

Proyecto Entornos. Sistema de Reportes del foro de Riot Games.

Hecho por:

- Ángel Granados Monto
- Emilio Hidalgo Barcia

1º Trimestre:

A. DESCRIPCIÓN DETALLADA DE LA EMPRESA:

-. **Nombre de la empresa:** Riot Games Inc.

-. **Actividad empresarial:**

Riot Games es una empresa dedicada a la creación de videojuegos y a la fundamentación de valores que ayude a los jugadores de dichos videojuegos a crear un entorno en la comunidad donde todos tengan las mismas oportunidades para acceder a la educación y desarrollo de habilidades, además de incentivar un comportamiento ciudadano positivo en dichas comunidades.

Este, aunque es su actividad y objetivo principal, también aprovecha las historias creadas para el desarrollo de sus juegos y el cariño que la comunidad le toma a los personajes para crear merchandising que va desde artículos como figuras hasta colaboraciones con grandes marcas como logitech, Louis Vuitton, HyM, SecretLab.... etc. También ha llegado hasta el punto de poder sacar una serie en varias plataformas.

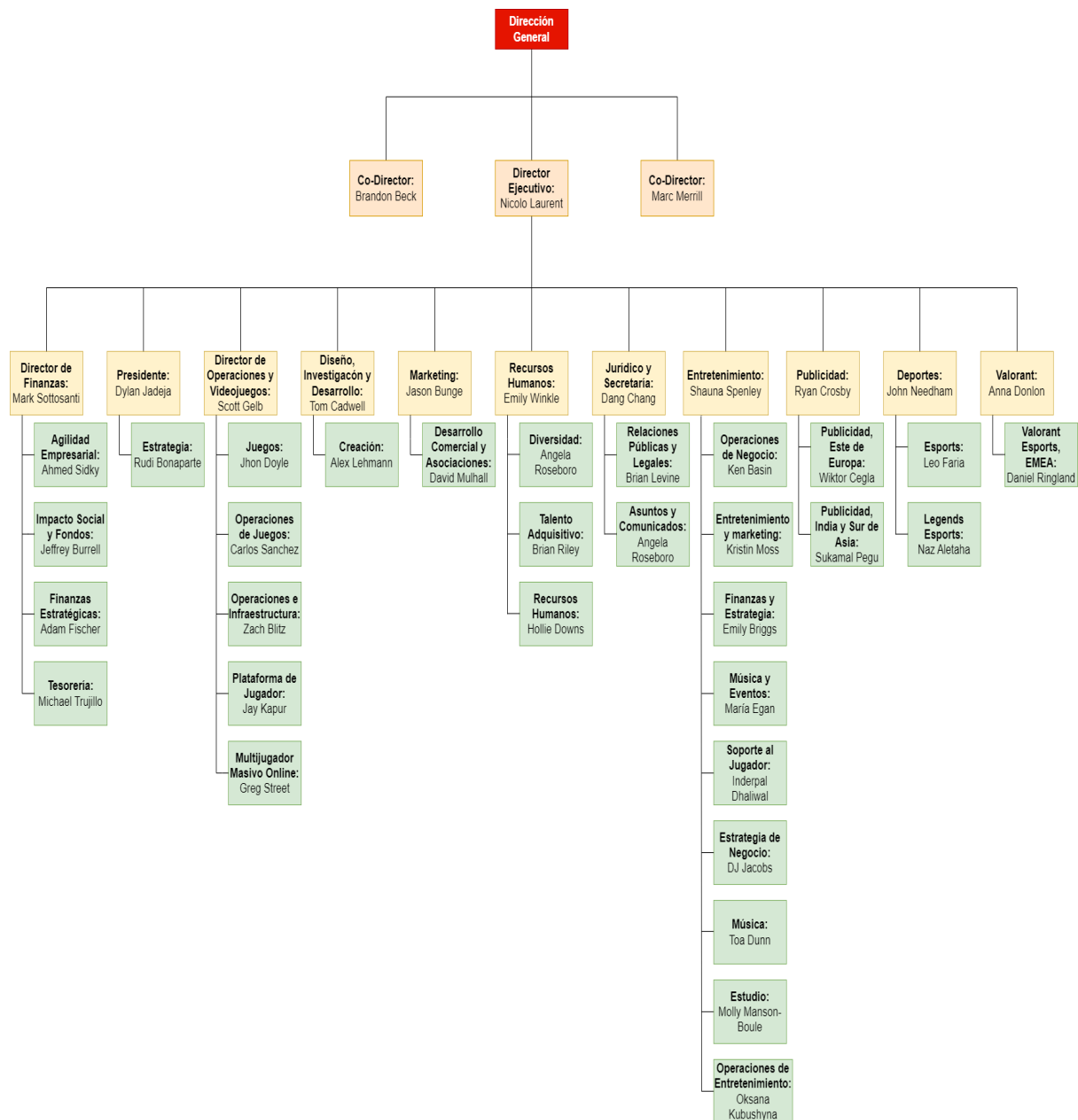
Otro punto fuerte de la actividad de la empresa son las redes sociales que le proporcionan una gran cantidad de ingresos. Gracias a estas Riot Games ha podido organizar una serie de eventos con múltiples fines, entre ellos colaboraciones con varias ONG para ayudar a las personas que más lo necesitan o eventos en colaboración con museos sobre el proyecto de su serie para recaudar dinero para la mantención de este.

