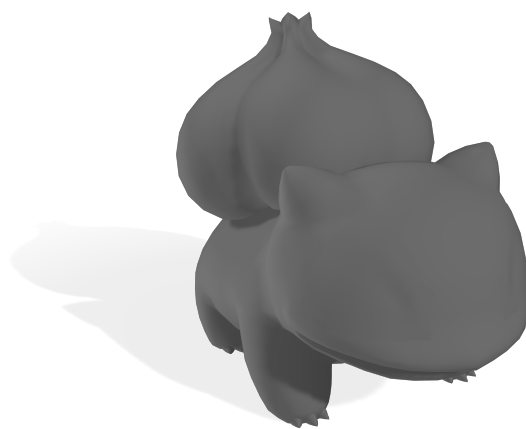


Figura 3: Modelo 3D de Bulbasaur

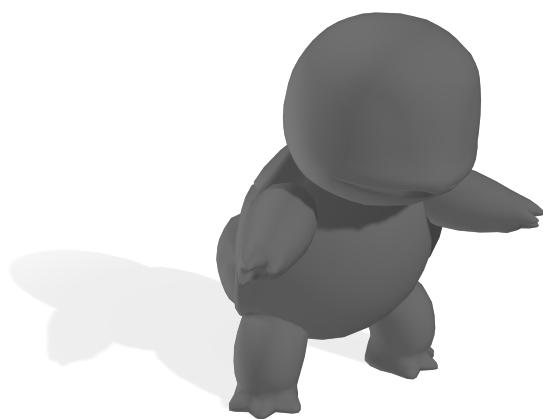


Figura 4: Modelo 3D de Squirtle

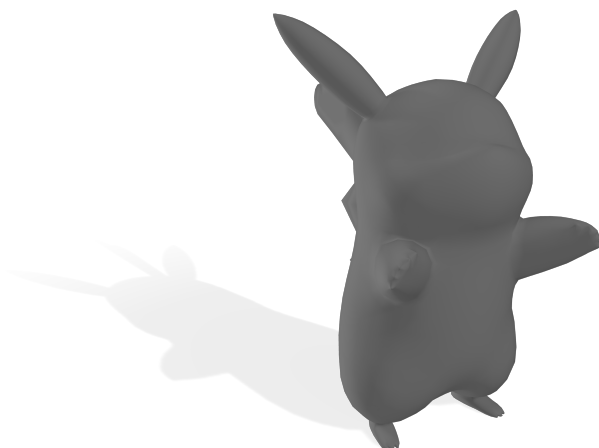


Figura 5: Modelo 3D de Pikachu

```
tarea2c/
├── ctr/
│   └── controller.py
├── lib/
│   ├── basic_shapes.py
│   ├── easy_shaders.py
│   ├── lighting_shaders.py
│   ├── object_handler.py
│   └── transformations.py
├── mod/
│   ├── tex/
│   │   ├── bulbasaur.obj
│   │   ├── charmander.obj
│   │   ├── game_over_1.py
│   │   ├── game_over_2.py
│   │   ├── game_over_3.py
│   │   ├── game_over_4.py
│   │   ├── ground.obj
│   │   ├── ground.png
│   │   ├── obstacle.obj
│   │   ├── obstacle.png
│   │   ├── pikachu.obj
│   │   ├── pokeball.obj
│   │   ├── pokeball.png
│   │   ├── squirtle.obj
│   │   ├── wall.obj
│   │   ├── wall.png
│   │   ├── water.obj
│   │   └── water.png
│   ├── floor.py
│   ├── food.py
│   ├── game_over.py
│   ├── models.py
│   ├── obstacle.py
│   ├── snake.py
│   ├── wall.py
│   └── water.py
├── view/
│   ├── camera.py
│   └── view.py
├── demo.mp4
├── main.py
├── report.pdf
└── requirements.txt
```

Figura 6: Estructura de la carpeta tarea2c

```
MINGW64:/c/Users/camil/OneDrive/Documentos/GitHub/CC3501-tareas/tarea2c
camil@DESKTOP-7KMKKNV MINGW64 ~/OneDrive/Documentos/GitHub/CC3501-tareas/tarea2c
(master)
$ python main.py
```

