**Invasión Alienígena**

**Integrantes del proyecto**

* **Akira Molas**
* **Esteban Rodríguez**
* **Nicolás Romero Ortiz**
* **Oscar Ruina**

**Resumen del proyecto**

Se nos ha asignado en la materia Seminario de Lenguajes realizar un juego desarrollado en el lenguaje Python. Para realizar esto, hemos utilizado un controlador de versiones GIT junto con la metodología ágil Kanban para mantener la documentación requerida. Durante el transcurso del ciclo de vida del proyecto, hemos tenido algunos problemas en la programación. Algunos los hemos solucionado, otros problemas que no hemos podido solucionar en tiempo y forma se han dejado para futuras investigaciones. El resultado es un juego básico del tipo arcade codificado en Python.

**Abstract**

We have been assigned in the subject Seminar of Languages ​​to realize a game developed in the Python language. To do this, we have used a GIT version driver along with the agile Kanban methodology to maintain the required documentation. During the course of the project life cycle, we have had some problems in programming. Some we have solved, other problems that we have not been able to solve in time and form have been left for future investigations. The result is a basic arcade type game coded in Python.

**Key words**

* Python
* Pygame
* GIT
* GitHub
* SourceTree
* Trello
* Kanban
* IDLE de Python
* Eclipse
* Juego/GAME
* Repositorio/Repository

**Introducción**

Para realizar este proyecto, lo primero que se hizo fue crear un repositorio en la plataforma GitHub y descargar e instalar el cliente GIT SourceTree. El nombre del repositorio es “Viaje al Centro de la Bombonera”, debido a que en un principio teníamos pensado hacer un juego con temática de fútbol (sobre barrabravas), pero como se explicará más adelante la idea del juego cambió a medida que pasaba el tiempo.

Cada integrante del grupo para familiarizarse con este controlador de versiones creó y modificó archivos de prueba. Luego, se eligió el IDE para programar en Python y se descargó e instaló la librería Pygame correspondiente.

Como grupo hemos discutido aprender el lenguaje en un tiempo máximo de dos semanas aproximadamente.

***PRIMEROS NOMBRES E IDEAS DEL JUEGO***

Los posibles nombres del juego son:

- Hoolingan's trip

- Voyage to Manchester

La posible idea acerca de qué va a tratar el juego es sobre un fondo estático de un tren con personas. De repente aparecen los 'hoolingans' y comienzan a pelear. La meta es llegar al estadio de fútbol en tren, para ello deberá recorrer los 5 vagones del transporte y enfrentarse a sus rivales para sobrevivir, además de un 'boss' en cada vagón que le impondrá más dificultades a superar. Al final del último vagón, estará el 'major boss' que determinará si llega o no al destino.

***CAMBIOS DEL PROYECTO***

La decisión que se tomó en grupo es la de cambiar el IDLE de Python por el IDE Eclipse.

A causa de este cambio, repensamos la idea principal del juego a algo con mayor acción y movimiento en pantalla. En los siguientes días comenzaremos a iterar los primeros sistemas del juego y establecer una idea concreta.

Títulos provisorios:

- BarraBros

- Super Barrabrava Bros

Por último, el escenario del juego no será un tren como en un principio sino sobre una imagen de fondo con temática de fútbol (por definirse).

***APRENDIZAJE SOBRE PYTHON***

Dada la nula experiencia en el lenguaje de programación Python como así también en el manejo de librerías el grupo pasará durante unas semanas leyendo y practicando con tutoriales en Internet para tener los conocimientos necesarios para el desarrollo del juego según lo solicitado en la materia.

Debido al diseño rústico y simple del IDLE de Python, en una reunión con los miembros del equipo se decidió cambiar al IDE Eclipse gracias al consejo de un compañero ajeno al proyecto.

***CAMBIO DE TEMÁTICA E IDEA DEFINITIVA DEL JUEGO***

En conjunto con los integrantes del grupo hemos decidido cambiar la temática del juego por cuestiones de dificultades y tiempo, ya al ser la primera vez que utilizamos el lenguaje de programación Python y la librería Pygame la anterior idea probablemente llevaría un plazo mayor al pedido en el proyecto.

En pleno desarrollo, la temática pasará a ser la de una batalla espacial dónde un protagonista humano a bordo de una nave deberá defender al planeta Tierra de los ataques alienígenas de todas partes de la Galaxia que se disputan a toda costa el control de la misma. Al estilo 'survival mode', se deberá permanecer el mayor tiempo posible atacando a los "enemigos" hasta derrotarlos a medida que la dificultad vaya en aumento.

Tras la siguiente reunión, como se explica en la minuta anterior, las primeras ideas del juego fueron abandonadas y reemplazadas por otras ideas más simples.

Después de esta reunión, uno de los objetivos era el de tratar de investigar este tipo de juegos mediante blogs, tutoriales y todo lo relacionado con ello.

Hemos podido programar pequeños procedimientos básicos de Pygame, como por ejemplo la creación de ventanas, carga de imágenes de fondo en ellas, implementación de sonidos y colocación de personajes inmóviles.

***REGRESO AL IDLE DE PYTHON***

Tras una reunión con todos los miembros del grupo hemos decidido regresar al IDLE que viene por defecto con Python como plataforma para el desarrollo del juego debido a las múltiples dificultades que se presentaron con las extensiones que se instalaron en el IDE Eclipse, los cuales son necesarios para que pueda interpretar dicho lenguaje. Dada también la inexperiencia a la hora de instalar complementos allí, sumados a los múltiples errores que surgían para compilar el código han influido a volver al IDE original. Si bien en nuestra opinión nos parecía demasiado básico, nos convencimos de la practicidad a la hora de realizar el desarrollo y la compilación del proyecto.

Luego de este cambio, el IDLE de Python nos pareció fácil y simple de usar a pesar de su rusticidad.

***DEBATE ACERCA DE LA ESTRUCTURA DEL JUEGO***

Tras una clase en la universidad, los miembros del grupo hemos conversado sobre definir la estructura que tendrá dicho proyecto. Constará de un menú principal con los siguientes accesos:

- Nuevo juego

- Puntuación

- Ayuda

- Créditos

- Salir del juego

Habrá dos niveles de dificultad: el más fácil que se inicia al arrancar una nueva partida y a medida que se vayan eliminando los alienígenas a combatir se acumularán puntos. Luego de haber terminado con todos los “enemigos” allí se pasará al siguiente nivel, donde la velocidad de los oponentes aumentará drásticamente poniendo a prueba la destreza de los jugadores hasta exterminar completamente a todas las amenazas que buscan de todas las formas posibles conquistar y dominar al planeta Tierra.

En la puntuación se guardarán los 5 mayores puntajes de cada partida jugada, mientras que en la sección de ayuda se podrá consultar el manual de usuario. En la parte de créditos se pondrán los nombres completos de cada miembro del proyecto y en la salida se le preguntará al usuario si quiere salir del programa o continuar jugando.

Tras consultar varios blogs y tutoriales, hemos logrado programar el esquema del juego, pantalla de inicio e implementado algunas imágenes provisorias junto con un programa básico de disparos.

***AVANCE EN EL DISEÑO DEL JUEGO***

Luego de las primeras pruebas del programa principal (partida del juego) con elementos básicos como protagonistas en el juego se han subido a GitHub los primeros diseños preliminares de los elementos definitivos que compondrán la pantalla de cada partida: los enemigos a combatir, sus disparos, la nave que deberá combatirlos (usuario) y también el fondo de temática espacial junto con la música de fondo. Debido a que aún quedan muchos detalles para pulir y corregir, estas imágenes son provisorias para continuar con el desarrollo y la prueba del juego. Todos estos archivos han tenido el visto bueno por unanimidad de todo el grupo de trabajo.

Para mantener y actualizar la documentación del proyecto se eligió la metodología ágil Kanban, para ello gracias al consejo de otra compañera ajena al grupo nos recomendó usar la aplicación Trello de Bitbucket para organizar las correspondientes tareas.

***DIVISIÓN DE TAREAS***

Dado un común acuerdo entre todos los integrantes, la división de tareas en el grupo del proyecto es la siguiente:

• **Creación de tareas en GIT:** Akira Molas

• **Actualización de minutas:** Esteban Rodríguez

• **Realización de las imágenes y del sonido del juego:** Nicolás Romero Ortiz

• **Desarrollo del manual de uso y de la historia:** Oscar Ruina

• **Aprendizaje del lenguaje Python y de la librería Pygame**: todos

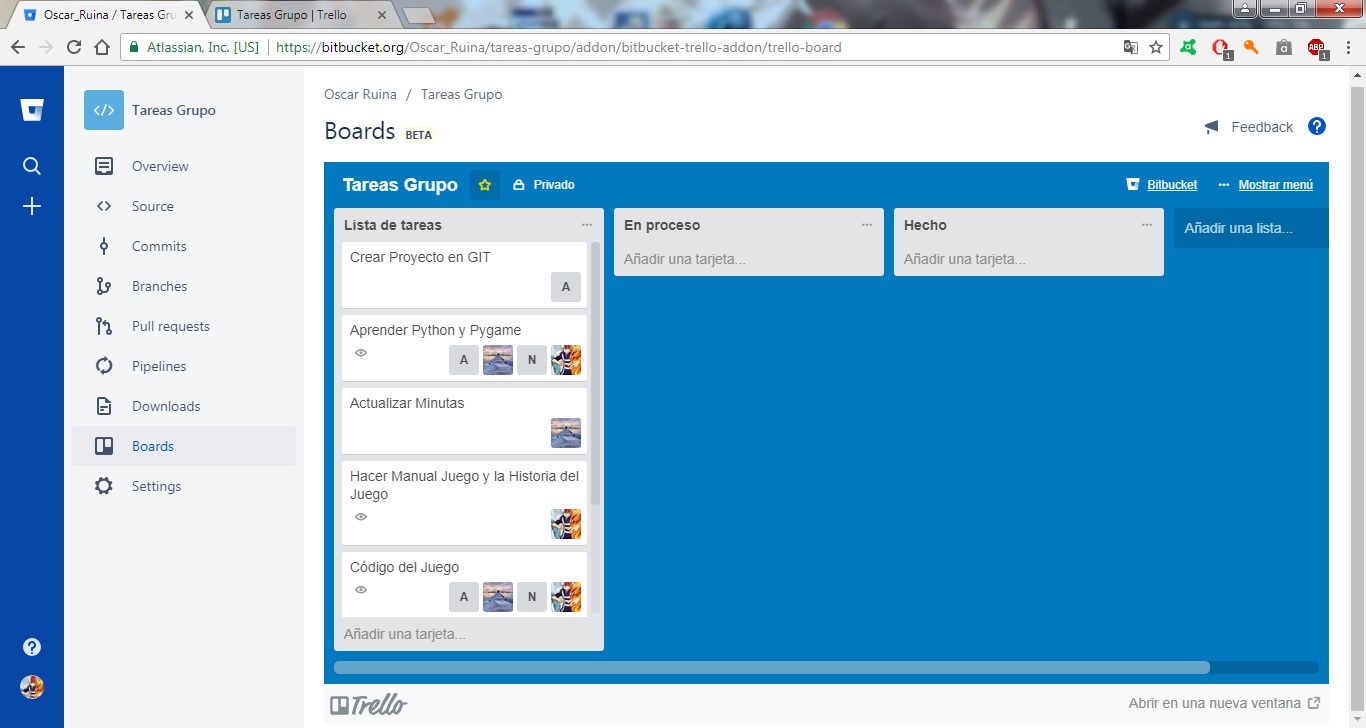
• **Codificación inicial y división por clases:** Akira Molas y Oscar Ruina

