

Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Estado de México

**Ciclo 3. Evidencia final**

Materia:

**Construcción de software y toma de decisiones (Gpo 401)**

Alumnos:

**Héctor Gónzalez Sánchez - A01753863**

**Luis Adrián Abarca Gómez - A01798043**

**Gerardo Rios Mejía- A1753830**

**Juan Carlos Carro Cruz - A01748640**

**Alfredo Azamar López - A01798100**

Profesores:

**Aram Baruch González Pérez**

**Humberto Cárdenas Anaya**

**Ariel Ortiz Ramírez**

**Roberto Martínez Román**

Campus Estado de México Mayo 2023

**Índice**

[**Contexto del problema 2**](#_b1shw49dvt4x)

[**Requerimientos funcionales 2**](#_1fob9te)

[**Diagrama de Caso de Uso 3**](#_3znysh7)

[Casos de uso extendidos 4](#_2et92p0)

[**Requerimientos no funcionales 7**](#_tyjcwt)

[**Definición de perfiles de usuario 8**](#_3dy6vkm)

[Jugadores 8](#_1t3h5sf)

[Administrador 9](#_4d34og8)

[**Requisitos del despliegue del sistema 10**](#_vbyf5lgdev2v)

[Requisitos de Hardware 10](#_oek8djenkep3)

[Requisitos de Software 10](#_r8624v2ixu9i)

[Requisitos de Red 11](#_5a9mvx8g2wni)

[Requisitos de seguridad 11](#_cish66kiys7k)

[Requisitos de configuración 11](#_9uh92v73496p)

[Requisitos de documentación 11](#_mftec0fiqnpl)

[**MockUp 12**](#_6cp9md5awgkp)

[**Modelo relacional de la base de datos 17**](#_26in1rg)

[**Diagramas de clases 21**](#_lnxbz9)

[**Diagrama de despliegue 22**](#_aho970oyd78m)

[**Diagrama de componentes 23**](#_35nkun2)

[**Avance del videojuego 24**](#_1ksv4uv)

[Historia del videojuego 24](#_2sx17pjb569j)

[Reto a resolver 25](#_8n5xk1spfpzv)

[Desarrollo de personajes 25](#_grj54u2gnizq)

[Prototipo 27](#_aq9b0itdtg5)

[**Prototipo de la plataforma tecnológica 29**](#_t9qkqoniigg2)

[**​​Definición de las pruebas unitarias 31**](#_t76jkqpo6ko6)

[**Licencia de software 34**](#_wlfe93nogf0i)

[**Derechos de autor 36**](#_dbvzp12nwjc8)

# Contexto del problema

El nombre del proyecto será **“Proyecto Titor”**.

Uno de cada diez colegios enseñan programación intracurricular limitando el aprendizaje de prueba y error junto con el desarrollo de habilidades primordiales como la resolución de problemas y la mejora de habilidades cognitivas. Además, el mundo se enfrenta a un gran desafío ya que en la actualidad, existe una necesidad de 1.2 millones de programadores en Latinoamérica en 2025 y se tiene un déficit del 38%.

*Crack The Code* es una empresa dedicada a enseñar a los niños a programar, la cual nace a partir de esta problemática. Ellos nos han pedido realizar un videojuego que sea capaz de enseñar a los niños a programar y que a la vez sea divertido, además de llevar un registro de las puntuaciones y de las cuentas de las personas que ingresan a jugar.

El juego tiene como objetivo ser atractivo para enseñar a los niños y a los adolescentes que la programación es divertida y tiene un beneficio en sus vidas, como el desarrollo de su pensamiento lógico y analítico. También deberá correr en el navegador y evitar descargas de archivos en la computadora.

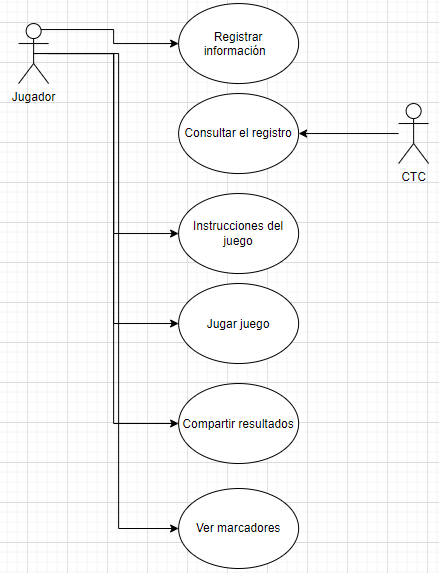
### 

# Requerimientos funcionales

* Se requiere almacenar los datos de contacto de los jugadores y de los padres/tutores.
* Crear un videojuego que sea visualmente atractivo (colores de fondo, diseño de personajes y niveles, y música de fondo) para un rango de edades de 12 a 18.
* Diseñar y crear una base de datos relacional tiene que estar normalizada.
* Permitir *queries* en el formato de SQL.
* Debe ser capaz de recuperar datos de la base de datos para que los jugadores puedan ver sus puntuaciones y la de los demás jugadores.
* Debe ser capaz de correr el videojuego que se creara sin problemas.
* Debe contener elementos como una historia, que deje algún aprendizaje, estética, reglas y un objetivo alcanzable.
* Interoperabilidad: todos los diferentes sistemas incluidos en el producto final deben ser capaces de intercambiar información entre ellos y utilizarla con diversos propósitos, siempre apegado a la integridad de estos mismos.
* Debe de mantener la puntuación más alta de los distintos usuarios en la base de datos .

### 

# Diagrama de Caso de Uso



**Imagen 1**. Se representa el diagrama de caso de uso del producto

## Casos de uso extendidos

Caso de uso extendido de “Registrar información”:

**Actores:** Jugador.

**Objetivo:** Registrar la información de contacto del jugador y de sus padres/tutores.

**Resumen:** Mario es un niño de 12 años que no sabe qué estudiar pero tiene inclinaciones por la programación. Al buscar información por internet encuentra la página de "Crack The Code" y observa qué le puede brindar más información sobre la programación.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El jugador busca la página web de Crack The Code. |  |
| **2.** El jugador entra a la sección de “Juego”. | **3.** Despliega campos para ingresar datos de contacto. |
| **4.** El jugador llena los campos con sus datos. | **5.** El sistema despliega un botón para iniciar. |

Pasos alternativos:

* En el paso 2 el jugador puede pedir ayuda a sus padres/tutores para el llenado de la información.

Caso de uso extendido de “Consultar el registro”:

**Actores:** Crack The Code.

**Objetivo:** Poder visualizar todos los jugadores que se han registrado.

**Resumen:** Un product manager de la empresa Crack The Code puede solicitar un reporte para ver los registros de los jugadores que se registraron.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El product manager entra a la sección de “Administración”. | **2.** Despliega un menú para obtener las estadísticas del uso del juego. |
| **3.** El product manager oprime el botón de “Users per day” | **4.** Presenta la información. |
| **5.** El product manager oprime el botón de “Active Users” | **6.** Presenta la información. |

Caso de uso extendido de “Instrucciones del juego”:

**Actores:** Jugador.

**Objetivo:** Dar a conocer las reglas del juego.

**Resumen:** Una vez que el jugador haya ingresado sus datos de contacto podrá apretar un botón de inicio para empezar el juego.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El jugador oprime el botón de “inicio”. | **2.** Inicia la introducción del juego. |
|  | **3.** Se imprimen las instrucciones (reglas) del juego. |

Caso de uso extendido de “Jugar juego”:

**Actores:** Jugador.

**Objetivo:** Empezar a jugar el juego.

**Resumen:** Después de que el jugador haya leído las instrucciones podrá jugar el juego.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El jugador escoge un personaje. | **2.** Empieza el nivel 1. |
| **3.** El jugador juega los diferentes niveles hasta llegar al “Boss Level”. |  |
| **4.** El jugador termina el juego. | **5.** Se despliega la tabla de puntuación. |

Pasos alternativos:

* En el paso 3, el jugador puede no completar algún nivel. En este caso se despliega la tabla de puntuación.