-. **Sedes:**

Riot Games fue fundada en Los Ángeles, California, lugar donde sería instalada su primera sede y actualmente la sede principal de la empresa. Hoy en día cuenta con 20 sedes diferentes repartidas por todo el mundo.

País	Ciudad
España	Barcelona
Alemania	Berlín
Emiratos Árabes	Dubái
China	Hong Kong
Turquía	Estambul
Estados Unidos	Los Ángeles
México	Ciudad de México
India	Nueva Delhi
Francia	París
Estados Unidos	Bahía de San Francisco
Brasil	Sao Paulo
Reino Unido	Reading
Irlanda	Dublín
Rusia	Moscú
Corea	Seúl
China	Shanghái
Singapur	Lavender
Singapur	Singapur
Estados Unidos	San Luis
Japón	Tokio

-. Organigrama:



- **Número de empleados y ocupación de estos:**

Actualmente Riot Games cuenta con cerca de 3.000 trabajadores repartidos en sus 20 sedes. Una parte de estos trabajadores realizan teletrabajo por lo cual, en vez de ir a la sede trabajan asociados a la sede de Riot Games más cercana a ellos.

Los puestos de trabajo que podemos encontrar en esta empresa son muy variados así que se agrupan por categorías donde cada uno se encarga de una parte importante dentro de su categoría. Podemos encontrar: Equipo artístico, equipo de balance y desarrollo, equipo de programación, equipo de marketing, equipo de guión y equipo de soporte técnico.

- **Gestión de la información:**

Riot Games actualmente cuenta con una gran cantidad de servidores, ya sea físicos, ubicados en las sedes, o en la nube, donde se guarda la información de sus proyectos, juegos y usuarios que se registran.

Aparte, para ayudar a los jugadores y para almacenar la información cuentan con un proyecto junto a la empresa AWS donde desarrollan inteligencias artificiales que llevan a cabo diversas tareas que ayudan con la gestión dentro de la empresa, las cuales vienen con un software de aprendizaje automático y aprendizaje profundo incorporado que les facilita el poder reconocer problemas de manera más precisa y poder encontrar una rápida y práctica solución a estos mismos.

B. DESCRIPCIÓN DE LOS RECURSOS DISPONIBLES:

Riot Games dispone de un gran presupuesto debido a que es una empresa asociada a la compañía Tencent, la cual cuenta con más de 584.000 millones de euros en sus alcancías. Esto deja mucho juego a Riot Games para poder invertir en equipo de hardware y en programas de software para poder mejorar el rendimiento de la empresa lo máximo posible.

Con lo que respecta a hardware, todos los trabajadores cuentan con un equipo apropiado para su trabajo dependiendo de a que campo dentro de la empresa pertenezcas, y si teletrabajas o trabajas a distancia, la empresa te aporta un equipo adecuado al trabajo que vas a llevar a cabo. Si trabajas para el equipo de arte, normalmente cuentan con una tableta gráfica de la marca Kamvas con un ordenador de sobremesa que lleva equipado una tarjeta gráfica Nvidia, que suele ser de gran potencia debido a los programas y el tiempo de trabajo que van a dedicarle a estos, normalmente de la serie 2000 de Nvidia. Lo mismo pasa con el equipo de desarrollo de juegos, que necesitan testear el motor gráfico de los juegos creados y para ello necesitan un ordenador con una tarjeta gráfica potente.