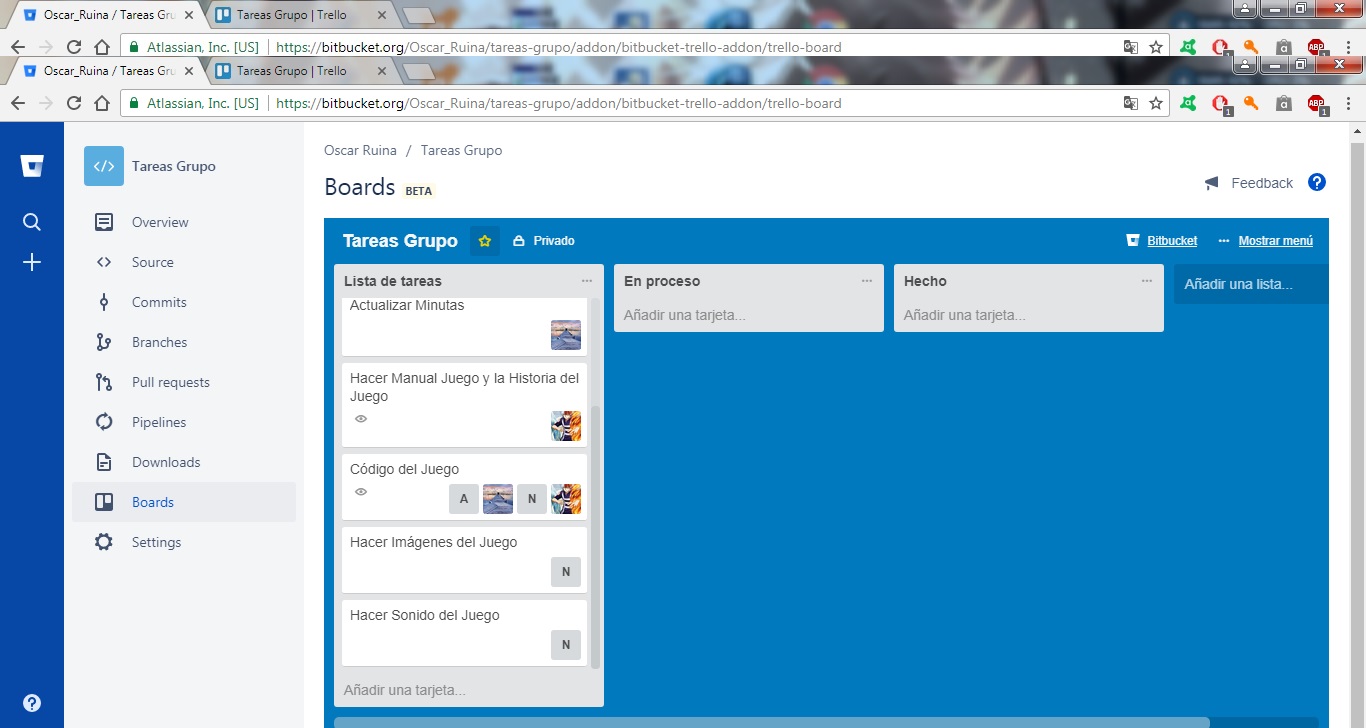
• **Elaboración de la pantalla de inicio:** Nicolás Romero Ortiz

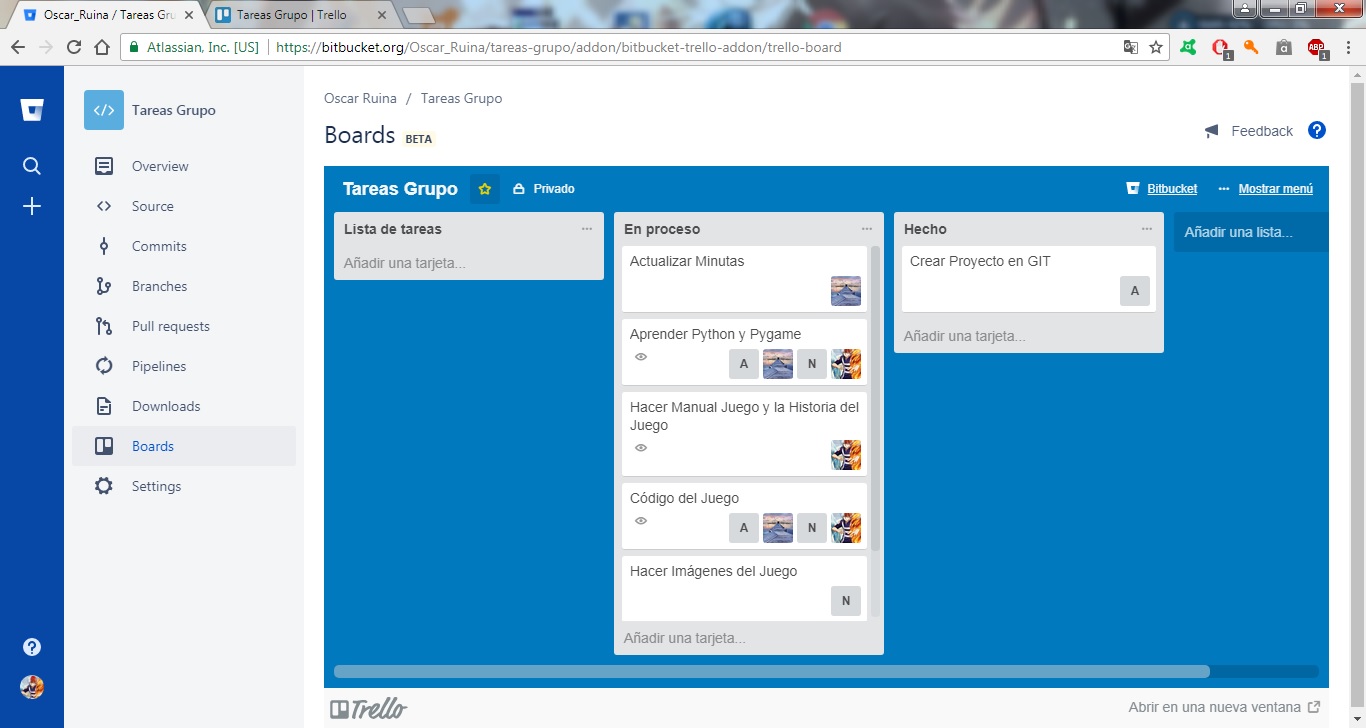
• **Desarrollo del reinicio del juego:** Oscar Ruina

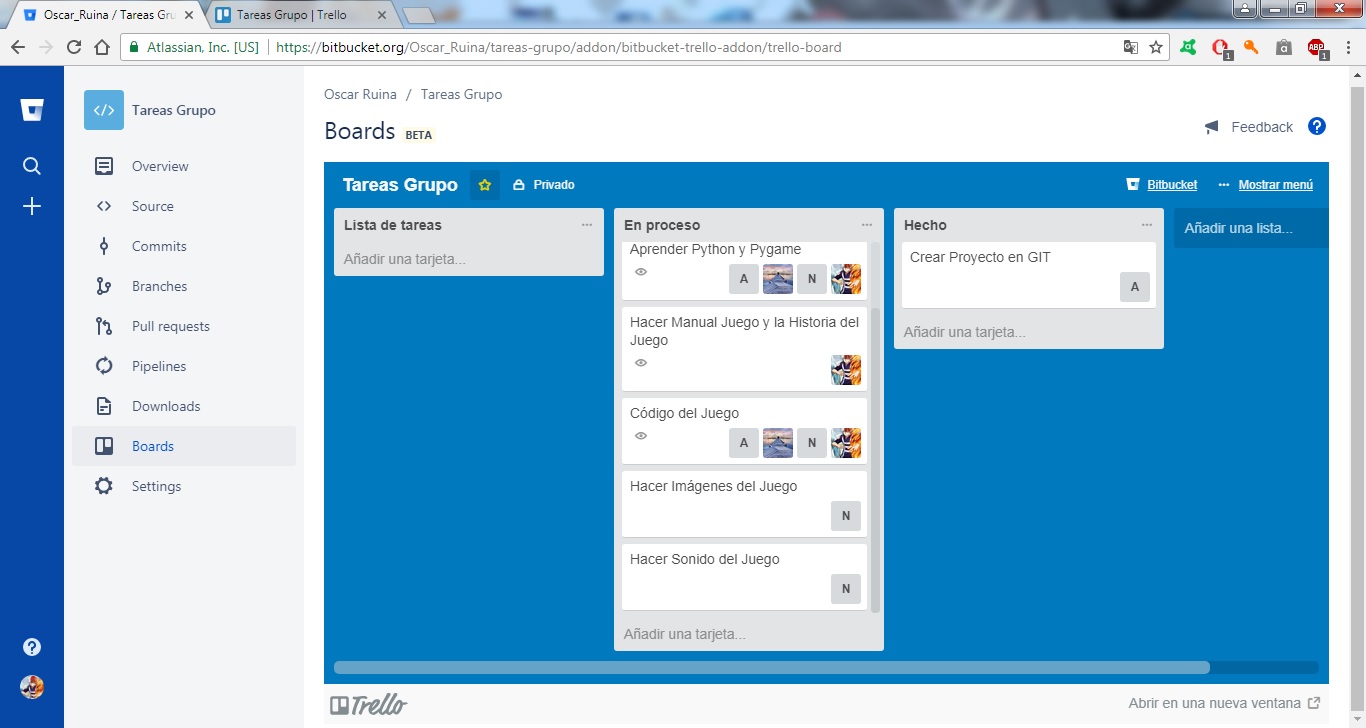
• **Codificación de la pantalla de inicio:** Akira Molas