Caso de uso extendido de “Compartir Resultados”:

**Actores:** Jugador.

**Objetivo:** Visualizar los resultados del juego.

**Resumen:** Cuando el jugador haya terminado el “level boss” o haya perdido en un nivel, podrá ver su puntaje final.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El jugador termina el juego. | **2.** Despliega el desempeño del jugador. |
| **3.** El jugador pública / comparte su puntaje con sus amigos. |  |

Pasos alternativos:

* En el tercer paso el jugador puede no compartir sus resultados.

Caso de uso extendido de “Ver marcadores”:

**Actores:** Jugador.

**Objetivo:** Visualizar las mejores puntuaciones de otros usuarios.

**Resumen:** Cuando el jugador haya terminado el “level boss”, haya perdido en un nivel, o quiera consultar los demás puntajes, podrá ver su puntaje final y el de los jugadores con mejores puntuaciones.

| **Usuario** | **Sistema** |
| --- | --- |
| **1.** El jugador desea ver las mejores puntuaciones. | **2.** Muestra las mejores puntuaciones registradas hasta el momento |
| **3.** El jugador se motiva para superar dichas puntuaciones  4. El jugador vuelve a jugar hasta superar su objetivo |  |

Pasos alternativos:

* Ocurre un error y el jugador no puede ver las puntuaciones de los demás.

# Requerimientos no funcionales

* Se requiere un producto final confiable, divertido, disponible y de calidad que funcione correctamente; previniendo así errores futuros y un mantenimiento costoso.
* La página web debe ser intuitiva de manera que, sin contar con algún manual de uso o tutorial, tanto los niños como los padres sean capaces de identificar rápidamente todos los componentes del sitio y no se sientan abrumados, provocando así, su salida del sitio web.
* Integridad de datos: el sitio web debe asegurar que los datos recabados serán de uso exclusivo de Crack The Code.
* Disponibilidad: aunque el sitio web y todo lo que incluye no estará disponible siempre debido a las limitaciones que se tienen al usar AWS, hay que garantizar que el sitio web no presente ningún tipo de error en los intervalos de tiempo en los que se encuentre disponible.
* Seguridad: el sitio web debe garantizar una conexión segura a todos los usuarios, sin riesgo a un ataque o a un virus informático.
* Eficiencia: la base de datos debe contar con un buen diseño para facilitar el envío de reportes a Crack The Code sobre los datos recaudados.
* Documentación: si bien muchos niños podrían tener cierto interés por los videojuegos y probablemente tengan una pequeña noción de cómo jugar, hay muchos que no cuentan con ello, por lo que incluir instrucciones de cómo jugar el juego es imprescindible.
* Portabilidad: el juego deberá correr de manera eficiente por lo menos en los navegadores más populares.
* El videojuego debe motivar a los jugadores a seguir jugando, aumentando así el tráfico web de Crack The Code y que los niños compren algún curso de los que la empresa ofrece.

# Definición de perfiles de usuario

## Jugadores

*Niño:*

Como muchos otros niños, Alberto vió una escena de una película en donde sale un programador y se le hizo algo interesante. Alberto buscó en internet acerca del tema y encontró una página que enseña a niños de su edad a programar. Después de registrar sus datos, Alberto jugó el videojuego, se interesó aún más por la programación y decidió comprar un curso de Crack the Code.

*Niña:*

El papá de Paulina trabaja en una consultoría de SAP, ella ha visto el trabajo de su padre y le ha surgido un interés por saber programar, pero no tiene idea de donde empezar. Su padre no cuenta con el tiempo suficiente para enseñarle cosas básicas de la programación. Un día la madre de Paulina vio en redes sociales un anuncio de Crack The Code informando sobre los cursos que su hija puede tomar para acercarse a los conocimientos de programación.

*Padre de familia:*

José es padre de familia único y está bastante preocupado por el futuro de su hijo ya que no pasa mucho tiempo con él y le resulta muy complicado llevarlo a una escuela de forma presencial. Un día su hijo le cuenta a José acerca de un videojuego que jugó en internet acerca de la programación y le dijo que se interesó bastante por ello y que le gustaría comprar un curso en línea de Crack the Code para aprender a programar. José se da el tiempo de checar la página y le compra a su hijo un curso.

## Administrador

*Encargado de Marketing:*

Lisa, junto con otras personas, está a cargo del área de marketing de Crack the Code. Una de las principales funciones de Lisa es llevar un registro de la cantidad de clientes nuevos que reciben en ciertos intervalos de tiempo y de los cursos que más inscriben. Lisa debe tener acceso a la base de datos para poder hacer estos reportes y sobre todo, para evaluar las posibilidades de abrir nuevos cursos o cerrar los menos populares.

*Desarrollador Full-Stack:*

Mario es uno de los tantos desarrolladores full-stack de Crack the Code. Mario, a diferencia del resto, no da clases a los niños, ya que su función principal es darle mantenimiento a la página web. Mario debe de tener acceso al código fuente de toda la página para actualizarla constantemente y checar que todo funcione como debería.

*CEO:*

María Vélez, como fundadora de Crack the Code, debe estar al tanto de todo lo que ocurre en su empresa. Al ser este un proyecto bastante importante, María debe tener acceso a absolutamente todo el sistema, a pesar de que es probable que María jamás modifique nada de esto.

*Ingeniero de base de datos:*

Alan es uno de los tantos ingenieros de base de datos de Crack the Code. Mario es el encargado de dar mantenimiento a la base de datos y checar que todo esté funcionando correctamente para que Lisa, la encargada de marketing, pueda cumplir con su labor.

# Requisitos del despliegue del sistema

## Requisitos de Hardware

En cuanto a los requisitos de hardware utilizamos un servicio en la nube para alojar nuestro sistema, el utilizado es el T3.small de AWS.

Una instancia T3.small es una instancia de uso general ampliable de última generación, la cual puede ampliar el uso de la CPU, diseñadas para un uso moderado y de los cuales puede haber un pico en temporal en su uso. Tienen créditos de CPU que se acumulan y que se gastan cada vez que se supera el límite de uso de CPU.

* Procesador Intel Xeon Scalable hasta 3.1 GHz (Skylake 8175M o Cascade Lake 8259CL)
* CPU ampliable con los créditos
* AWS Nitro System
* Rendimiento de red de 5Gbps

Un dispositivo con conexión a internet por el cual se pueda ingresar al sistema.

## Requisitos de Software

AWS: Amazon Web Services es un servicio de computación en la nube y que sirve para tener el sistema alojado en una nube y poder acceder a ella a través de internet.

cloud9: es un IDE en línea que permite manipular el código de varios lenguajes y que además permite ejecutarlo y depurarlo. Admite varios lenguajes de programación, incluidos C, C + +, PHP, Ruby, Perl, Python, JavaScript con Node.js y Go.

Unity: es un motor de videojuegos que permite modificar los archivos del mismo y hacer pruebas de funcionamiento así como exportar el juego para alguna plataforma.

