

N|N: GoldenStar

Ejercicio Final

Introducción

Neptune Inc., la compañía que produce los famosos chocolates N|N's, ha logrado finalmente la aprobación de un nuevo producto: GoldenStar, una estrella de chocolate recubierta de una muy delgada capa de oro. Como parte de la campaña de marketing asociada, nos han encargado la creación de un juego.

Los representantes de Neptune no fueron del todo claro con los requerimientos, más allá de indicar que debía ser “adictivo”, “fácil de usar” y, especialmente, “dejar claro el lugar central de GoldenStar en la línea de productos N|N”. En base a estos requerimientos, nuestro equipo creativo desarrolló un concepto de juego que será descrito a continuación.

Estructura general del juego

El juego se dividirá en niveles, cada uno de ellos con un número asignado, una tablero en especial y un puntaje requerido para pasarlo. Todos los jugadores empezarán en el nivel 1 y deberán ir pasando niveles para acceder a los posteriores.

Los niveles se juegan con al menos dos jugadores, aceptándose en un mismo nivel múltiples jugadores hasta un tope especificado al crear el nivel. El jugador podrá elegir contra que grupos de persona jugar o quedar en espera hasta que otros elijan jugar con él.

Mecánica de juego básica

Cada nivel del juego se desarrolla en un tablero compuesto por una cierta cantidad de celdas, cada una de ellas ocupada en un principio por uno de los siguientes productos N|N:

- **Button:** es un pequeño disco, viene en uno de cinco colores: rojo, amarillo, verde, azul y violeta.
- **MiniBar:** es una pequeña barra cilíndrica. También puede obtenerse en los cinco colores mencionados anteriormente.
- **GoldenStar:** es el producto central, para nuestros fines se considera sin color.

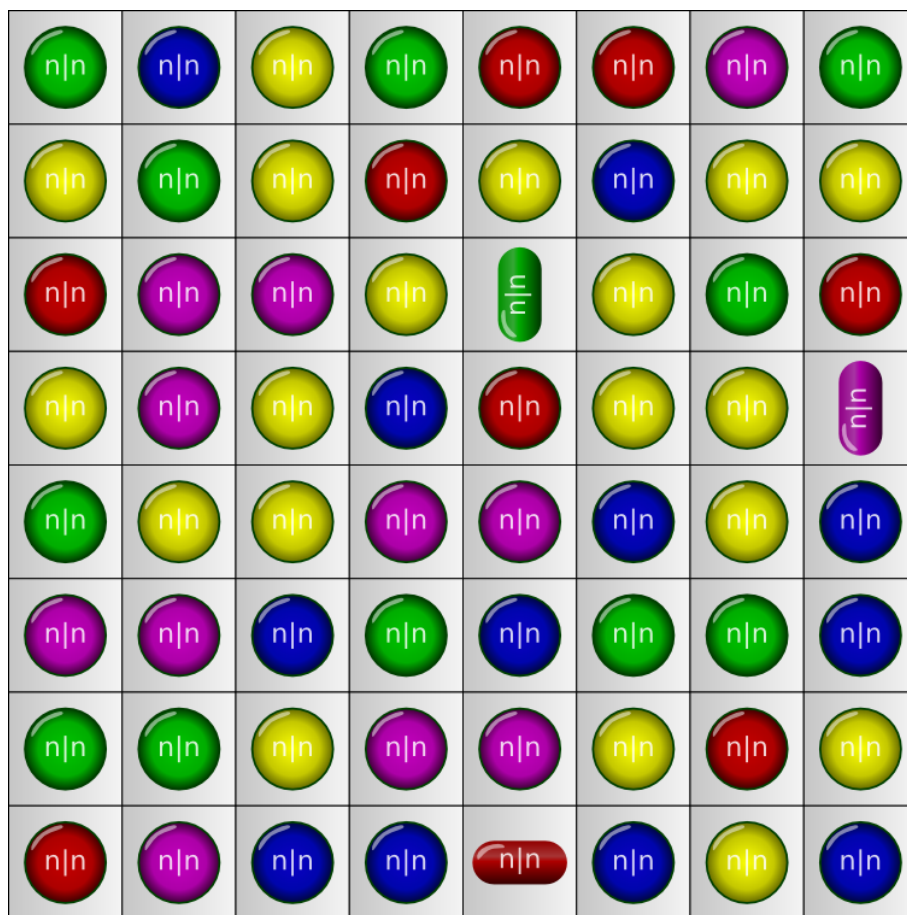


Figura 1 - Tablero básico de juego.