• **Informe final del proyecto:** todos









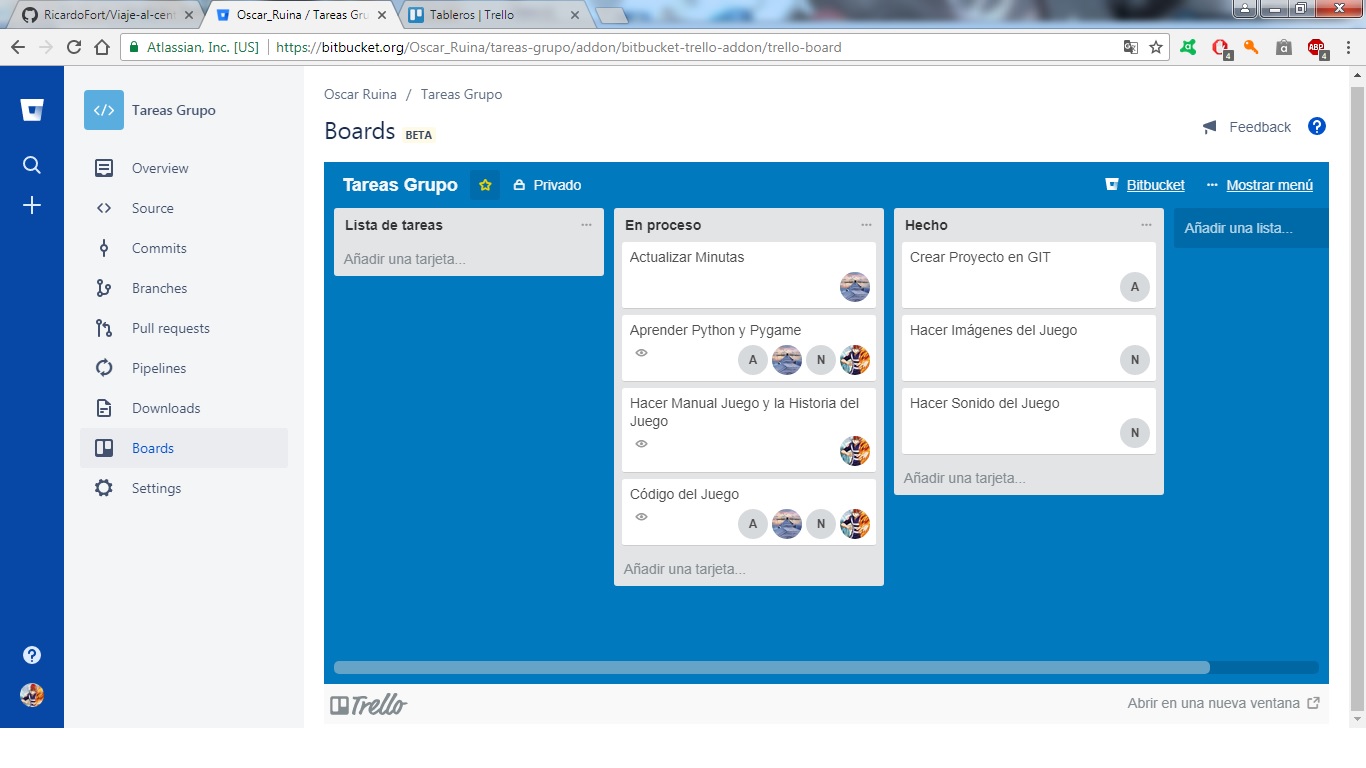
Con respecto al desarrollo del juego, lo que se realizó fue usar programación orientada a objetos para realizar el mismo dividiendo el código en clases.

De esta manera para ese entonces teníamos un juego casi hecho faltando las imágenes finales, el reinicio del juego en caso de ganar o perder, la imagen de presentación, niveles y puntuación.

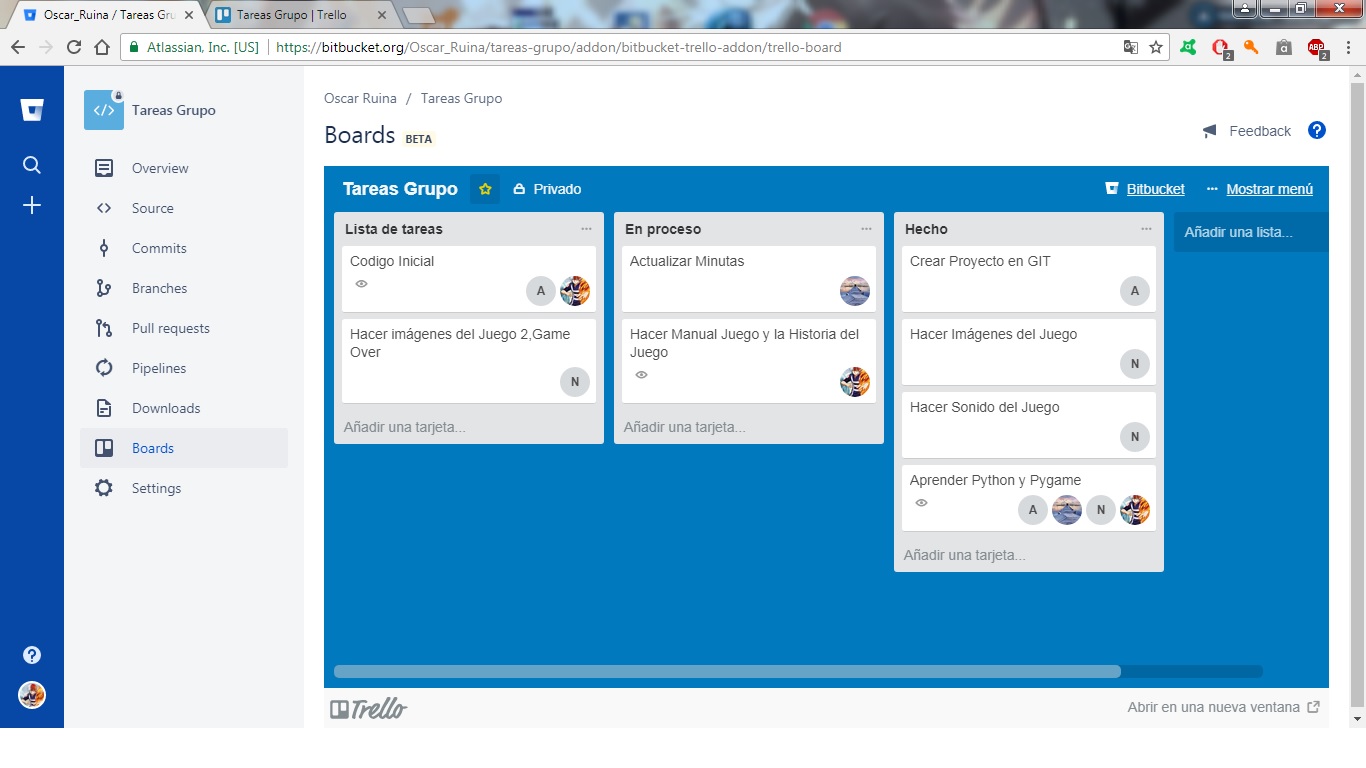
***DIFICULTADES EN EL CÓDIGO***

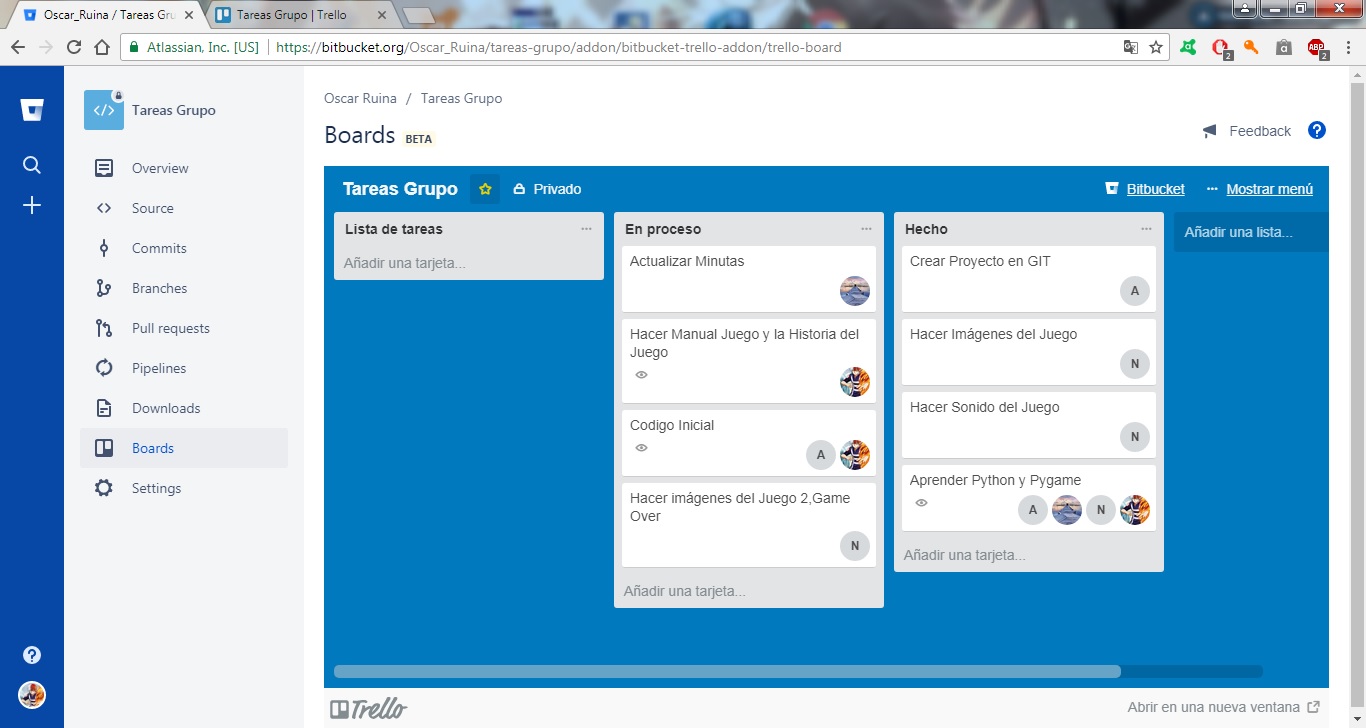
El grupo, junto con las personas encargadas de dicha tarea, han experimentado algunas complicaciones en el desarrollo del juego. Específicamente, los conflictos se evidencian en implementación del menú principal, el movimiento de los proyectiles destinados para atacar a las naves enemigas y el reinicio del juego, el cual por el momento no hace posible que se finalice el ciclo de las partidas y su posterior turno siguiente, las cuales marcan un notorio retraso en el tiempo disponible para la entrega.