Microsoft SQL server: en SQL Server permite la modificación de la estructura de la base de datos.

HTML: HyperText Markup Language utilizado a través de cloud9 permite hacer páginas web además de poder editar la estructura de cómo se desplegarán los elementos de la misma.

CSS: Cascading Style Sheets es utilizado para darle estilo a la página como en texto e imágenes.

JavaScript: un lenguaje de programación utilizado para las operaciones en la página web.

Node js: utilizado para la comunicación del sistema y manejar los paquetes de las mismas.

Navegadores compatibles con cloud9

## Requisitos de Red

En cuanto a los requisitos de red se requiere una conexión a internet estable y confiable para acceder al entorno, ademas de tambien ser lo suficientemente rápida para subir y bajar archivos

## Requisitos de seguridad

La página se asegura de encriptar los datos sensibles de quienes ingresan sus datos además de tener un apartado donde solo se da acceso a administradores.También al estar en AWS hay opciones para mantener la seguridad de las mismas.

## Requisitos de configuración

En cuanto a los requisitos de configuración, si se llegara a hacer una modificación de los datos de los usuarios de la información almacenada en la base de datos se puede configurar. Otras se podrían realizar mediante la modificación de los programas que se ejecutan.

## Requisitos de documentación

En cuanto a los requisitos de documentación se encuentra en el mismo código el cual está comentado para saber cual es su utilidad y en cuando el uso de las demás partes se puede hallar el manual en los siguientes archivos.

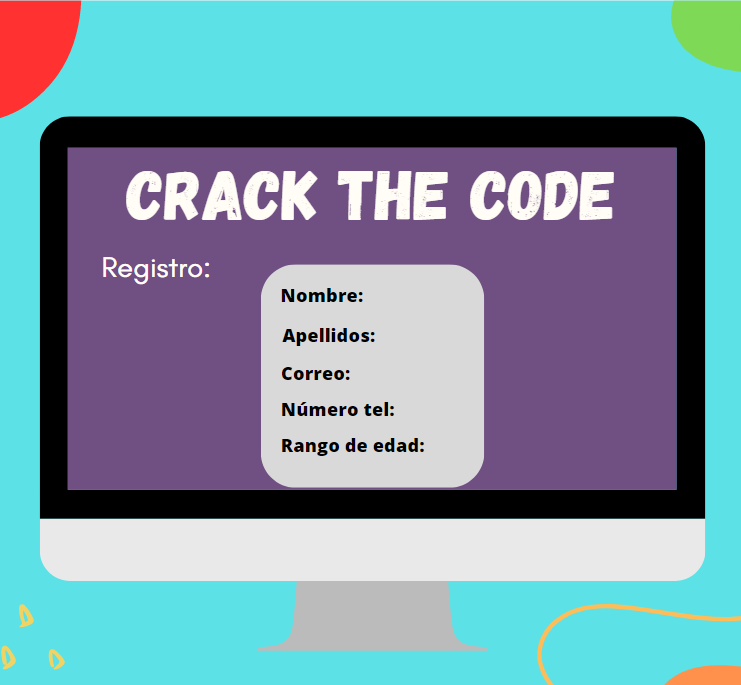
[ManualDeUsuario\_Jugador.pdf](https://drive.google.com/file/d/1Uzody1G4aoGdJnRIYCvtOzvm_FK91GQo/view?usp=share_link)

[ManualDeUsuario\_Administrador.pdf](https://drive.google.com/file/d/18HHBt0TIz1QaM4wo59jUizHjZqreeVk-/view?usp=share_link)

Así como los datos que se muestran ya en el juego:

[ValoresDelJuego\_E4.pdf](https://drive.google.com/file/d/1RG8WI18j4xt9Qrl0IQfJMGrC683dGAss/view?usp=share_link)

# MockUp



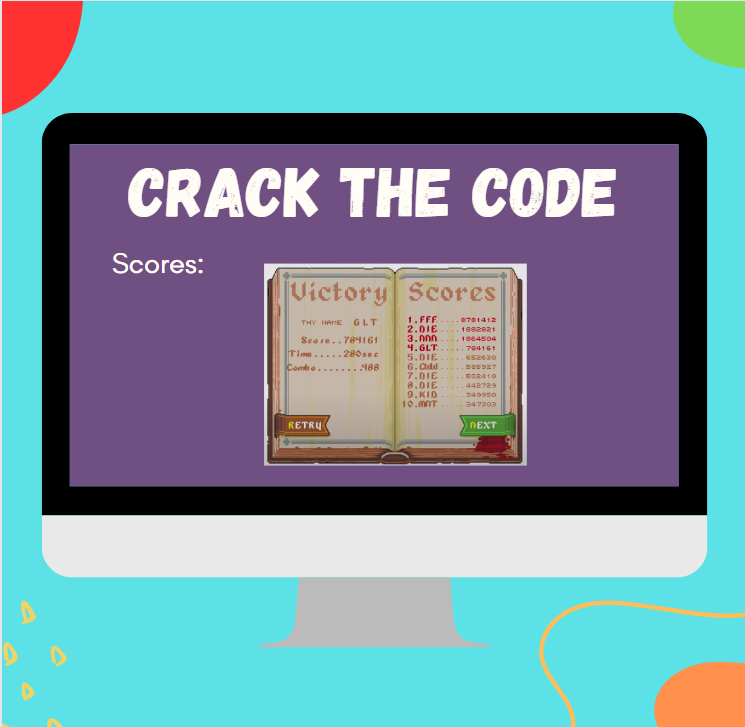
1. El usuario deberá introducir sus datos tales como el nombre, apellidos, correo electrónico y teléfono además de su rango de edad, posteriormente dará “Enter” y será dirigido a la pantalla de inicio.



1. Una vez en la pantalla de inicio se le será mostrado al usuario sus datos y las opciones tales como la entrada o acceso al juego, su tabla de puntuaciones en caso de ya haber jugado anteriormente y finalmente un apartado en el que se le redirija a Whatsapp en caso de contar con dudas.



1. Si el usuario selecciona la opción de “Juego” será redirigido a una pestaña en la cual podrá acceder al juego y hacer uso de este mismo siempre que el usuario quiera.



1. Finalmente si el usuario selecciona la opción de “Scores” se le desplegará la tabla de puntuación más alta que ha tenido hasta el momento, además de poder visualizar las puntuaciones de otros usuarios y ver el historial de estas mismas.