Figura 7: Ejecución del código en el intérprete de comandos

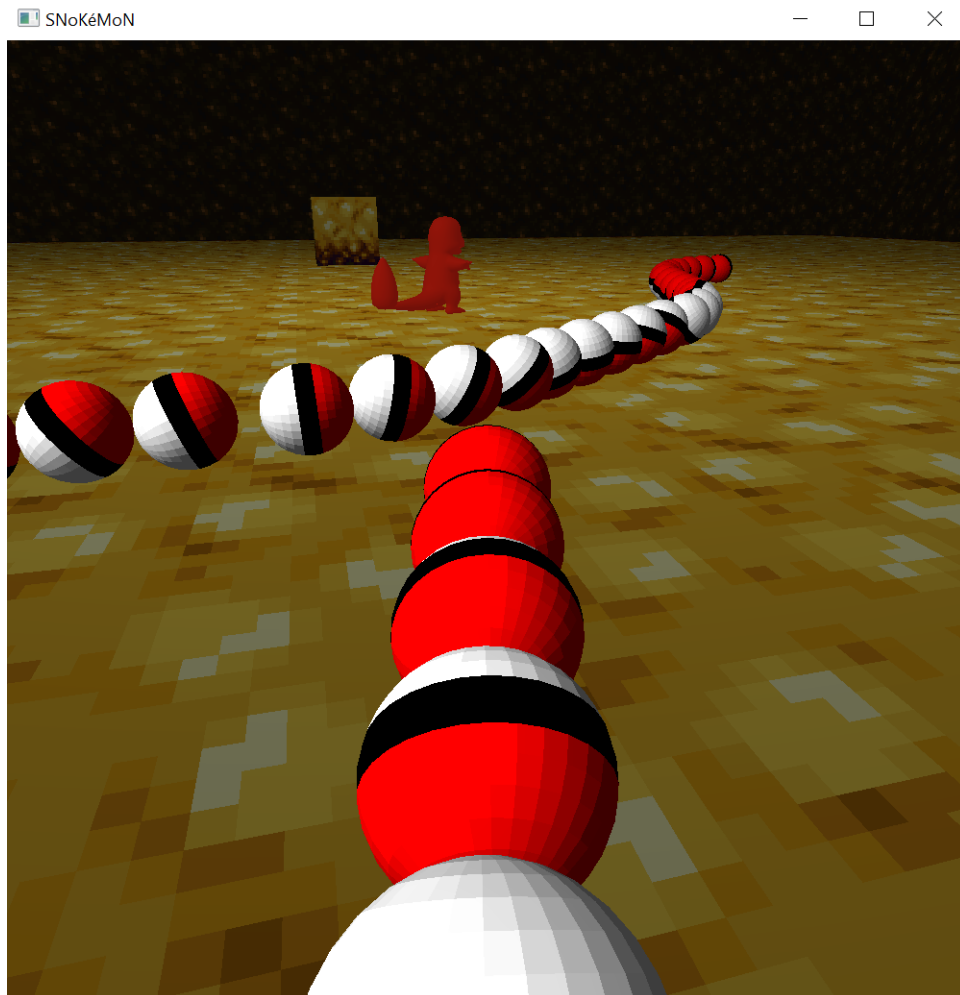


Figura 8: Larga cola de Poké Balls a punto de colisionar consigo misma, detrás está Charmander y el obstáculo. Vista en primera persona detrás de la cabeza