Luego de varias pruebas, el grupo continúa avanzando en cumplir los objetivos propuestos dentro de los plazos establecidos. Se continúa con el mejoramiento de las imágenes y se avanza en la corrección de errores según los conocimientos adquiridos.



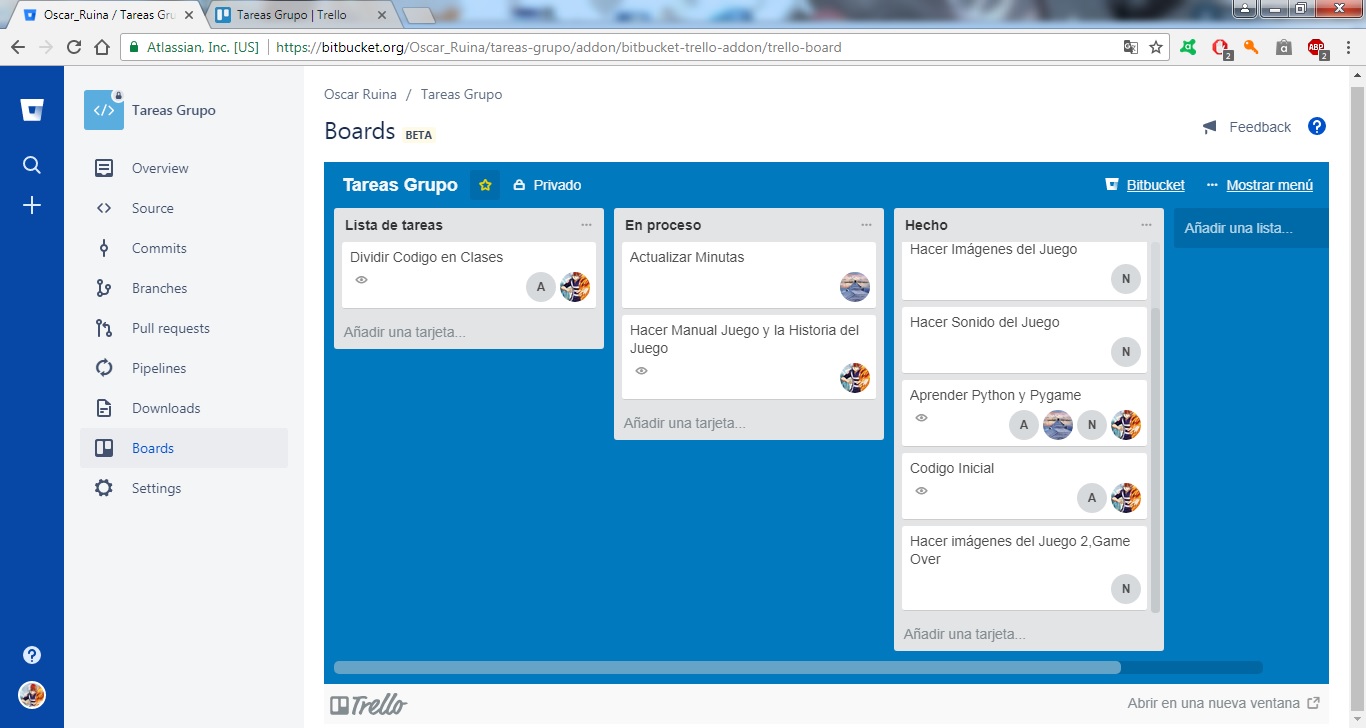
Con algunas imágenes ya realizadas (game over, win, nave espacial, enemigos y proyectiles) surgió el problema que se ha podido solucionar rápidamente modificando las dimensiones de las imágenes de los proyectiles, ya que anteriormente no lograban compilar correctamente. Con respecto al reinicio del juego, se pudo solucionar creando un loop cuando se perdía o se ganaba, limpiando la pantalla de enemigos y actualizando la misma.

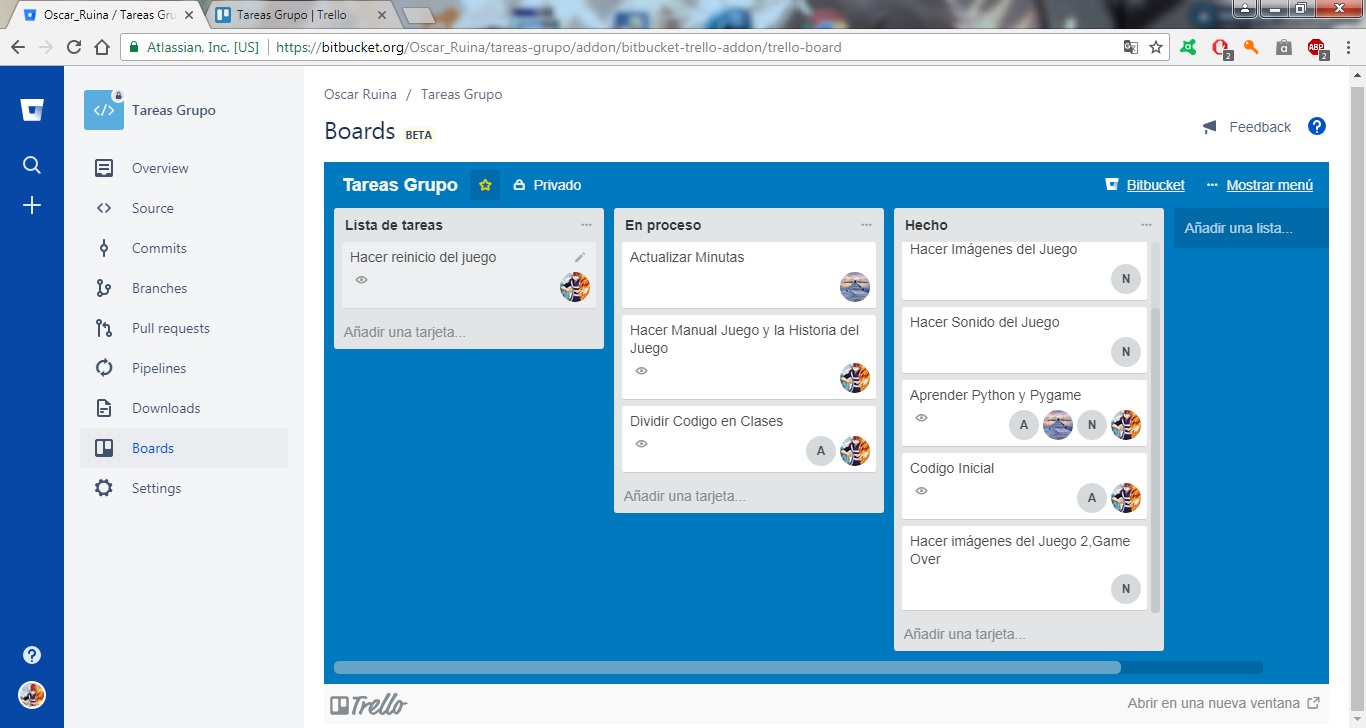




***COLOCACIÓN DE NIVELES***

Se decidió en una posterior reunión que el juego conste de dos niveles: uno con dificultad intermedia (inicial) y el otro con mayor complejidad (dificultad alta), dado el código fuente ya establecido y probado. Por otro lado, se ha logrado solucionar el problema de reinicio del juego, pudiendo ahora volver a iniciar la partida nuevamente luego de haberla perdido (pantalla con la leyenda “GAME OVER”). Las imágenes continúan en proceso de mejora como así también la división por clases del desarrollo en Python.





Se logró dividir el juego en clases. Una clase corresponde a la nave espacial, la segunda a los enemigos y la tercera a los proyectiles. Se colocó una imagen de fondo y se realizaron imágenes de los niveles.