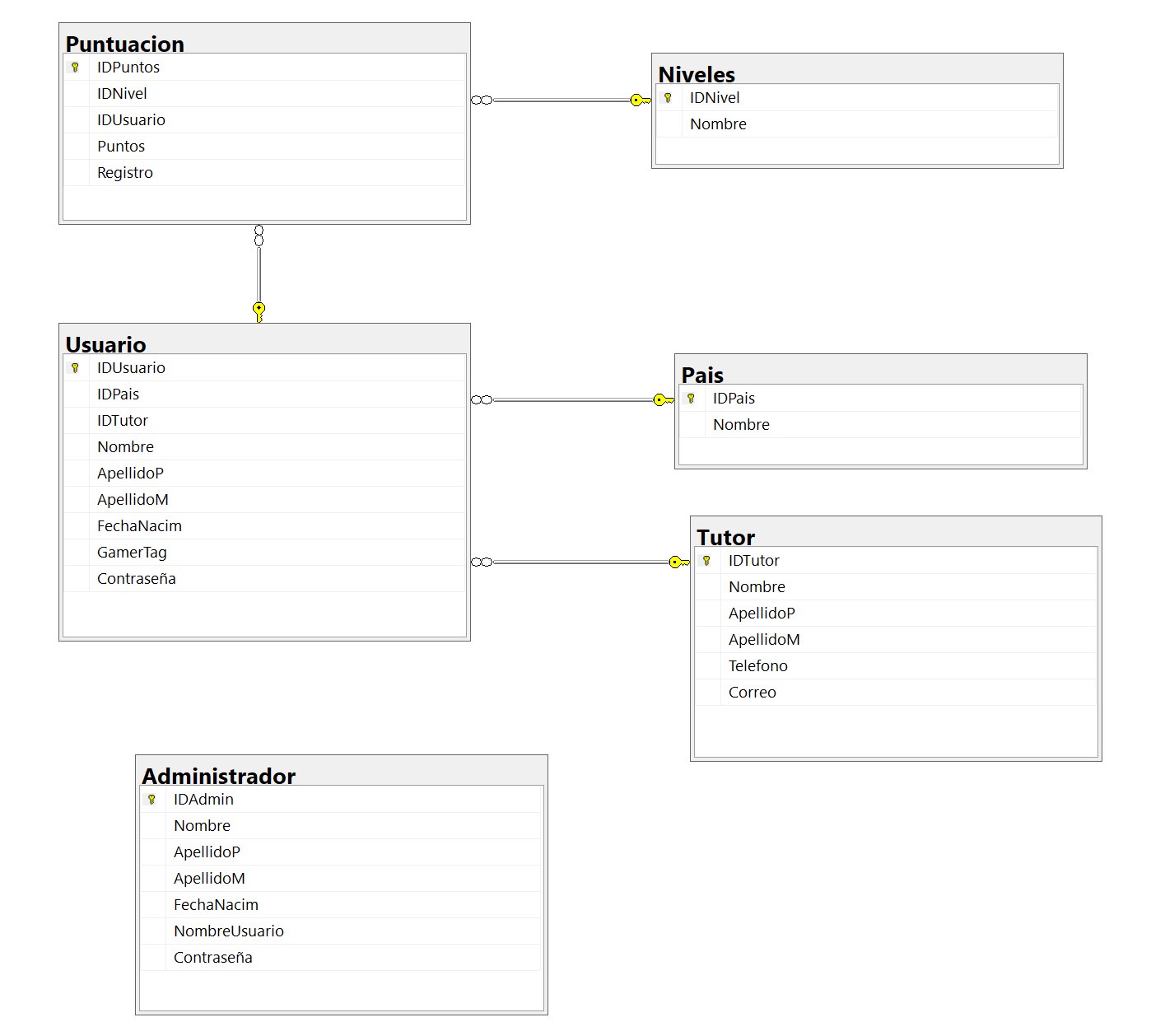
1. Por parte de los trabajadores de Crack The Code, contarán con una acceso especial para poder visualizar un listado de usuarios que se han suscrito, este mediante el botón de “administrador”, además de una gráfica que muestre los índices mensuales de suscriptores y el plan que seleccionaron.

### 

# Modelo relacional de la base de datos

La base de datos está conformada por 3 tablas fuertes y 2 débiles; las tablas fuertes son “Niveles” con su *Primary Key* “IDNivel”, “Pais” con su *Primary Key* “IDPais” y Tutor con su *Primary Key* “IDTutor”. Las tablas débiles o de operación son “Usuario” con su *Primary Key* “IDUsuario” (está tabla cuenta con dos *Foreign Key*, una de la tabla “Pais” y otra de “Tutor”), y la tabla “Puntuación” con su *Primary Key* “IDPuntos” (también cuenta con dos *Foreign Key*, una de la tabla “Niveles” y otra de “Usuario”).

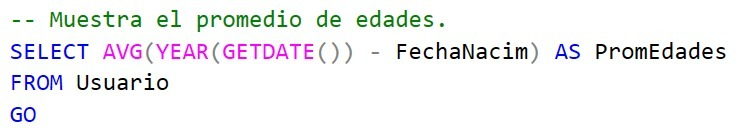
También existe una tabla llamada “Administrador” con su *Primary Key* “IDAdmin”, está tabla no se encuentra relacionada a las demás porque está funciona como un “mini sistema” dentro de la Base de Datos. Brinda funcionalidad a una sección designada en la página web, donde solo los administradores pueden registrar nuevos administradores, manipular datos (actualizar / borrar registros de los usuarios, tutores y países) y visualizar gráficas importantes sobre los usuarios.

A continuación se muestra de manera gráfica la base de datos y sus relaciones:

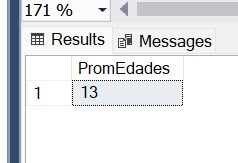
**Imagen 2.** Diagrama de la base de datos del proyecto Titor.

Queries propuestos para ser presentados en la página web del proyecto:

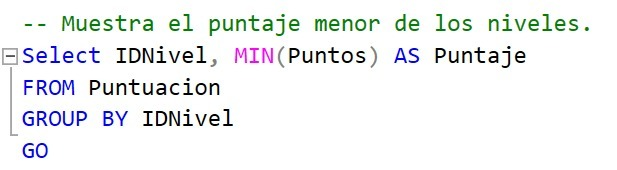
~Script~



~Resultado~



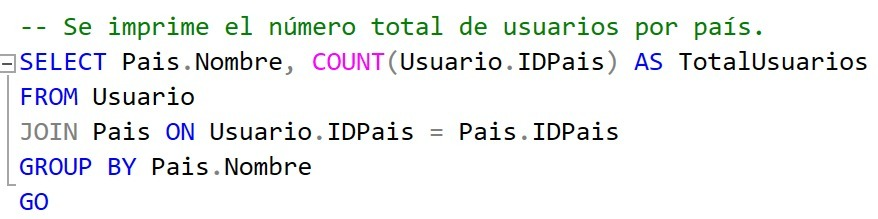
~Script~



~Resultado~



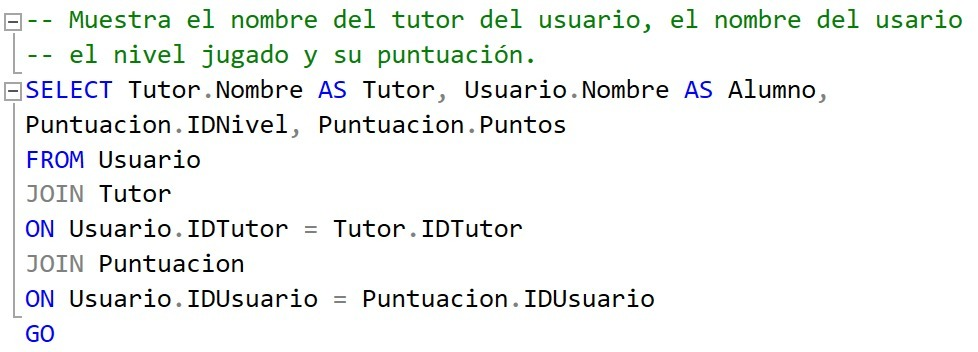
~Script~



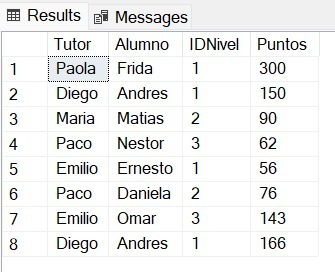
~Resultado~



~Script~



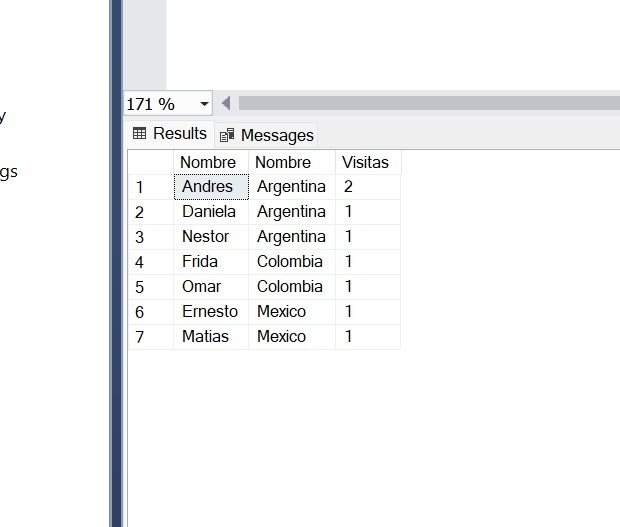
~Resultado~



~Script~

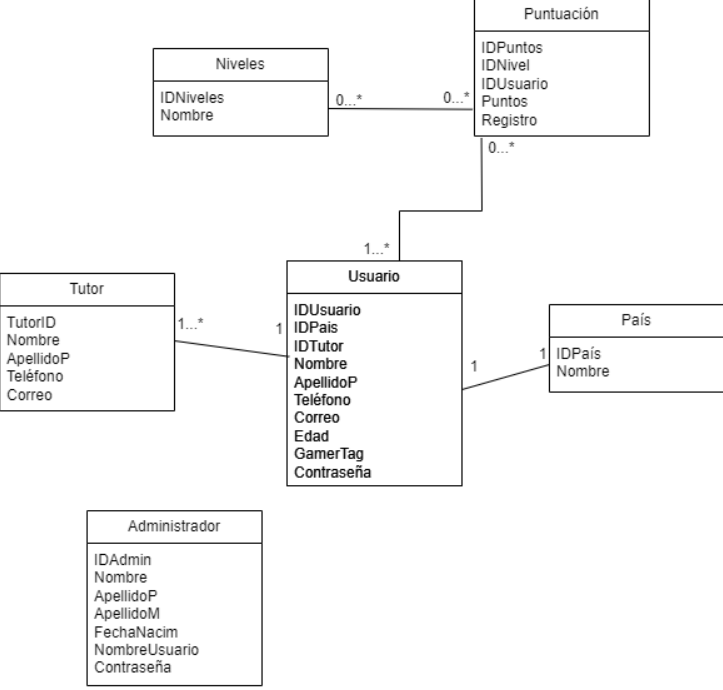


~Resultado~



# Diagramas de clases

En el diagrama podemos observar como las tablas anteriormente descritas en el diagrama de base de datos interactúan entre sí. En el caso de la tabla “Usuario”, se expresa que los usuarios pueden registrar únicamente un país, al igual que un tutor, pero un tutor puede tener uno o más usuarios y para la puntuación puede tener más de un punto. Y en el caso de las relación entre la tabla “Puntuación” y ”Niveles”, se expresa que pueden tener de cero o varios datos mutuamente.

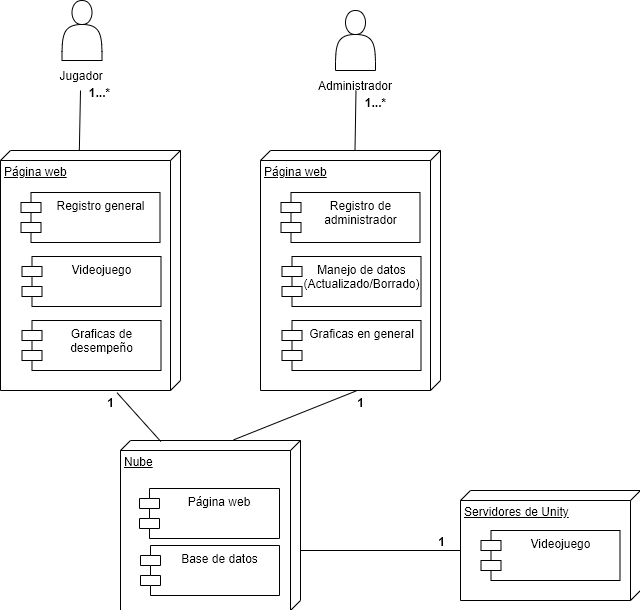


**Imagen 3.** Diagrama de clases.

# Diagrama de despliegue

En el diagrama de despliegue podemos observar como ambos usuarios que previamente describimos podrán interactuar dentro de los componentes que posteriormente se describirán más adelante en este documento.

Como podemos observar en el diagrama, ambos usuarios podrán acceder a sus respectivos apartados,el servidor donde se subirá el juego en donde se enviaran los datos que se requieren almacenar, y a su vez, los tres componentes están conectado a la nube, la cual será en donde se almacenarán los datos y mostrarlos después en la página web.



**Imagen 4.** Diagrama de despliegue.

### 

# Diagrama de componentes

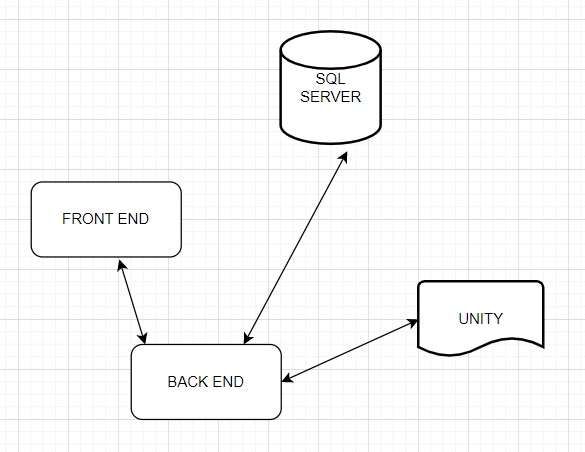
En el diagrama de componentes podemos observar cuatro entidades, sin las cuales nuestro proyecto no podría cumplir con lo estipulado, ya que cada entidad tiene funciones muy específicas e importantes para el desarrollo general del proyecto.

En este diagrama podemos observar que para la base de datos se utiliza el sistema de gestión de base de datos relacional "SQL Server", gracias a ese componente podemos crear, modificar y desarrollar la base de datos que se emplea para el reto.

Posteriormente se observa el componente de "Unity", en donde se desarrolla el videojuego que se centra en demostrar el porqué aprender a programar puede ser divertido.

Para el siguiente componente podemos observar el "Front End", en este componente se desarrolla la página web en la cual describe toda la información del reto, y los registros de cada usuario que desee jugar el juego. Para este componente se utilizará HTML, CSS y Javascript.

Finalmente observamos el componente "Back End" y como podemos observar, todos los componentes anteriormente mencionados están conectados al "Back End" ya que prácticamente este componente se encargará de conectar todo mediante el uso de "APIs" y enviarlo al "Front End". Para este componente se utilizará Javascript.



**Imagen 5.** Diagrama de componentes.

# Avance del videojuego

## Historia del videojuego

La historia comienza contigo, un trabajador en una empresa de tecnología avanzada que ha desarrollado una máquina del tiempo. Te apasiona la programación y estás emocionado por la oportunidad de trabajar en un proyecto tan innovador.