Para el resto de los trabajadores, suelen usar ordenadores ya sea portátiles o sobremesas, dedicados a la ofimática, ya que el resto de los equipos de Riot Games, se dedican al desarrollo y programación de la estructura de los juegos y software usado dentro de la empresa para poder seguir desarrollando y aumentando la eficiencia de la empresa.

Con respecto a los servidores de esta empresa, estos son grandes servidores que pueden llegar a ocupar 15.000 metros cuadrados los más grandes que se encuentran cerca de las sedes o en estas mismas.

Sobre el software que utiliza y/o desarrolla Riot Games, este suele ser software de programación y diseño gráfico que es empleado para desarrollar los proyectos de videojuegos y ahora en su nuevo proyecto enfocado con la nueva serie de Arcane. El lenguaje en el que programan sus juegos es C++ y C y suelen usar varios tipos de software de programación entre los cuáles se encuentra Visual Studio como uno de los más reconocidos.

Si hablamos del apartado gráfico, la cantidad de software utilizado es incluso mayor, ya que contamos con el uso de varios motores gráficos como es Unity 3d y Unreal Engine, y para solo el modelado y arte que proporciona la empresa en sus proyectos, encontramos varios softwares como Procrate, Clip Studio y Blender como los más significantes. Pero si queremos hablar de lo mejor del software, tenemos que irnos a la inteligencia artificial que desarrolla junto a AWS, que posee softwares de aprendizaje automático y profundo que sobre todo son utilizados para poder arreglar problemas complejos de forma rápida, precisa y de manera automática. El uso en el que más relevancia tiene para el público es a la hora de detectar hackers que intentan robar datos y cuentas o que intentan hacer trampas en los juegos, estas inteligencias detectan rápidamente a estos usuarios y los bloquean permanentemente. Aun así, las IAs desarrolladas se usan en casi todos los campos que abarca la empresa y son un gran aporte para avanzar.

C. DESCRIPCIÓN DE LOS REQUISITOS PARA LA APLICACIÓN:

La empresa Riot Games destaca en la industria gracias a sus videojuegos innovadores y a la gran comunidad de jugadores que juegan diariamente a los videojuegos creados por esta empresa.

Normalmente en los videojuegos competitivos, las grandes cantidades de jugadores de todo tipo ocasionan rivalidad y toxicidad en muchas de sus partidas. Para hacer lo más leve posible la toxicidad y mejorar la experiencia de juego vamos a desarrollar una aplicación que ayude al soporte a detectar a los jugadores que arruinan la experiencia de juego que consistirá un sistema de reporte en el que los propios jugadores tendrán la oportunidad de reportar al jugador especificando a través de una solicitud de baneo, que consistirá en un cuestionario donde el jugador que reporta puede elegir el motivo, especificando los detalles del mal comportamiento del jugador reportado en un mensaje el cual se puede escribir después de especificar el motivo en el cuestionario. El motivo con el mensaje de especificación de reporte llegará al equipo de soporte, el cual se encargará de revisar la solicitud de baneo.

Tras esto el equipo de soporte revisará todas las alertas y, posteriormente, el soporte decidirá si la acción es baneable y la aplicación procederá al baneo de la cuenta.

Requisitos Funcionales de la aplicación:

Reconocimiento del jugador que está ocasionando la mala experiencia en la partida por alguna de las siguientes razones:

- Palabras Malsonantes como insultos o vejaciones
- Lenguaje que incite al odio como racismo u homofobia
- El uso excesivo de mayúsculas
- Nombres de usuario que resulten ofensivos
- Jugadores que no juegan la partida
- Jugadores que realizan jugadas para fastidiar a su equipo

El jugador que detecta el mal uso del chat de texto o mal comportamiento de juego procede a abrir el formulario de solicitud de baneo.

Selección de motivos para el baneo del jugador en el formulario

Relleno del comentario con las especificaciones de baneo.

Envío de la solicitud al equipo de soporte.

El equipo de soporte recibe y lee la solicitud de baneo enviada por el jugador.

El equipo de soporte revisa que lo escrito en el formulario es verídico.

Toma de decisión del equipo de soporte:

- La solicitud es verídica y cumple con los requisitos de baneo.
- La solicitud no cumple con los requisitos para que se proceda al baneo de una cuenta.