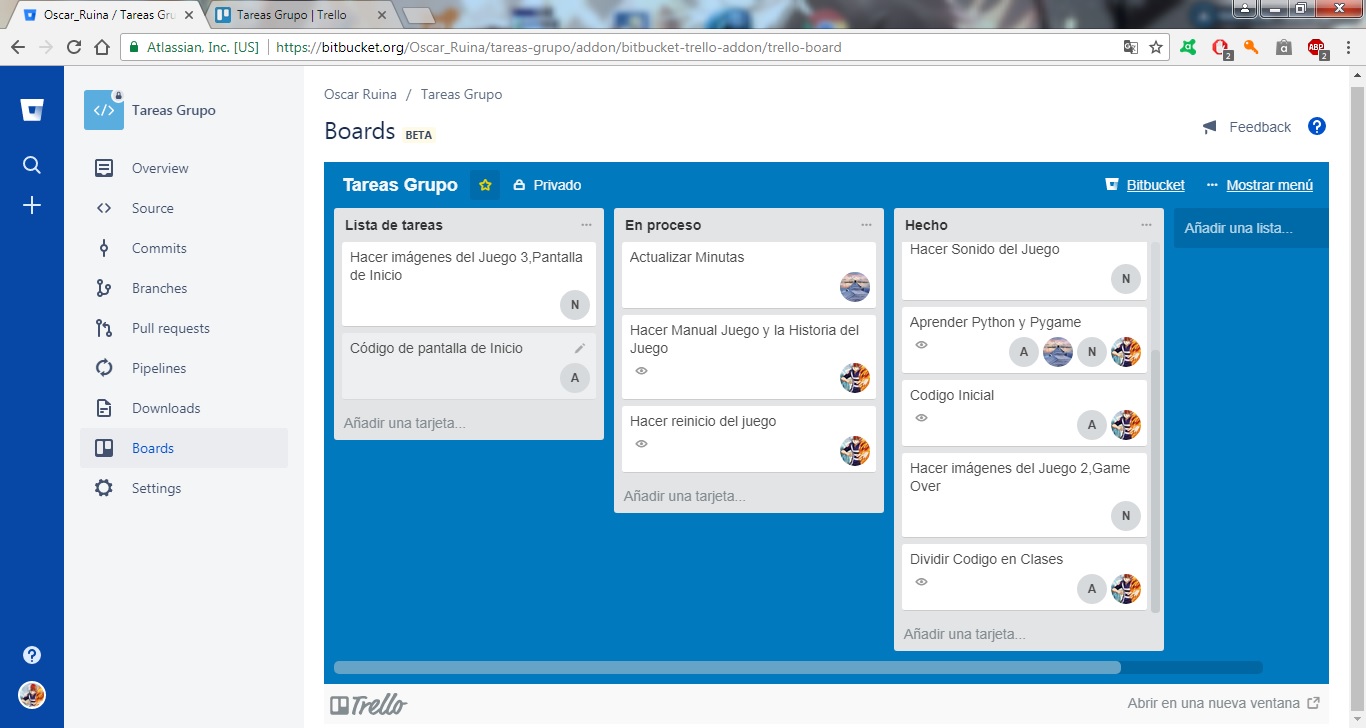
***PROBLEMAS CON LOS NIVELES***

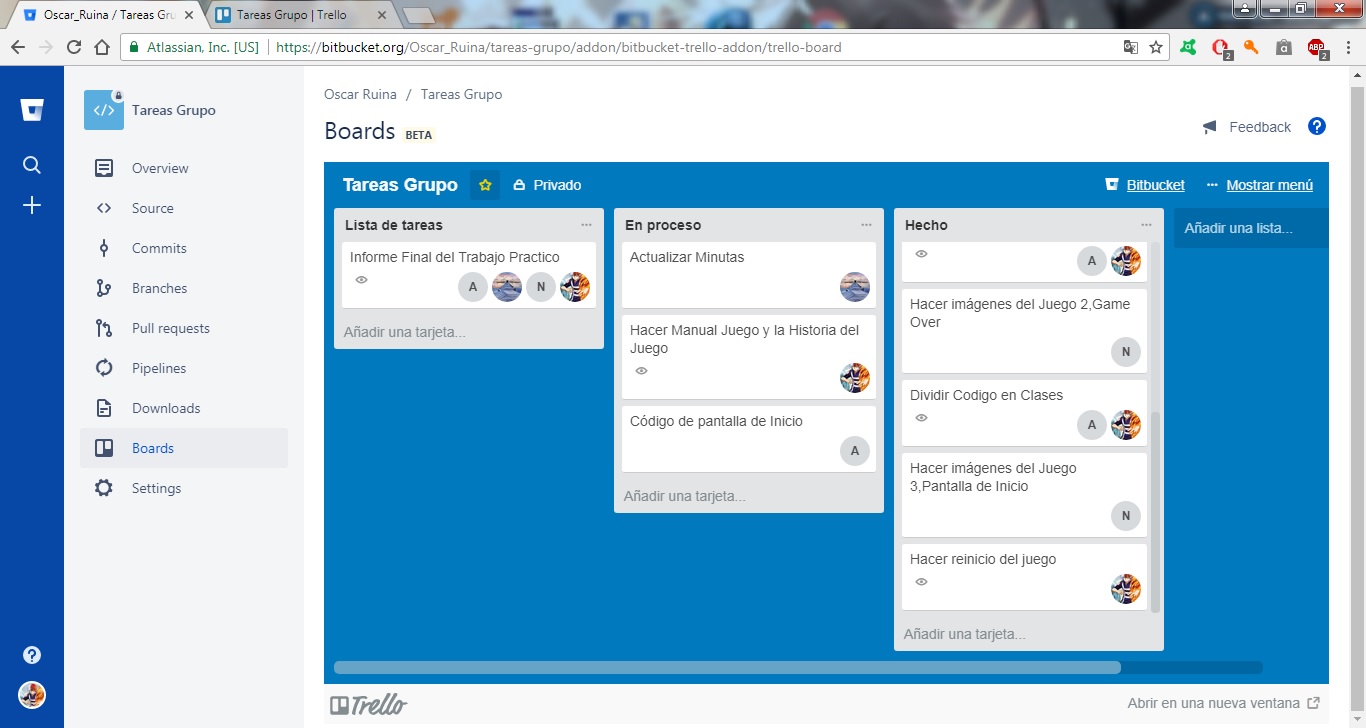
Luego de varias pruebas y correcciones realizadas, el nivel de mayor dificultad no logra ser desarrollado como se esperaba dadas las limitaciones de los conocimientos adquiridos hasta el momento. Por lo tanto, el grupo decidió eliminar del proyecto este tramo, quedando así un único nivel de dificultad intermedia para evitar mayores retrasos dentro del plazo de entrega estipulado. También, se avanza en la codificación y en la creación de imágenes para la pantalla de inicio.

Los niveles quedarán para futuras líneas de investigación.

***ACERCA DE LA PANTALLA DE INICIO***

La pantalla de inicio del juego consistirá en una primera imagen con su nombre seguido de una pequeña instrucción para avanzar y otra transición para comenzar la partida. La misma estará acompañada por otra imagen de fondo con la temática “espacial” que caracteriza al desarrollo a presentar. Quienes se encargarán de esta parte serán Nicolás Romero Ortiz creando las imágenes necesarias a través de la herramienta Photoshop y Akira Molas en la codificación necesaria para las animaciones y transiciones hasta llegar al escenario de batalla.





Esto se realizó en tiempo y forma, se cargó la pantalla de inicio y se implementó en el código. Tras varias pruebas, funcionó de manera correcta. Para esta altura del proyecto, el juego que pensamos desarrollar estaba completo.

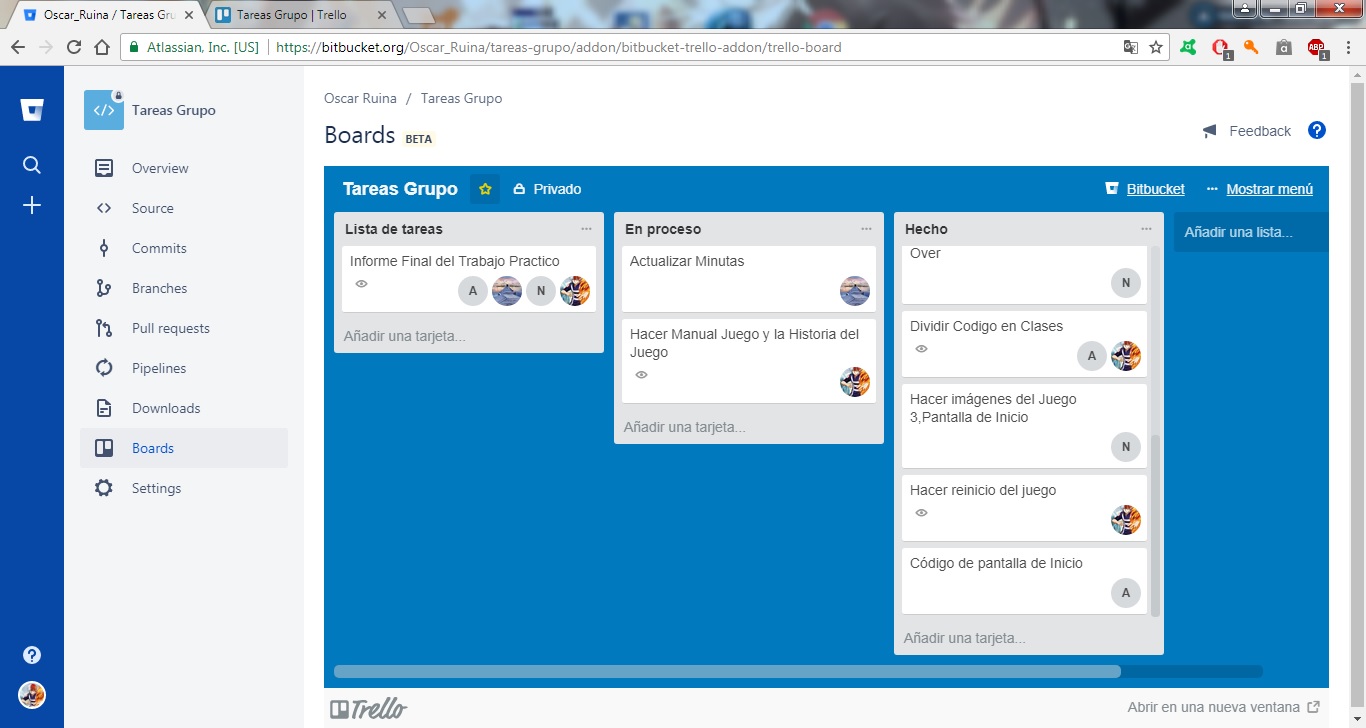
Al tener el juego desarrollado completamente, hemos realizado el manual y la historia del juego.