Un día, mientras trabajabas en la máquina del tiempo, descubriste un error en el código que podría causar un fallo catastrófico en la máquina. Decidiste arreglar el error por tu cuenta, pero al hacerlo, accidentalmente te envías a ti mismo a una época desconocida.

Después de tu viaje te despiertas dentro de una nueva época, esta es completamente diferente a lo que estabas acostumbrado. Sin embargo, pronto descubres que puedes usar tu conocimiento de programación para sobrevivir en esta nueva época y cualquier otra que se atraviese en tu regreso al presente.

A medida que avanzas en tu misión, te das cuenta de que tu conocimiento de la programación puede ser la clave para resolver los desafíos que enfrentan. La máquina del tiempo, que en su tiempo había sido un proyecto en desarrollo, ahora es una herramienta crítica que les permitirá navegar a través del tiempo y del espacio para salvar el mundo y encontrar la manera de regresar a casa.

Pero al explorar las diferentes épocas, descubres que hay un especie de enemigo que te dificulta el viaje entre las épocas. Se trata de una Inteligencia Artificial llamada **H.A.R.B** (Highly Advanced Robotic Brain) que busca adueñarse de tu época original. Para volver a casa deberás utilizar tus habilidades y conocimientos para encontrar una manera de detener esta amenaza y salvar tu propia época.

Con cada éxito, te das cuenta de que estás aprendiendo nuevas habilidades y técnicas de programación que nunca habías imaginado. La experiencia de viajar en el tiempo te ha abierto nuevas perspectivas y habilidades.

Finalmente, después de muchas aventuras y desafíos, encuentras una manera de regresar a tu propia época y enfrentarte a **H.A.R.B**. Pero, a pesar de todo lo que han aprendido, sabes que siempre habrá más por descubrir en la programación y en los viajes en el tiempo.

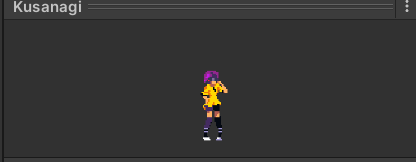
## Reto a resolver

Nosotros buscamos resolver el reto 2, nuestro juego tiene el objetivo de dejar en las personas una perspectiva positiva de la programación.

El juego consiste en obtener diversas partes de un código el cual se encuentra disperso por el nivel, con ello los jugadores obtienen notas importantes sobre cómo funciona y un cierto número de hojas que formarán parte de un puzzle a resolver, pues con los fragmentos dispersos de ese código se formará una imagen con el código completo. Con ello se tratará de explicar en 3 secciones los fundamentos básicos de la programación, divididos en variables y operadores, ciclos y funciones.

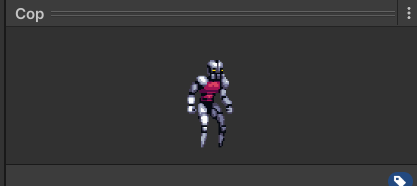
## Desarrollo de personajes

1. Kusanagi (Mujer):



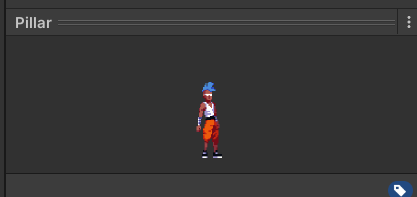
*Habilidades*: Mayor velocidad, doble salto y daño normal.

1. COP (Sin género):



*Habilidades*: Mayor daño, menor movimiento y mayor resistencia a los ataques.

1. Pillar (Hombre):

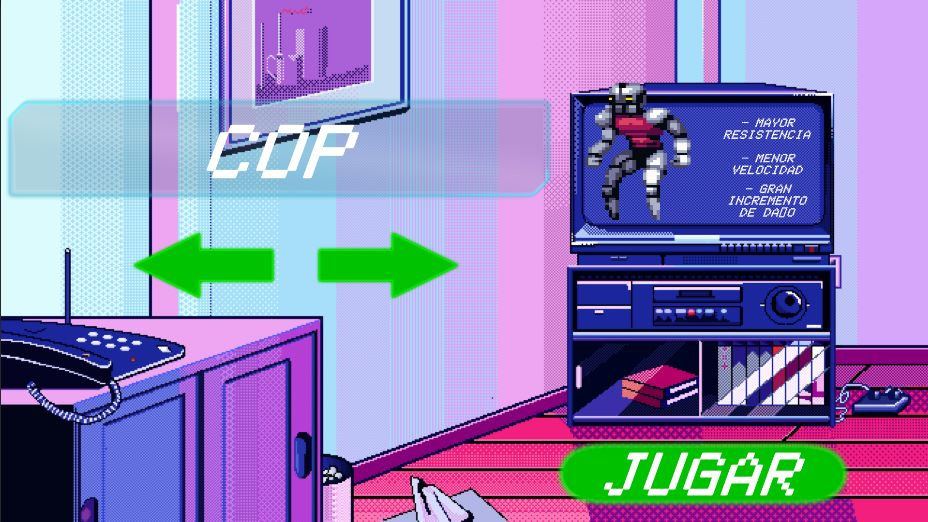


*Habilidades:* Más salud, velocidad normal y daño normal.