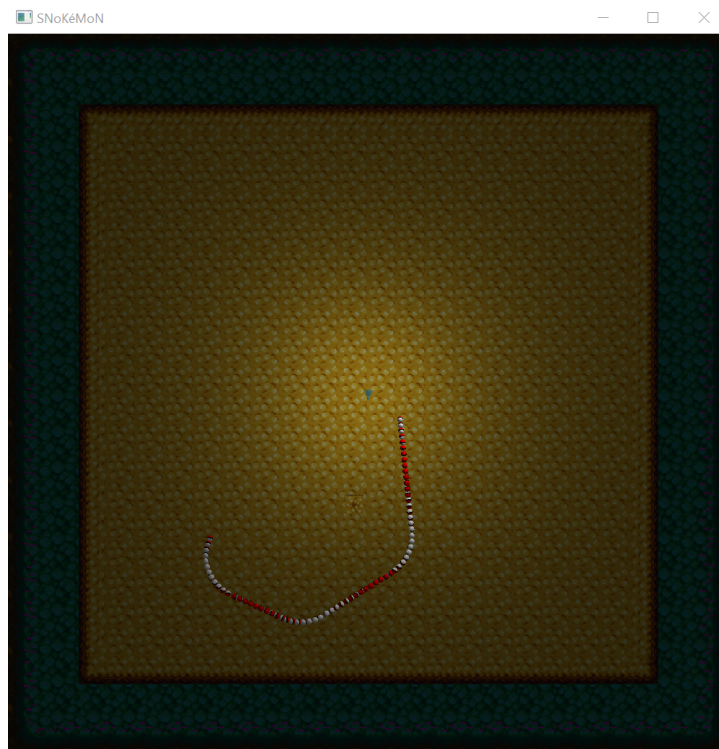


Figura 9: Vista superior estática

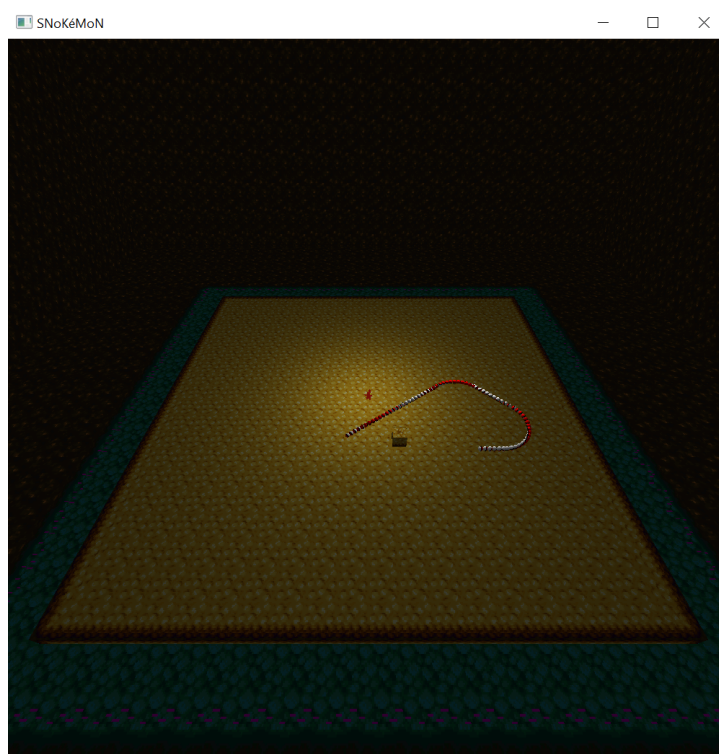


Figura 10: Vista diagonal estática

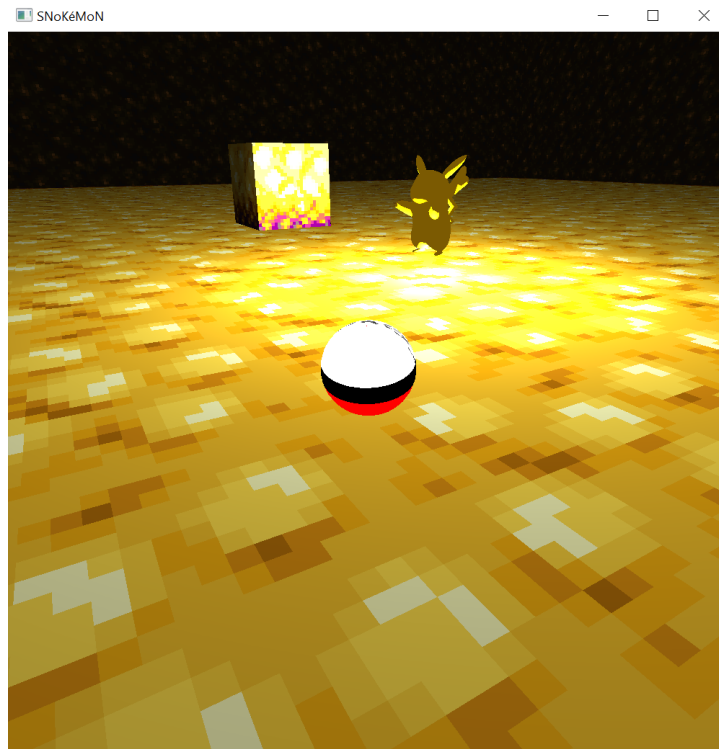


Figura 11: Poké Ball a punto de captura a Pikachu, durante el impactrueno, que ilumina la zona