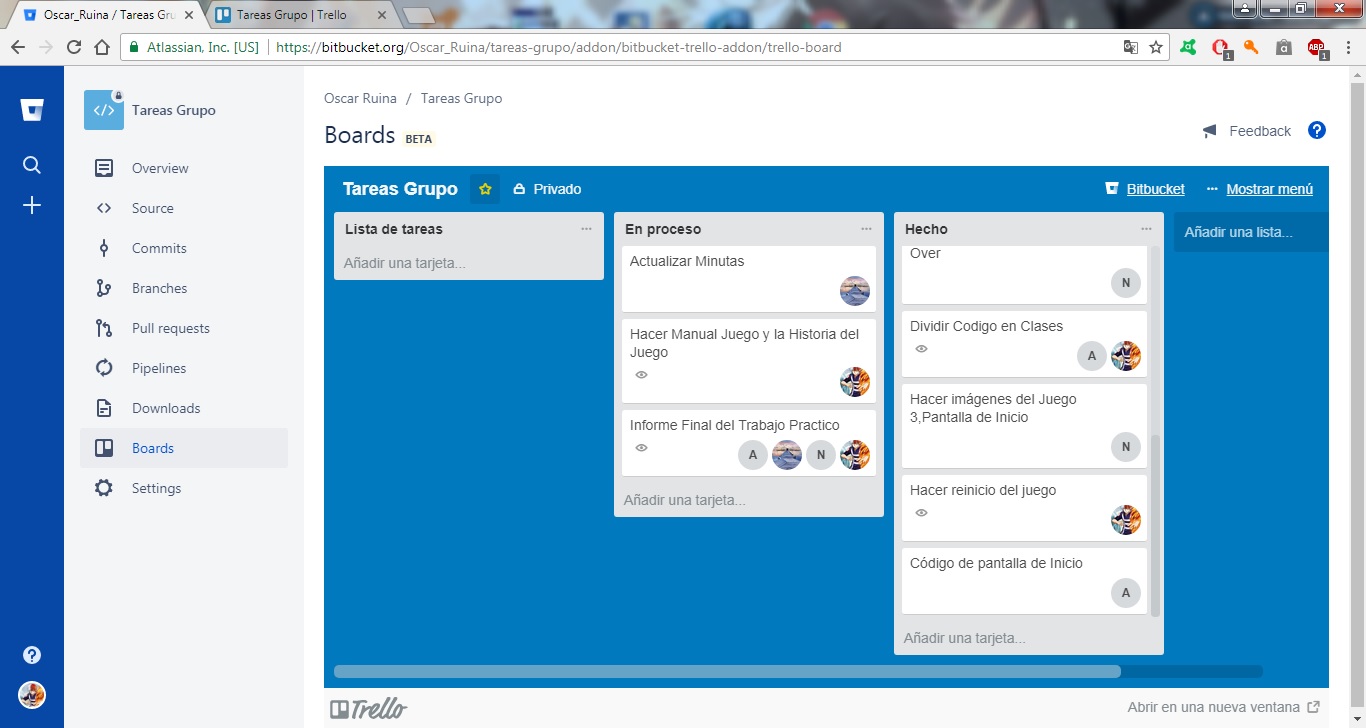
***TÍTULO PROVISORIO DEL PROYECTO***

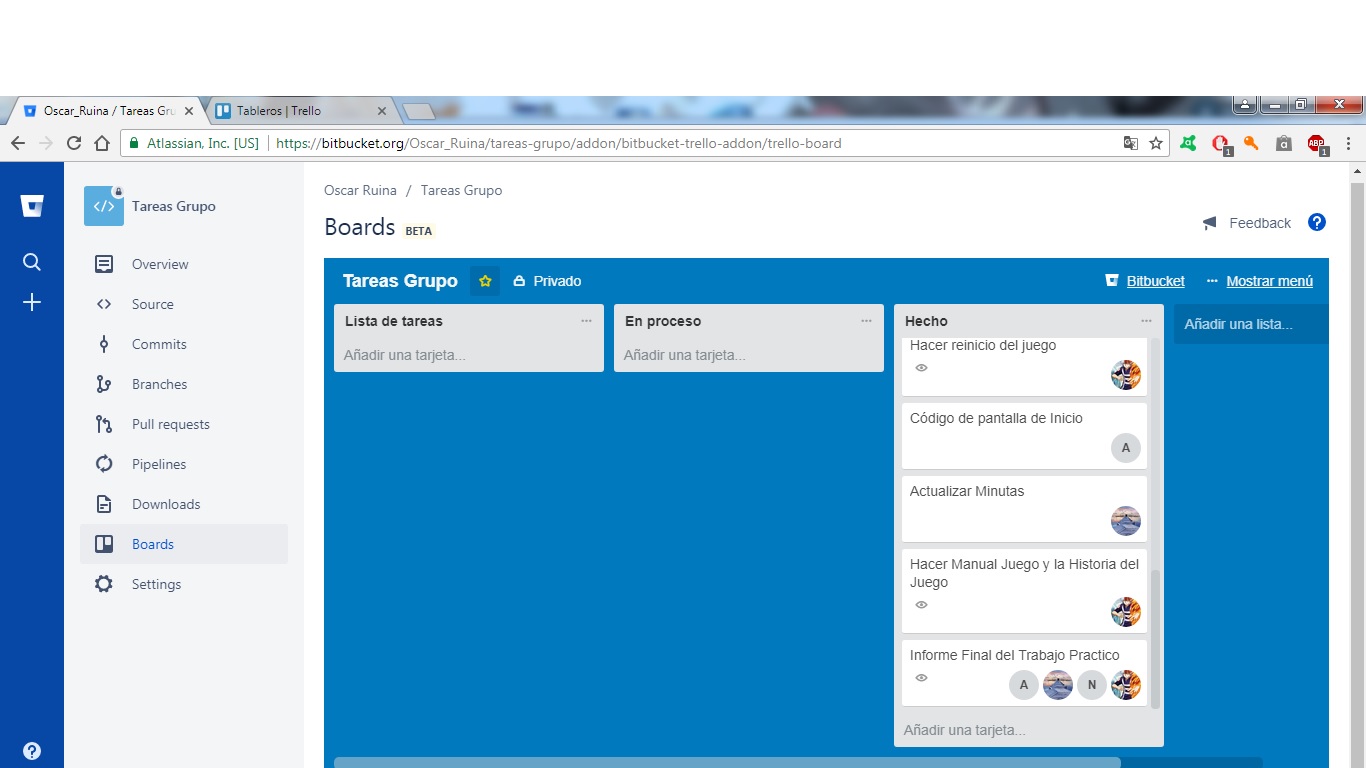
Por decisión del grupo, el nuevo nombre (provisorio hasta el momento) del juego será el de “Humans vs Aliens” (Humanos contra Alienígenas en español). Mientras tanto, se continúa avanzando en el tramo final del proyecto. El código a la fecha está casi terminado, con pequeños detalles a corregir y la documentación comienza a editarse en forma de borrador antes de la edición definitiva que se entregará junto con el programa finalizado en los tiempos de entrega estipulados por los profesores de la materia.

***TÍTULO DEFINITIVO DEL PROYECTO***

Por decisión del grupo, el nombre definitivo del proyecto es “Invasión Alienígena”, que es el mismo que el nombre del juego.