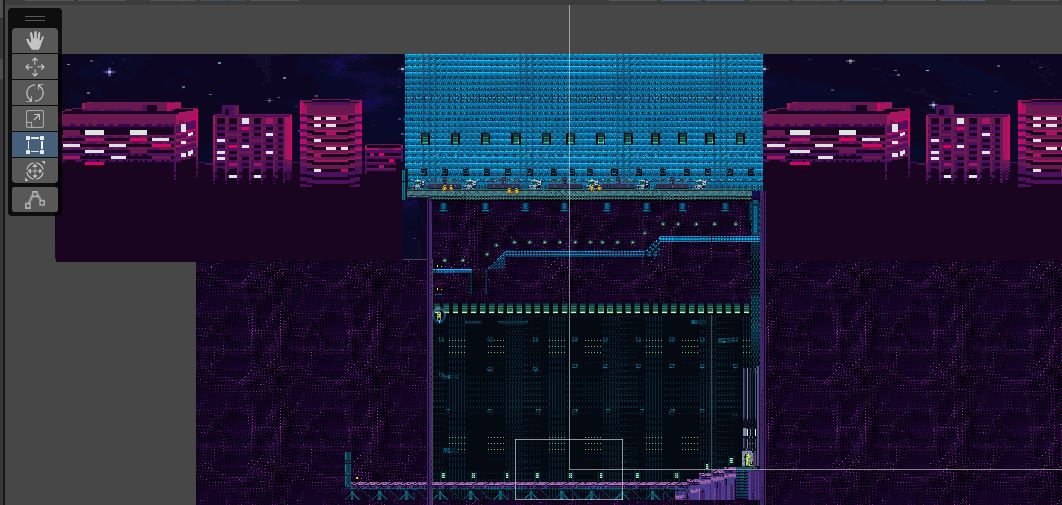
## Prototipo

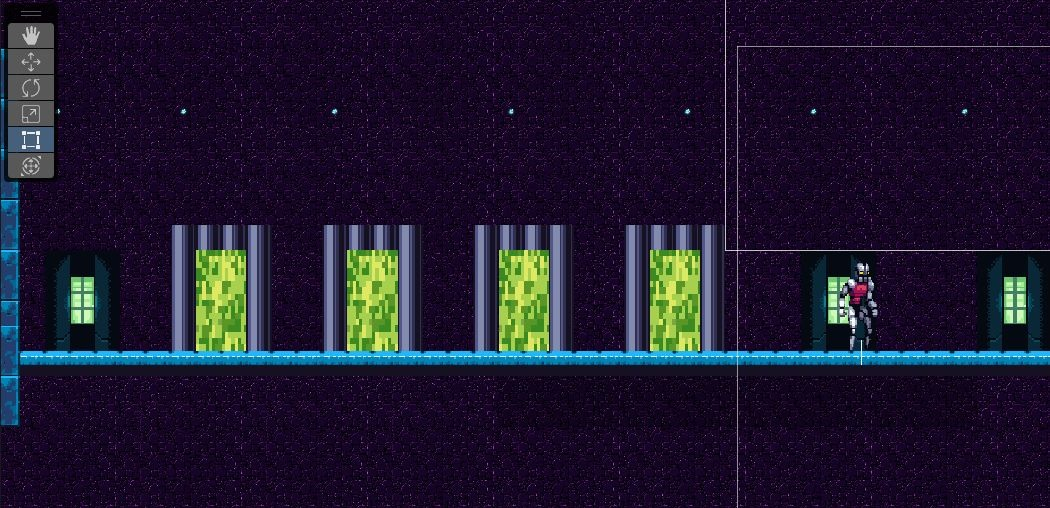


1. Pantalla principal del juego donde se dan las opciones y el comienzo del mismo.



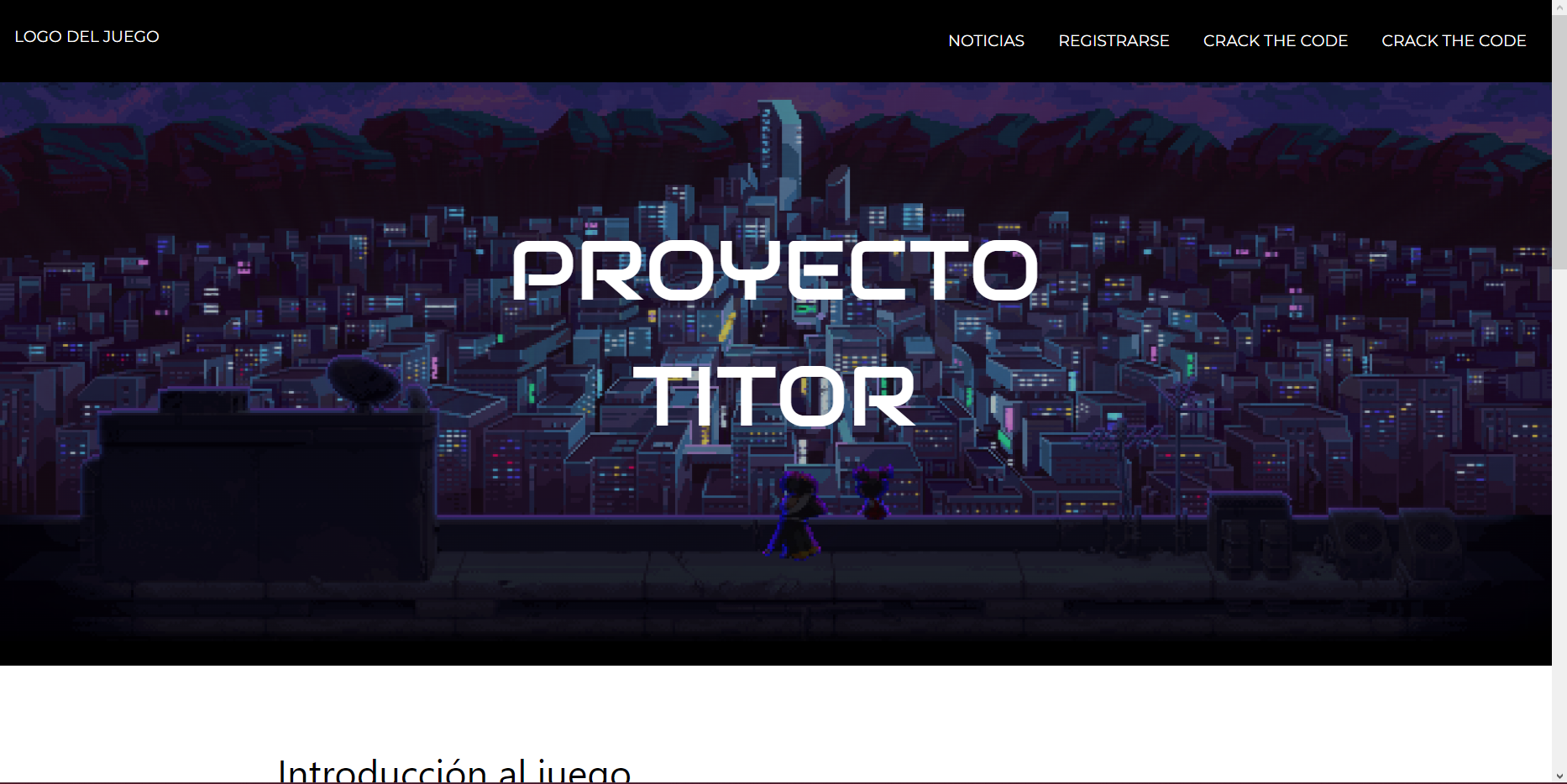
1. Selección de personajes que han sido descritos anteriormente, para empezar con el primer nivel, el usuario deberá oprimir el botón “Jugar”.





1. Ejemplos de los escenarios usados en el juego.

# Prototipo de la plataforma tecnológica







Hasta este momento, ya contamos con un prototipo de plataforma digital. Si bien esto sigue siendo un prototipo, la plataforma debe resultar atractiva para el público y proveerá la facilidad de registrarse por primera vez a los usuarios y poder probar nuestro juego, el cual, deberá correr en esta misma.

Se planea que la página principal se divida en cuatro secciones. La primera consiste en una barra de navegación estática junto con una imágen alusiva al juego que atrape a los nuevos jugadores, posteriormente contamos con una sección que contendrá una breve sinopsis de nuestro juego y seguido de esta, otra que menciona los diferentes beneficios a futuro de aprender programación a temprana edad.

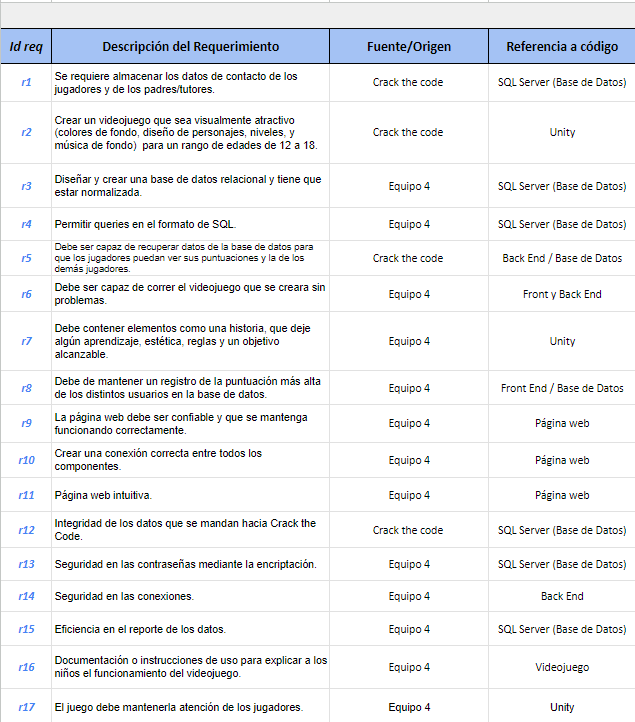
Por último, se encuentra la sección dedicada al registro del público y unos logos de las distintas redes sociales tanto del juego, como de Crack the Code. En cuanto a los datos guardados, registramos el nombre, primer apellido, teléfono y correo electrónico del interesado. Si el interesado es menor de edad, se le solicitará registrar los mismos datos de su tutor.