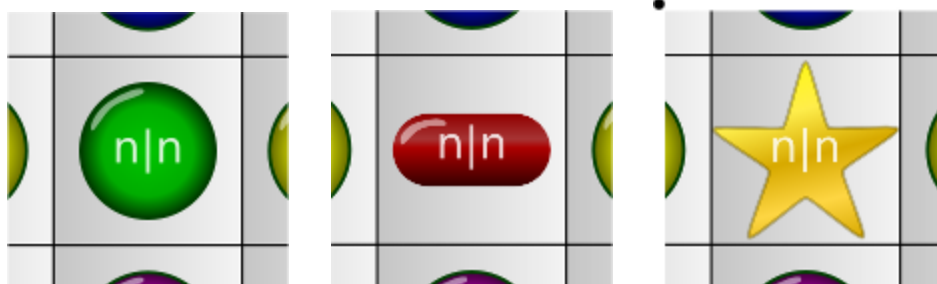


Figura 2 - Imágenes de productos: Button, MiniBar y GoldenStar, respectivamente.

El objetivo principal del juego es lograr ciertos patrones de colores y tipos de productos en celdas adyacentes. Dichas configuraciones se consiguen mediante el intercambio de los productos contenidos en celdas consecutivas. Los intercambios ocurren siempre de a pares de celdas donde el producto de la primera pasa a la segunda celda y viceversa. Las configuraciones o patrones están asociadas con ciertos puntajes y consecuencias:

- **Tres Buttons del mismo color en línea:** 10 puntos por cada uno. Los productos simplemente desaparecerán.
- **Cuatro Buttons del mismo color en línea:** 20 puntos por cada uno. El producto cuyo movimiento

haya producido la configuración será transformado en una MiniBar, orientada en el mismo sentido que los cuatro Buttons. Los otros Buttons desaparecerán.

- **Cinco Buttons del mismo color en línea:** 30 puntos por cada uno. El producto cuyo movimiento haya producido la configuración será transformado en una GoldenStar, mientras que los otros Buttons desaparecerán.
- **Más de tres productos del mismo color en línea, no todos Buttons:** actuará del mismo modo que si se tratara de Buttons, pero activando las MiniBars incluidas. Cada MiniBar destruirá toda la columna o fila en la que se encuentre, dependiendo de su orientación.

La GoldenStar sólo se activará mediante una acción directa del jugador, intentando intercambiarlo con un producto adyacente. El resultado de esta acción dependerá de cual sea este producto:

- **Un Button:** eliminación de todos los productos del mismo color.
- **Una Minibar:** eliminación de todos los productos del mismo color, además de la fila y columna que contengan a la minibar.
- **Una GoldenStar:** eliminación de todos los productos contenidos en el tablero.

En todos estos casos se darán 40 puntos por cada uno de los productos eliminados.

La eliminación de productos dejará celdas vacías en el tablero, que serán llenadas mediante la caída de los productos presentes en las celdas superiores. Las celdas situadas “arriba del borde superior” se considerarán rellenas con productos aleatorios (sin considerar si estas combinaciones son estables o no). Queda a criterio de los desarrolladores como manejar la caída “alrededor” de obstáculos (celdas vacías).

Cuando el proceso de caída termine, la configuración se volverá a evaluar en busca de configuraciones especiales. En caso de producirse una nueva configuración especial, el puntaje que corresponda será multiplicado por dos. El proceso se repite hasta que se llegue a una configuración estable y, mientras este proceso se repita, los jugadores no podrán realizar acciones.

El juego terminará cuando no haya más acciones posibles (todos pierden) o cuando uno de los jugadores llegue al puntaje especificado (ese jugador pasará de nivel y los otros tendrán que reintentarlo).

Aplicaciones requeridas

El juego se dividirá en tres aplicaciones: cliente, servidor y editor de niveles. A continuación se describirán estas tres aplicaciones.