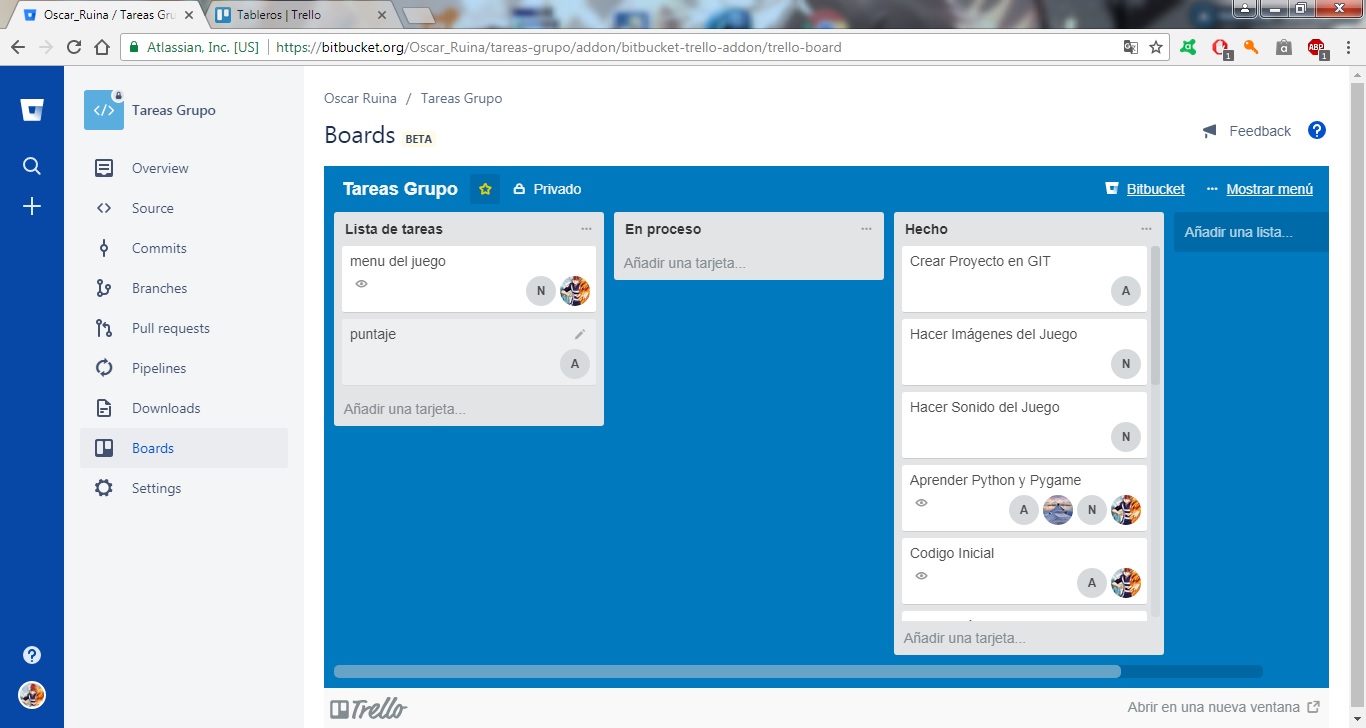


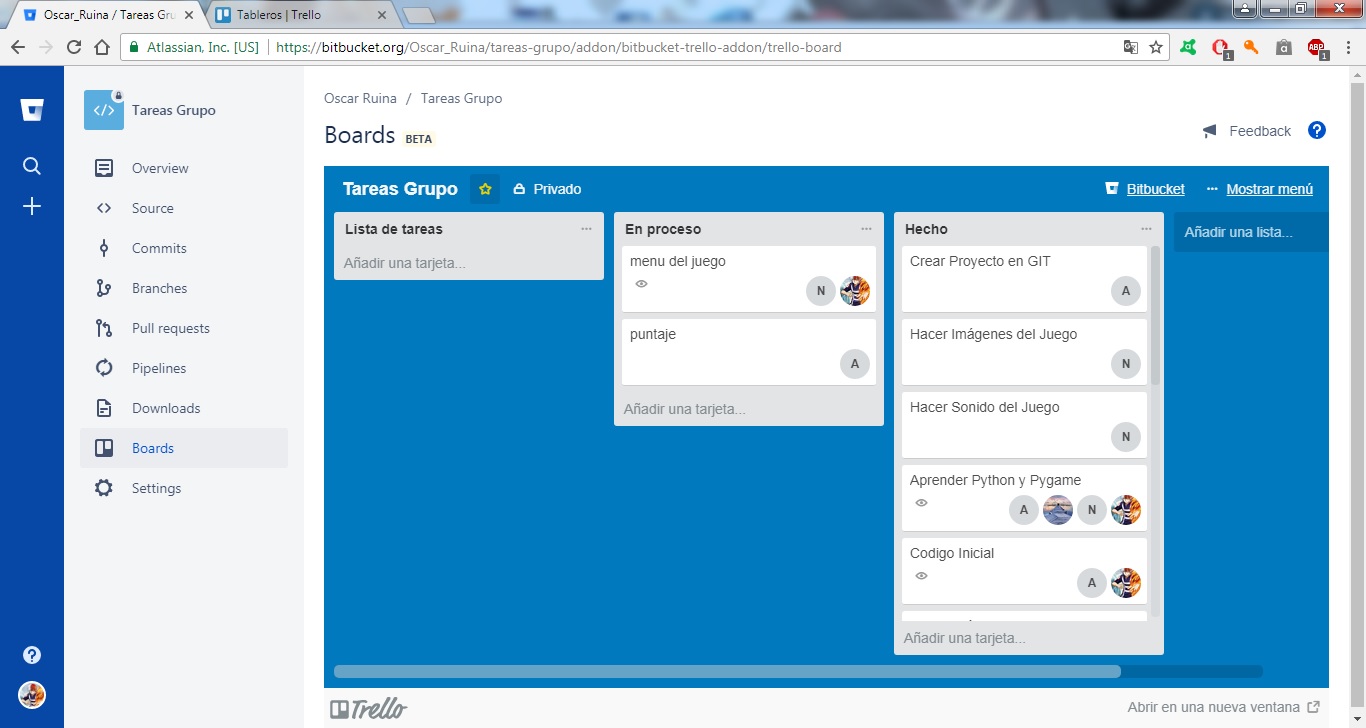
****

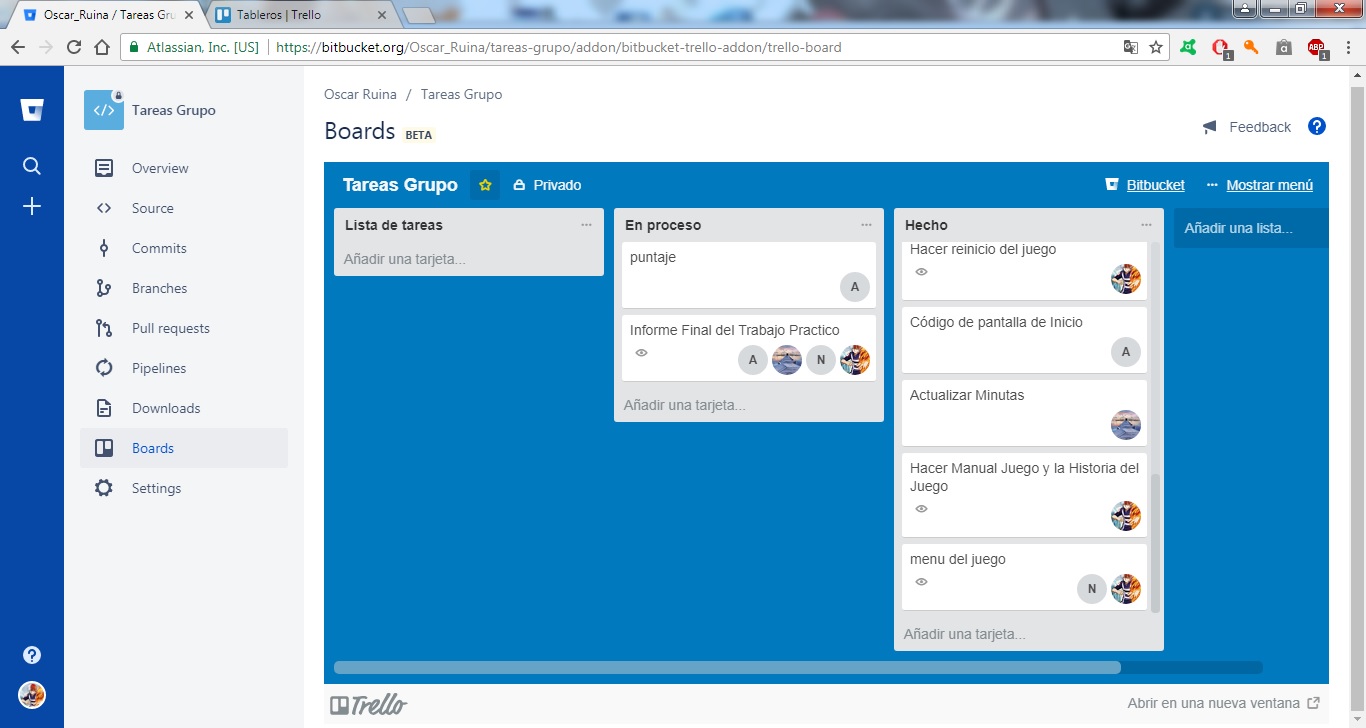


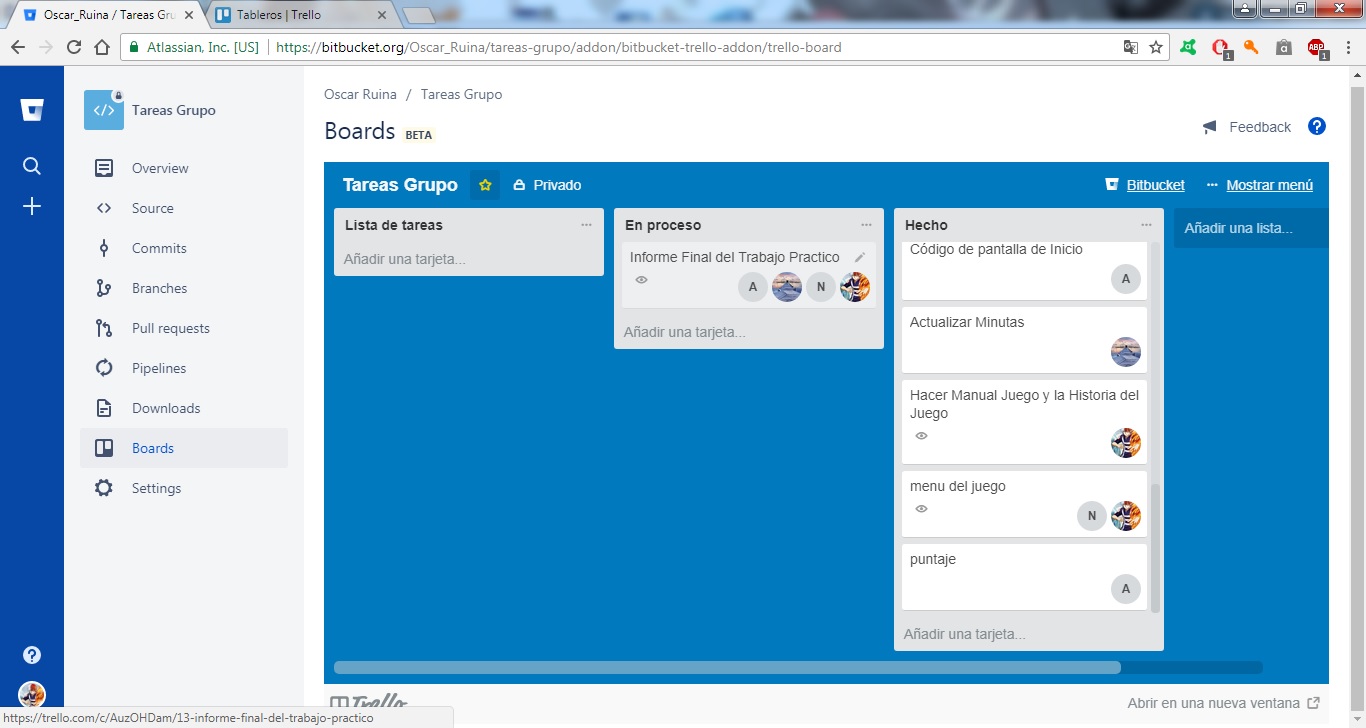
***CORRECCIONES PARA LA ENTREGA DEFINITIVA***

Tras una corrección provisoria del proyecto por parte del profesor de la materia Seminario de Lenguajes, procedemos a realizar las pertinentes modificaciones en el juego “Invasión Alienígena” como así también en el informe final del mismo. Se agregará el pedido menú del programa con su correspondiente imagen de fondo como así también se le incluirá un contador de puntaje acumulado por partida. En el caso del informe del proyecto, se incluirán un diagrama de clases del desarrollo en Python y una explicación sobre el diseño orientado a objetos (UML) elegido para dicho trabajo.