# 

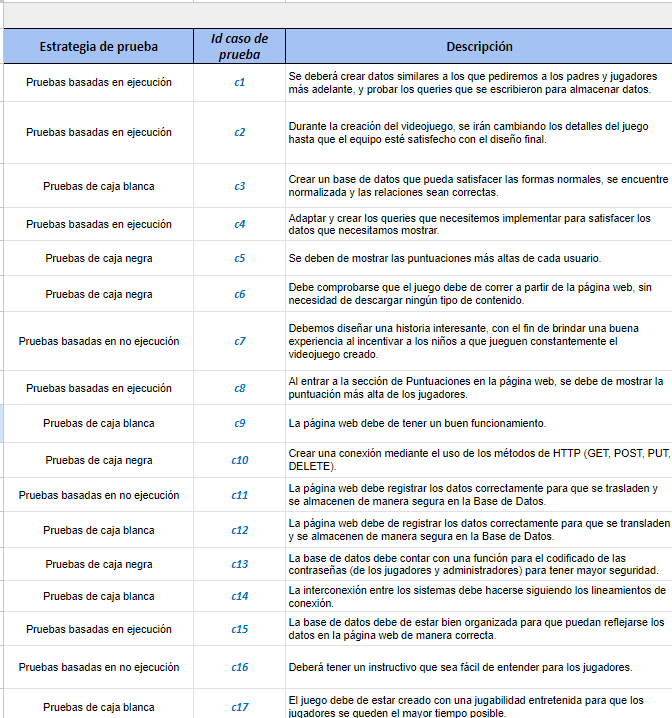
# ​​Definición de las pruebas unitarias

Para el desarrollo de las pruebas unitarias que se llevarán a cabo, se implementó la siguiente tabla para describir y adecuar mejor lo que se necesita mejorar o adaptar para cada componente durante el desarrollo del proyecto.

Dentro de esta tabla se describe cada uno de los requerimientos funcionales y no funcionales que anteriormente detallamos en el documento, posteriormente se detalló más sobre qué parte de los componentes se podrán hacer las pruebas que se plantearon en la tabla, a la par de introducir lo que se necesita como datos de entrada y detallar lo que se tiene planeado como datos de salida, y en caso de que se llegaran a detectar posibles defectos, poder añadir las observaciones necesarias con las posibles formas de arreglar los defectos que encontramos y algunas observaciones detalladas en las que pudimos o no pudimos arreglarlos y el porqué de esos resultados.

****

**Imagen 5.** Primeras cuatro columnas donde se describen los requerimientos funcionales y no funcionales, además de su origen y en qué parte de los componentes se ejecutarán las pruebas.



**Imagen 6.** En las columnas se describe cómo será la estrategia que se usará para hacer las pruebas y la descripción de los requerimientos funcionales y no funcionales.



**Imagen 7.** En las columnas se observa los datos de entrada que se necesitan para ejecutar las pruebas, las salidas que necesitamos obtener y las que obtuvimos, y finalmente los defectos que se encontraron con sus respectivas observaciones en caso de ser necesarias.

Para las pruebas mostradas anteriormente, se puede observar en la descripción de las capturas como fue la ejecución de dichas pruebas a la par de cuáles fueron los resultados, independientemente de que si los resultado fueron buenos o malos, podemos determinar que en la gran parte de las pruebas, fueron muy favorecedoras, y en los pocos casos en los que no se concretaron, decidimos escribir cuáles fueron los cambios que se hicieron para al entregable final.

Para una mejor visualización entrar al siguiente link:

[Pruebas\_Unitarias\_E4.xlsx](https://docs.google.com/spreadsheets/d/1CBsBfI8s3cmBcGfh5tB4agspBqRZudpK/edit?usp=sharing&ouid=104593699076371374635&rtpof=true&sd=true)

# 

# Licencia de software

Para el desarrollo de este proyecto, utilizamos los siguientes softwares:

* Unity Software Inc. (Unity Technologies):

Plataforma de desarrollo de juegos y aplicaciones interactivas para crear experiencias 2D y 3D en múltiples plataformas, incluidas PC, consolas, dispositivos móviles y realidad virtual y aumentada.experiencias en 2D y 3D en múltiples plataformas, como PC, consolas, dispositivos móviles y realidad virtual y aumentada.

* SQL Server Management Studio. (SSMS):

Una herramienta de desarrollo y administración de bases de datos creada por Microsoft para su uso con Microsoft SQL Server. Esta herramienta se utiliza para administrar y configurar bases de datos, ejecutar consultas y secuencias de comandos transact-SQL, diseñar tablas y vistas y crear informes.

* Amazon Web Services (AWS):

Plataforma de computación en la nube que proporciona una variedad de servicios y herramientas para alojar y administrar aplicaciones y servicios en línea.

* Hypertext Markup Language (HTML):

Lenguaje de marcado estándar para crear páginas web y aplicaciones web. Es un lenguaje de programación para crear y presentar contenido web como texto, imágenes, video y otros elementos multimedia.

* Cascading Style Sheets (CSS):

Lenguaje de hoja de estilo utilizado para definir el diseño visual de una página web o aplicación web. Los desarrolladores web lo utilizan para determinar el estilo y la apariencia de los elementos HTML, como el tamaño y el tipo de fuente, el color, el diseño y la posición del elemento en la página.

* Javascript:

Lenguaje de programación de alto nivel utilizado principalmente para crear aplicaciones web y agregar interactividad a los sitios web. Es un lenguaje de programación de secuencias de comandos interpretadas que se ejecuta en el navegador del usuario, lo que permite que el contenido web se actualice y cambie dinámicamente en función de las acciones del usuario.

* Draw.io:

Herramienta de diagramación en línea gratuita y de código abierto para crear diagramas y diagramas de flujo para varios propósitos. Lo utilizan desarrolladores, diseñadores, ingenieros, arquitectos y otros profesionales que necesitan crear diagramas y diagramas de flujo para sus proyectos.

* W3Schools:

Sitio educativo en línea con tutoriales y ejemplos de varias tecnologías web que incluyen HTML, CSS, JavaScript, SQL, PHP y más. Este sitio es una referencia popular para desarrolladores web y estudiantes que desean aprender y mejorar sus habilidades de desarrollo web.

* Express:

Marco de aplicaciones web para Node.js se utilizan para crear aplicaciones web y servicios API de forma rápida y sencilla. Proporciona muchas funciones y herramientas útiles para el desarrollo web, lo que la convierte en una herramienta popular para los desarrolladores web que utilizan Node.js.

# Derechos de autor

Para el desarrollo de este proyecto se utilizaron elementos gratis de distintos lugares:

* Para el desarrollo del videojuego, se utilizaron assets y música obtenidas de la “Asset Store”.
* Para las imágenes desarrolladas para el apartado del rompecabezas, las generamos por nuestra cuenta.
* Para el uso de AWS, los sistemas nos fueron facilitados por el profesor Ariel Ortiz Rámirez.