Requisitos no funcionales:

El sistema almacena en una base de datos cada vez que una cuenta es baneada.

Cuando la aplicación vaya a banear una cuenta, revisará primero las veces que esta ha sido baneada

La aplicación determinará los días de ban de la cuenta dependiendo de las veces que esta haya sido baneada:

- Si no se ha baneado nunca, se le aplicará un ban de 7 días a esta cuenta
- Si se ha baneado de 1 a 2 veces, se le aplicará un ban de 1 mes

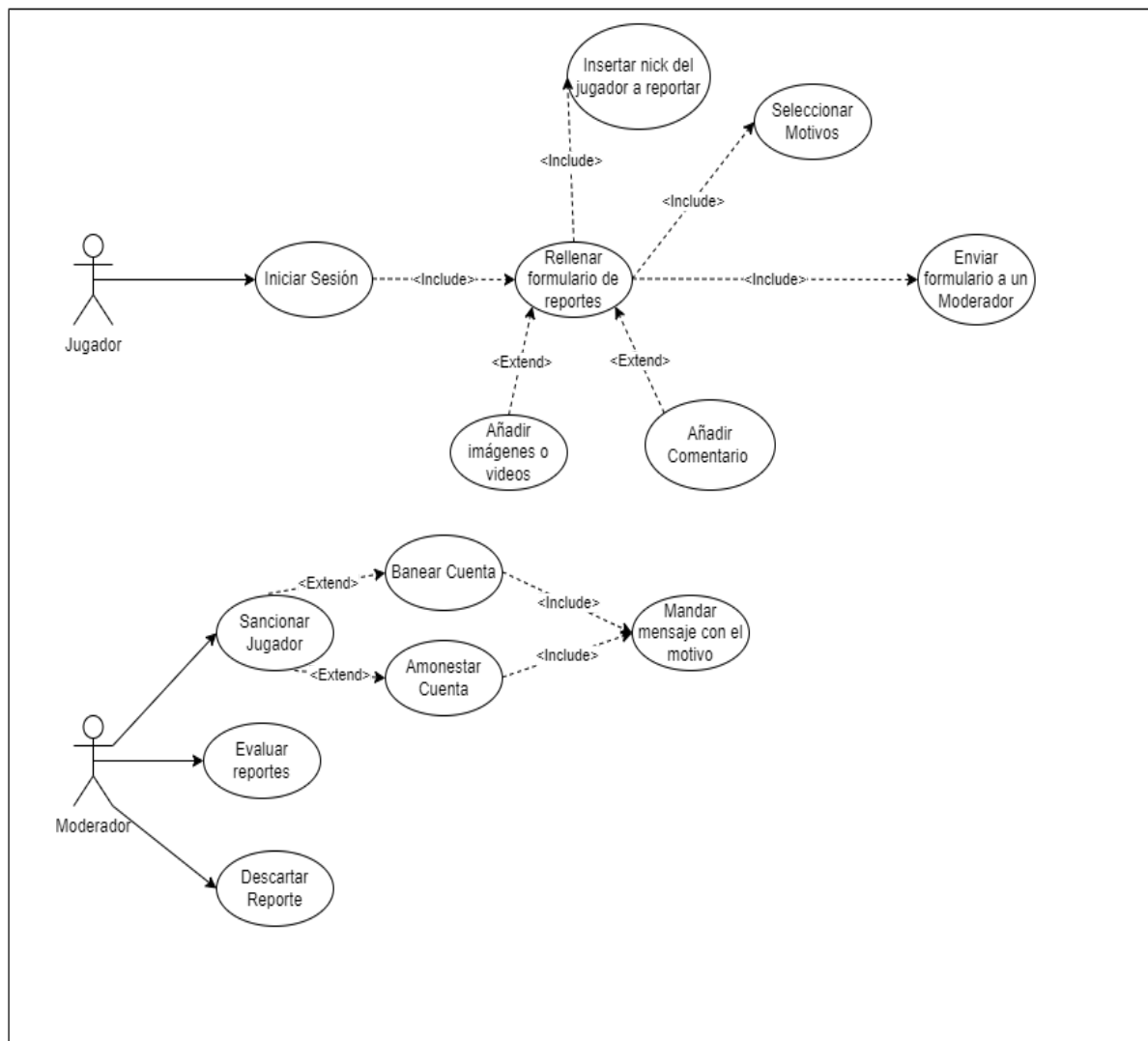
- Si se ha baneado más de 2 veces se procederá a banear permanentemente la cuenta.