**Historia del juego**

Casi al unísono, luego de que el elevador lunar fuera terminado y la primera operación de reconocimiento fuera de la vía láctea se llevó a cabo, la raza humana deseaba que no estemos solos en el universo.  
Si bien el viaje no pudo conectar con ninguna vida en los ajenos cuerpos estelares, dimos con un planeta con inmensas cantidades de minerales y elementos nunca antes vistos, así comenzó la primera operación de minería extraterrestre en la historia de la humanidad y el nombre Moonsanto se convirtió en un emblema de la conquista a la nueva frontera, el espacio exterior. Pero un cuarto reconocimiento al sistema estelar en que se ubicaba la operación minera detecto una pequeña civilización de apenas 100 habitantes en la cueva de una montaña. El contacto duro poco y los entonces Alienígenas huyeron a una velocidad nunca antes vista. Las noticias estaban siendo furor en la tierra, todos esperaban un re-encuentro con esta pequeña tribu. Pero lo que siguió fue una horda de cientos de naves acercándose a la operación minera. En ese entonces la tierra contaba con tan solo una docena de navíos espaciales equipados con armamento; La ofensa fue brutal, del millón de personas habitando ese lejano planeta que tan orgullosamente conquistamos solo un centenar lograron escapar.

Desde entonces la raza humana trabaja como una para poder derrotar a la inminente amenaza alienígena que sin lugar a dudas se dirige a nuestro planeta.

Mi nombre es Jon Starkiller, capitán del escuadrón rojo operado por NATO. ¿El año? S1XX, y estoy a cargo de defender la base Lativus en el corazón de la Antártida. A mi comando tengo mi fiel X-Wing Starfighter, el último modelo comisionado por un esfuerzo conjunto de E.E.U.U Y la URSS. A la distancia los veo acercarse lentamente a mi posición, mi misión es mantener mi posición hasta que lleguen los refuerzos y esta es mi historia.

**Manual de uso**

Al iniciar el juego aparecerá un menú, que le pedirá al jugador apretar teclas de acuerdo a lo que quiera realizar. TECLA ENTER para empezar a jugar, TECLA M para ver el manual de uso, TECLA D para ver los créditos y TECLA S para salir del juego.

El Jugador cuando inicie el juego controlara una nave espacial y tendrá que destruir a una horda de marcianos enemigas.

El objetivo principal del juego es destruir todos los marcianos enemigos que aparezcan, una vez que el jugador destruya a todos los enemigos ganara el juego y podrá decidir si quiere volver al menú del juego apretando la TECLA C, o ver el puntaje obtenido presionando la TECLA P.

Las teclas para mover la nave y disparar son:

-Movimiento de la nave derecho TECLA FLECHA DERECHA o TECLA D

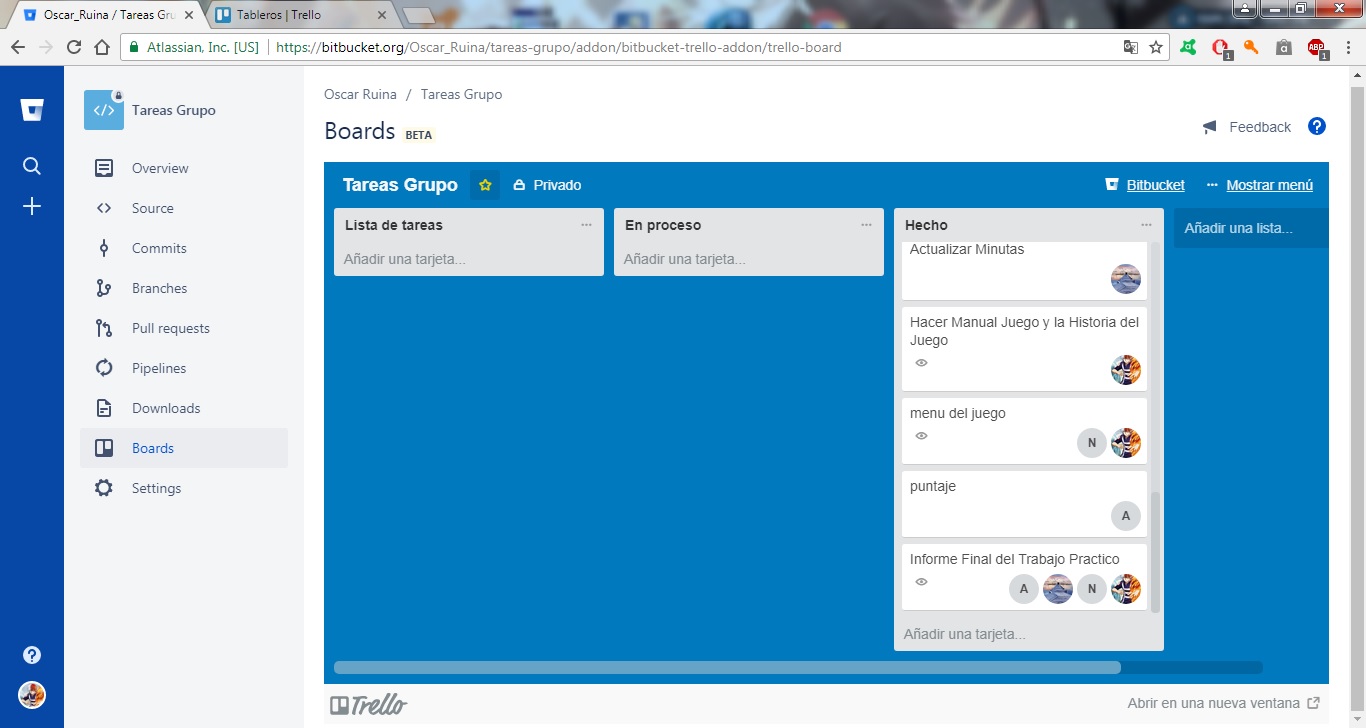
-Movimiento de la nave izquierdo TECLA FLECHA IZQUIERDA o TECLA A

(La nave se mueve lateralmente y para moverse rápido se requiere pulsar repetidamente las teclas)

-Disparar misiles BARRA ESPACIADORA

El jugador deberá prestar atención ya que los marcianos también van a disparar misiles, tendrá que esquivarlos para no perder el juego.

En caso de perder el jugador contara con una nueva posibilidad si aprieta la TECLA C volviendo al menú de inicio, o ver el puntaje obtenido pulsando la TECLA P.

****

Por último, con la documentación terminada y actualizada hemos realizado el presente informe final.

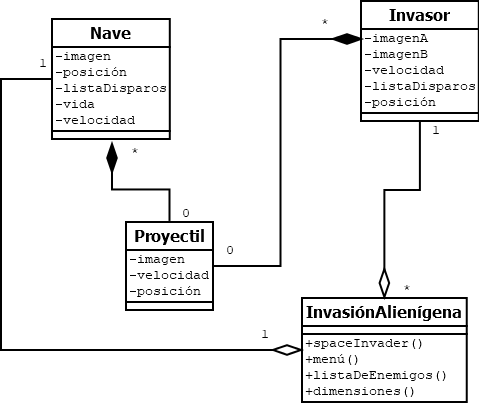
**Documentación técnica**

***Diseño de Sistema***

El presente programa se encuentra desarrollado en el lenguaje Python junto con la librería Pygame diseñada específicamente para videojuegos. Se utilizó la Programación Orientada a Objetos (POO), dividiendo el desarrollo en cuatro clases mediante relaciones de composición como de agregación.

Se realizó la estructura del menú de forma que sea lo más clara y sencilla posible para la fácil comprensión y uso por parte de los futuros usuarios, manejable por medio del teclado (cuyos accesos están detallados en el manual de uso, presente más adelante). En cuanto a las partidas, se trata de un estilo arcade donde se disparan proyectiles a las naves alienígenas enemigas que progresivamente van bajando hacia donde está la nave bajo una lluvia de proyectiles que deberán eliminarse junto con ellas por medio de disparos que irán acumulando puntos cada vez que una nave enemiga sea eliminada.

***Diagrama de Clases***

****

***Explicación de la Programación Orientada a Objetos (POO)***

La Programación Orientada a Objetos comprende la división del código fuente en clases, relacionadas entre sí, para facilitar las pruebas y el posterior mantenimiento y modificación en caso de errores, bugs o futuras funciones nuevas que se apliquen sobre el proyecto. Bajo el lenguaje de diseño UML es posible graficar esta división por clases para tener una rápida y clara visión de cada uno de sus atributos y sus métodos o casos de uso por los cuales se ejecutan todas las acciones que atañen al programa.

**Requerimientos del Sistema**

Los siguientes son los requerimientos mínimos para la ejecución del programa en computadoras.

* **Sistema Operativo:** Windows 7 o posterior
* **Memoria RAM:** 2 GB o mayor
* **Espacio en Disco Duro disponible:** 1 GB o mayor
* **Procesador:** Intel Atom N450 1,67 GHz o posterior
* **Resolución de pantalla:** 1024 x 600 píxeles o mayor

**Conclusiones**

Debido al escaso conocimiento en Python y sobretodo en la librería Pygame, sentimos que con los conocimientos adecuados hubiéramos podido haber hecho aún más de lo ya realizado. De todas maneras, por el nivel del grupo nos sentimos conformes con el resultado alcanzado hasta el momento.

**Futuras líneas de investigación**

> Agregar niveles de mayor dificultad.

> Mejorar la tabla de puntuaciones.

**Referencias**

* Google Argentina – http://www.google.com.ar/
* Pygame - https://www.pygame.org/news
* Python - https://www.python.org/
* Tutoriales Programación Ya - http://www.tutorialesprogramacionya.com/pythonya/
* Atlassian (Bitbucket) - https://bitbucket.org/product
* Atlassian (SourceTree) - https://www.sourcetreeapp.com/
* GitHub - https://github.com/
* Trello - https://trello.com/
* YouTube – http://www.youtube.com/
* Blogs de Python:
  + Pythonízame - https://pythoniza.me
  + Mi diario Python - http://www.pythondiario.com/
  + Python en Español - http://python-esp.blogspot.com.ar/
  + Codecademy (Python) - https://www.codecademy.com/es/tracks/python-traduccion-al-espanol-america-latina-clone-1

**Agradecimientos**

A Google por las imágenes provisorias.

A la alumna Micaela Peralta por las recomendaciones acerca de Bitbucket, SourceTree y Trello.

Al profesor Hernán Merlino por las clases sobre GitHub y metodologías ágiles.