Figura 12: Escena final. Fotograma 1/4 de la animación

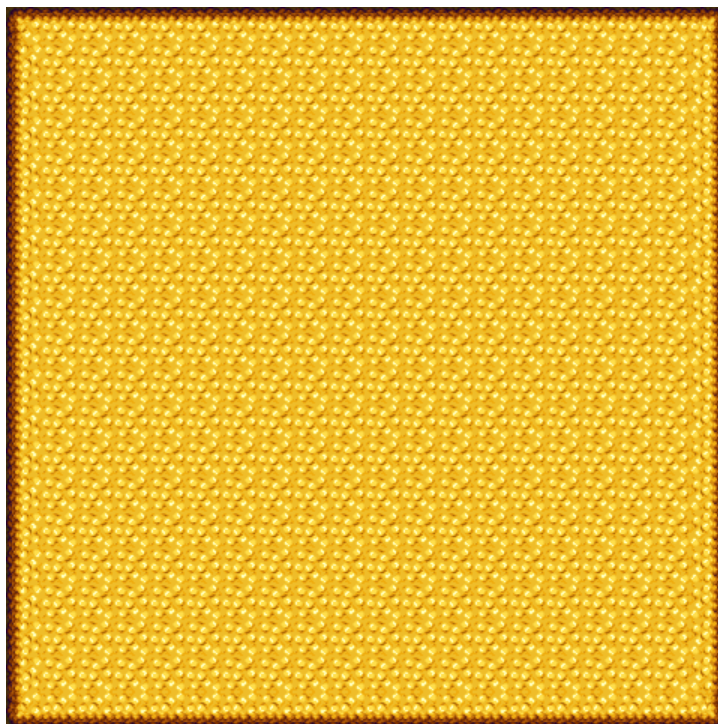


Figura 13: Textura del suelo del mapa

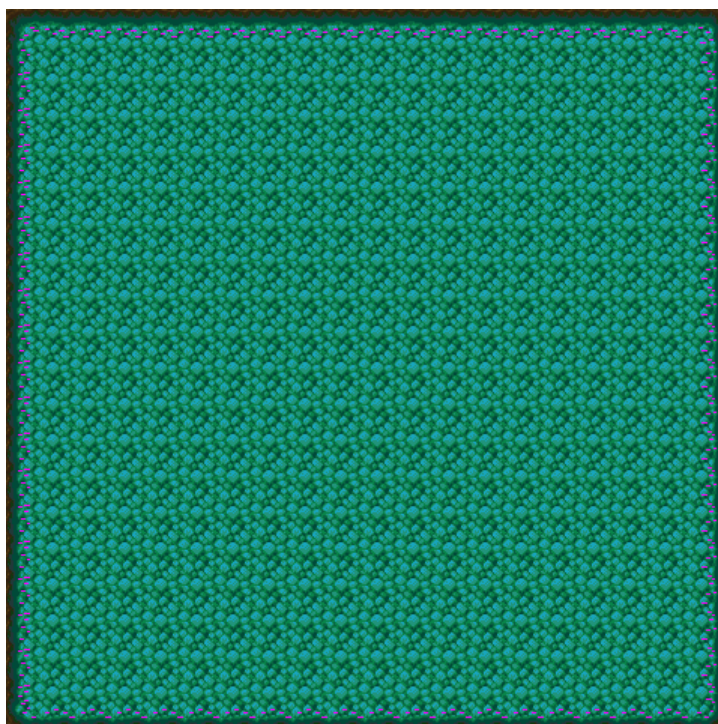


Figura 14: Textura del agua del mapa