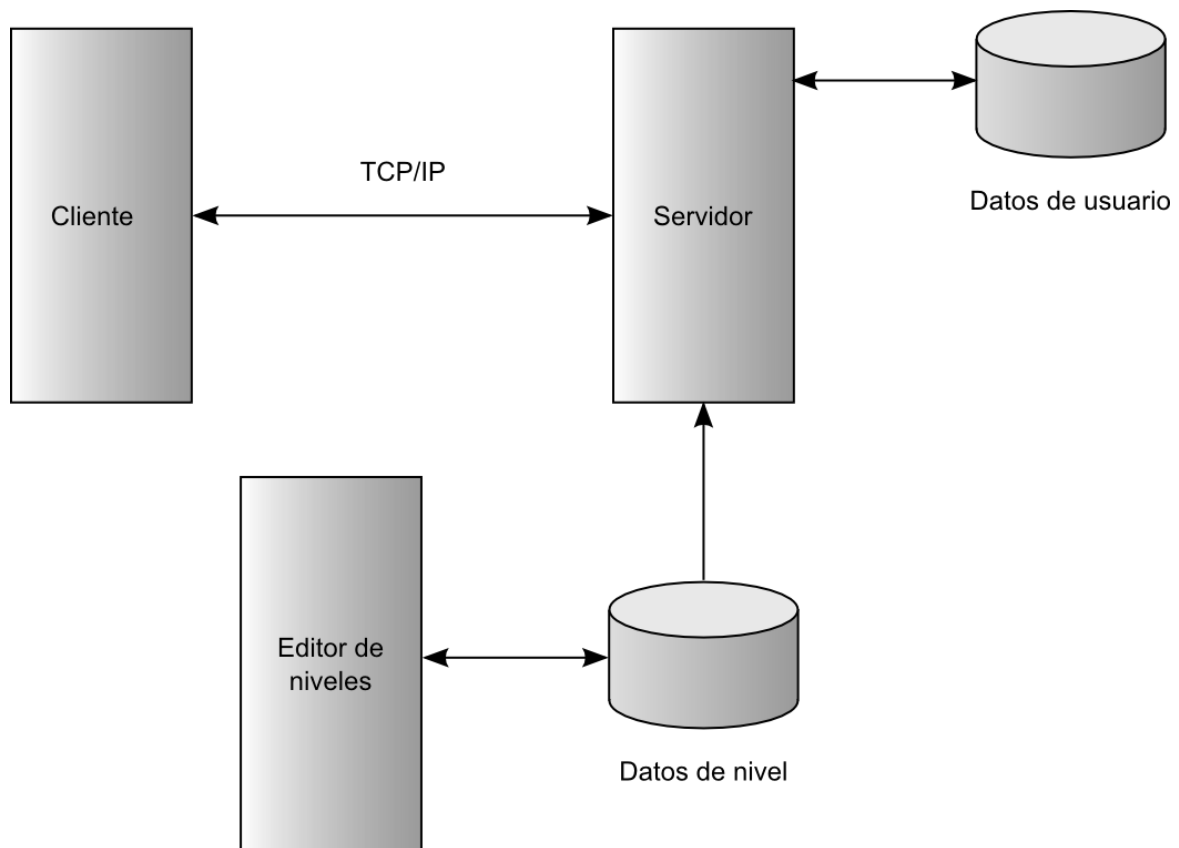


Figura 3 - Diagrama de interacción entre las aplicaciones.

Cliente

Será la aplicación gráfica utilizada para jugar. Deberá permitir conectarse al servidor (creando una cuenta de usuario en caso que esta no exista), seleccionar un nivel a jugar y manejar el desarrollo del juego.

Como el usuario final interactuará con esta aplicación, es de gran importancia que presente una interfaz lo más atractiva posible.

Servidor

Será una aplicación de línea de comandos que coordinará la operación de los clientes, asegurándose de que compartan un mismo estado del mundo. También será responsable de la distribución de los archivos de nivel, manejo de puntajes y autenticación de los usuarios.

Editor de niveles

Será una aplicación gráfica que permitirá la creación de niveles en forma lo más intuitiva posible. Debe soportar configurar por lo menos los siguientes parámetros:

- Nombre de nivel.
- Forma del nivel (celdas, agujeros).
- Imágenes de fondo e imágenes especiales por celda.
- Puntaje requerido para pasarlo.
- Probabilidades de aparición de distintas clases de productos y de distintos colores, tanto por celda (inicio del nivel) como por columna (para cuando desaparecen productos).
- Máximo número de jugadores simultáneos.

Restricciones

La siguiente es una lista de restricciones exigidas por el cliente:

- El sistema debe realizarse en ISO C++ [1].
- Pueden utilizarse las bibliotecas GTK+, gtkmm y SDL [2] (consultar sobre el uso de extensiones).
- Deben usarse sockets bloqueantes para las comunicaciones.
- Deben implementarse efectos gráficos y sonoros para la desaparición de los productos y la caída de los mismos.
- Con el objetivo de facilitar el desarrollo de las interfaces de usuario, se permite el uso de Glade.
- Los archivos de niveles deben ser almacenados en formato JSON [3], el cual debe ser parseado usando una biblioteca de terceros tal como JsonCpp [4].
- El protocolo de comunicación cliente-servidor también debe estar en formato JSON.
- Los archivos de niveles solo deben encontrarse en el servidor.
- El servidor debe confiar lo menos posible en los clientes.
- Las comunicaciones de los clientes deben estar “firmadas” usando HMAC [5], usando la clave del jugador como clave privada.
- Es condición necesaria para la aprobación del trabajo práctico la entrega de la documentación mínima exigida (consultar sitio de la cátedra). Es importante recordar que cualquier elemento faltante o de dudosa calidad pone en riesgo la aprobación del ejercicio.

Referencias

1. C++ Standards Committee. “ISO/IEC 14882: Programming Language C++ (last public draft)” - <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2013/n3690.pdf>
2. libsdl.org. “Simple DirectMedia Layer” - <http://www.libsdl.org/>
3. json.org. “Introducing JSON” - <http://json.org/>
4. Baptiste Lepilleur. “JsonCpp Documentation” - <http://jsoncpp.sourceforge.net/>

5. Wikipedia. "Hash-based message authentication code" -
http://en.wikipedia.org/wiki/Hash-based_message_authentication_code