Tras el ban de la cuenta la aplicación mandará un e-mail al correo electrónico asociado a la cuenta notificando al usuario que ha sido baneado.

2º Trimestre:

Diagrama de Casos de Uso

FORO DE REPORTE



ACTORES

[USUARIO]

Actor	Usuario	Identificador: A-01
Descripción	Usuario que dentro del foro interactúa con las distintas opciones de reporte para contactar con el moderador	
Características	<ul style="list-style-type: none"> - Necesita estar registrado en el foro - Puede interactuar con las opciones de reporte - Puede contactar con el moderador 	
Relación	Contacta con el moderador a través del sistema de reportes para informar sobre otro usuario	
Referencias	<ul style="list-style-type: none"> - Inicio de sesión - Rellenar formulario - Enviar formulario a moderador - Seleccionar motivos de reporte - Insertar nick de jugador a reportar 	

Atributos	
Nombre	Descripción
Nombre de usuario	Nombre del usuario en el foro
Correo electrónico	Correo vinculado al nombre
Contraseña	Contraseña para acceder a la cuenta del usuario dentro del foro

Comentarios
El Usuario es el que toma acción primero accediendo al foro para poder reportar los malos comportamientos de otros usuarios pudiendo también añadir comentarios y/o vídeos en los reportes para comentar o mostrar hechos adicionales que puedan agilizar y mejorar el trabajo del moderador.

[MODERADOR]

Actor	Moderador	Identificador: A-02
Descripción	Regula los distintos reportes dentro del foro sancionando o amonestando los malos comportamientos reportados	
Características	<ul style="list-style-type: none"> - Necesita estar registrado en el foro - Regula los reportes de los usuarios - Puede sancionar y bloquear a los usuarios 	
Relación	Recibe los reportes de los distintos usuarios sobre los malos comportamientos y sanciona a los usuarios que tengan una actitud contraproducente	
Referencias	<ul style="list-style-type: none"> - Sancionar jugador - Evaluar reporte - Descartar reporte - Sancionar jugador - Mandar mensaje con motivo de la sanción 	

Atributos	
Nombre	Descripción
Nombre de moderador	Nombre del moderador dentro del foro
Correo electrónico	Correo vinculado al nombre del moderador especial para moderar el foro
Contraseña	Contraseña dada por la empresa para acceder a la cuenta del moderador

Comentarios
El moderador recibe en su cuenta dentro del foro los reportes de los jugadores y él es el que tiene juicio sobre el comportamiento de los usuarios y decide qué sanción corresponde a qué comportamiento basándose en las reglas proporcionadas por la empresa

DESCRIPCIÓN DE LOS CASOS DE USO

NOMBRE	FUNCIÓN		
Precondición	Estar registrado		
Descripción	Iniciar Sesión		
Secuencia normal	Paso	Acción	
	1	El usuario abre la interfaz de cliente	
	2	Pincha en iniciar sesión	
	3	Introduce usuario y contraseña	
	4	Se validan	
Postcondición	Entra en la interfaz del juego		
Excepciones	Paso	Acción	
		Explicación de la excepción del paso	
	5	E.1	Introduce mal la contraseña. Se vuelve a pedir usuario y contraseña
	6	E.2	Introduce mal el usuario. Se vuelve a pedir usuario y contraseña
	7	E.3	Introduce mal 3 veces los datos. Se bloquea la cuenta por 24h.

NOMBRE	FUNCIÓN		
Precondición	Haber iniciado sesión		
Descripción	Rellenar formulario de reportes		
Secuencia normal	Paso	Acción	
	1	Inserta el nick del jugador que quiere reportar	
	2	Seleccionar motivos del reporte	
	3	Puede añadir un comentario	
	4	Puede añadir imágenes	
	5	Envía la solicitud de ban	
Excepciones	Paso	Acción	
		<i>Explicación de la excepción del paso</i>	
	5	E.1	Introduce mal el nick. Se vuelve a pedir el nick.