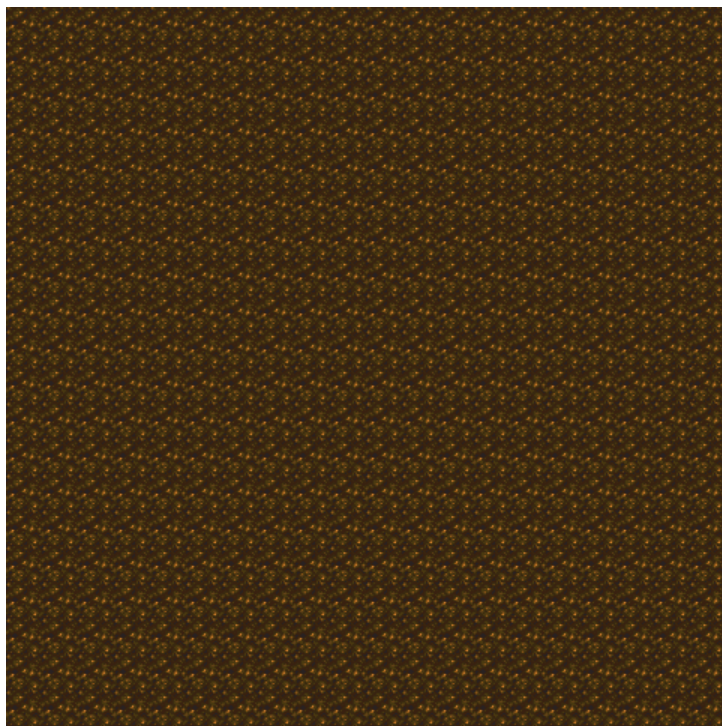


Figura 15: Textura del muro del mapa

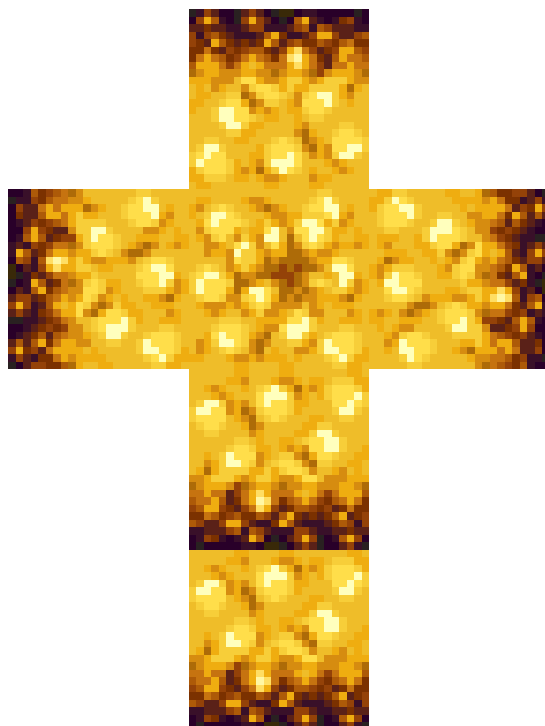


Figura 16: Textura del obstáculo del mapa

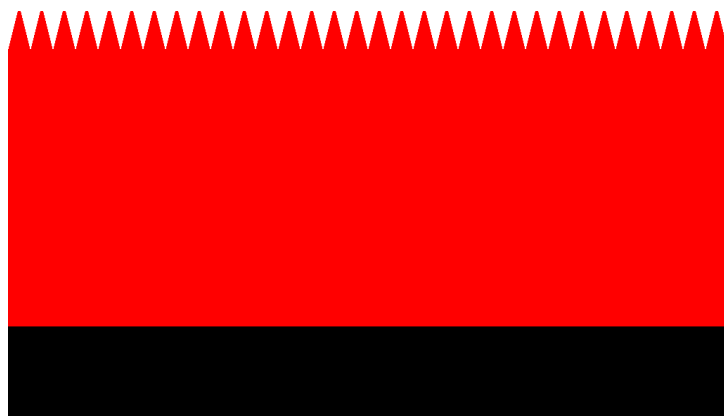


Figura 17: Textura de una Poké Ball