NOMBRE	FUNCIÓN		
Precondición	El jugador ha realizado una acción que empeora la experiencia de juego de otros jugadores		
Descripción	Evaluar reportes		
Secuencia normal	Paso	Acción	
	1	Revisión de los motivos de reporte	
	2	Revisión de las imágenes o comentario	
	3	Revisión de la acción cometida por el jugador que ha sido reportado	
	4	El moderador toma una decisión	
Excepciones	Paso	Acción	
		<i>Explicación de la excepción del paso</i>	
	5	E.1	Si la acción no es baneable el moderador rechaza la solicitud de ban
	6	E.2	Si la acción es baneable el moderador acepta la solicitud

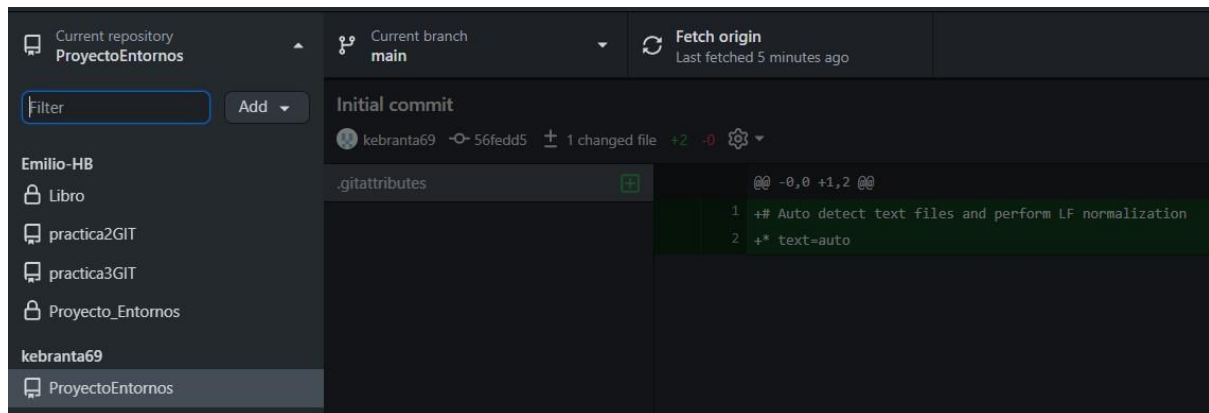
NOMBRE	FUNCIÓN		
Precondición	El jugador ha realizado una acción que empeora la experiencia de juego de otros jugadores		
Descripción	Sancionar jugador		
Secuencia normal	Paso	Acción	
	1	El moderador procede a buscar al jugador en la base de datos	
	2	El moderador observa su historial y decide el tiempo de ban	
	3	Banea al jugador el tiempo decidido previamente	
	4	Envía una notificación de ban al jugador	

NOMBRE	FUNCIÓN		
Precondición	El jugador ha realizado una acción que empeora la experiencia de juego de otros jugadores		
Descripción	Descartar reporte		
Secuencia normal	Paso	Acción	
	1	El moderador decide que la acción no es baneable	
	2	Rechaza la solicitud de reporte	
	3	Notifica al jugador que ha enviado la solicitud de que no es acción baneable	

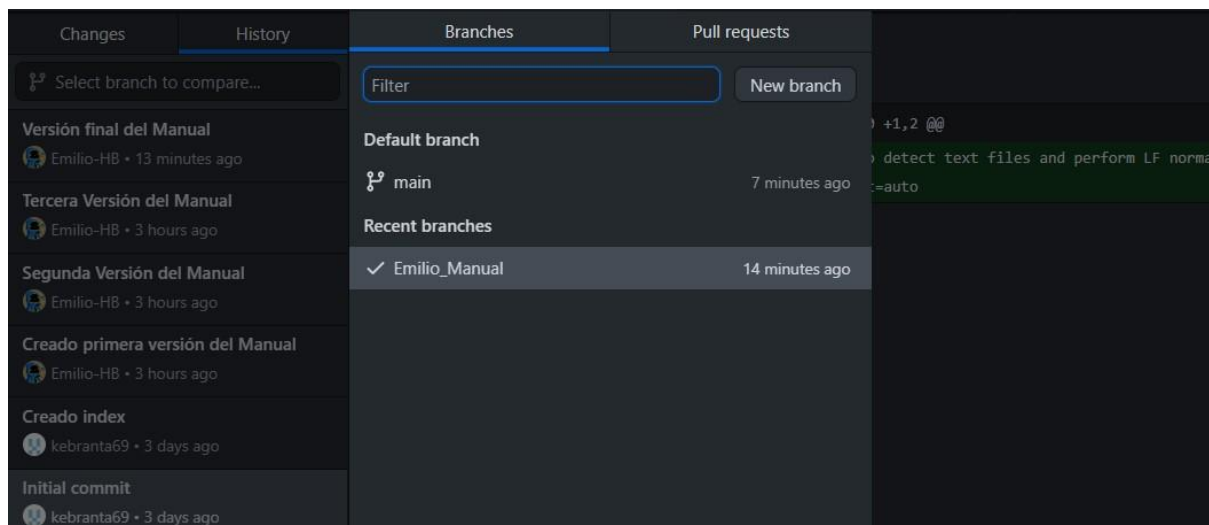
NOMBRE	FUNCIÓN	
Precondición	El moderador debe haber aplicado una amonestación/baneo al jugador	
Descripción	Mandar mensaje con el motivo	
Secuencia normal	Paso	Acción
	1	El moderador evalúa la condición del reporte
	2	El moderador sanciona al usuario
	3	El moderador notifica al usuario sobre dicha sanción con una breve descripción del motivo de la sanción

NOMBRE	FUNCIÓN		
Precondición	Sancionar al jugador/usuario		
Descripción	Banear usuario		
Secuencia normal	Paso	Acción	
	1	El moderador recibe un reporte	
	2	El moderador evalúa la condición del reporte	
	3	El moderador considera un comportamiento de gravedad	
	4	El moderador banea(Bloqueo permanente) al usuario	
Excepciones	Paso	Acción	
	5	E.1	Si más tarde se evalúa que fue un baneo injusto, se le devolverá al jugador el acceso

- Creamos el repositorio ProyectoEntornos dónde vamos a trabajar con el prototipo de nuestra página.



- Añadimos la rama Emilio_Manual dónde Emilio va a trabajar en el manual de soporte mientras que Ángel trabajará en la rama main.

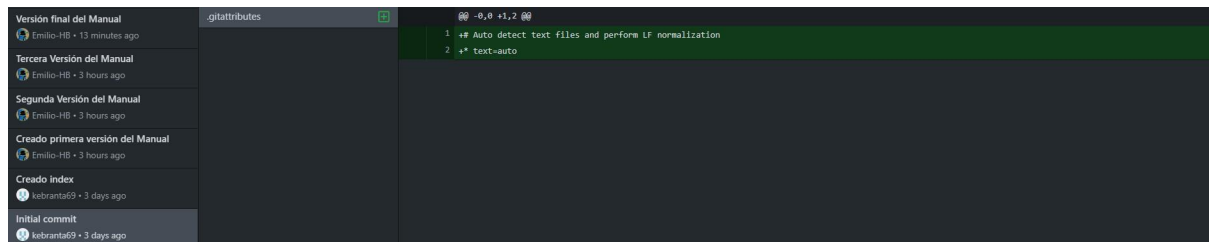


- En las siguientes capturas se muestra el avance en el trabajo en ambas ramas:

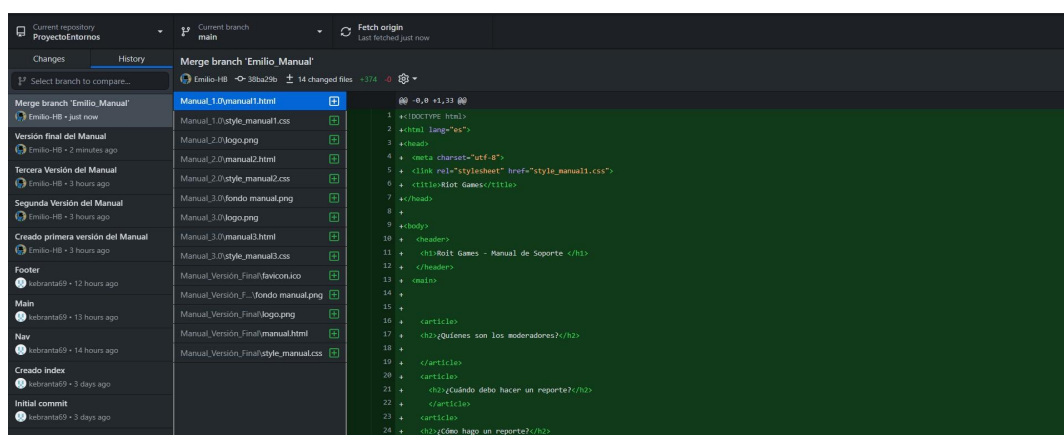
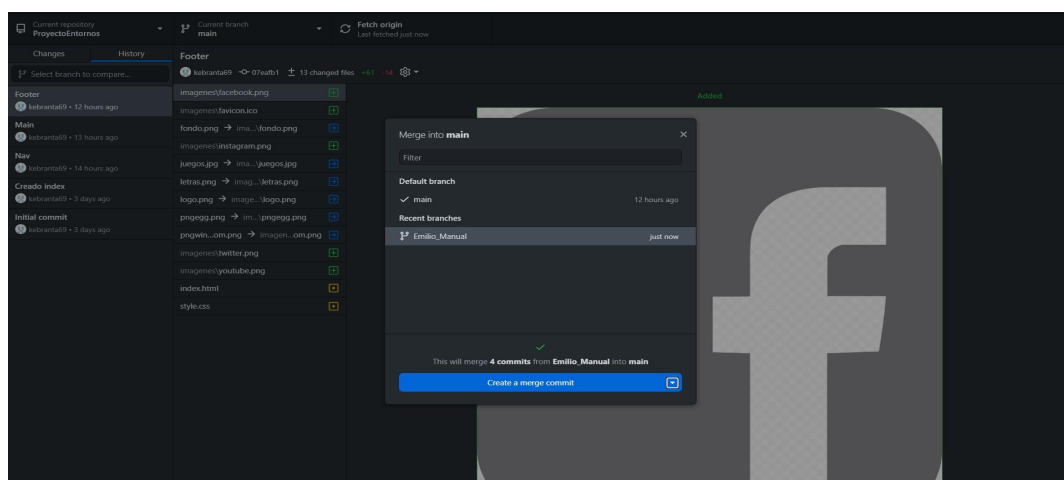
Ángel:



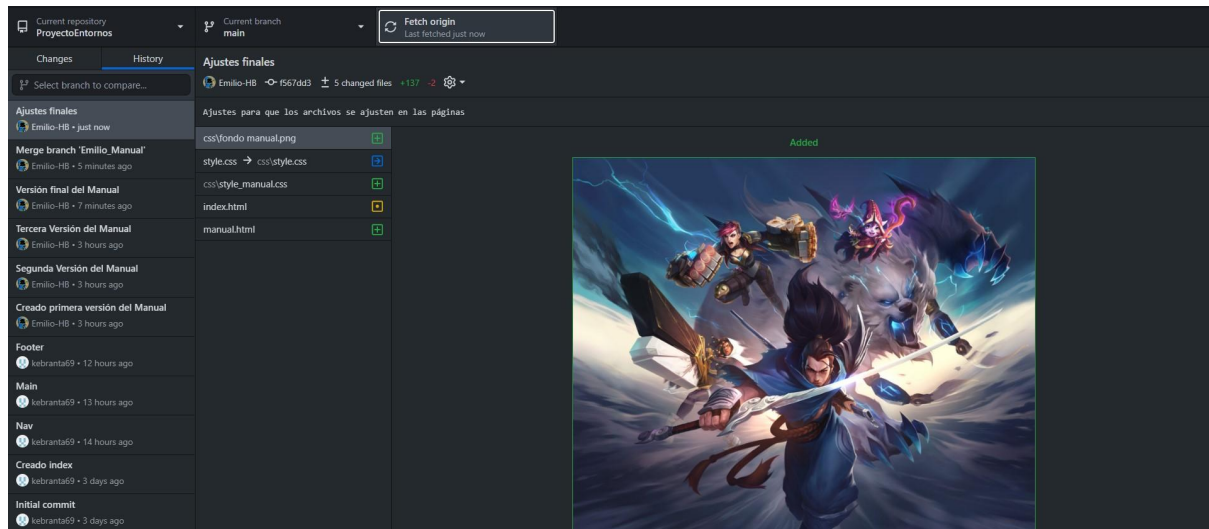
Emilio:



-. Una vez tenemos las dos ramas en sus versiones finales, realizamos una fusión de ramas para agrupar el proyecto en una versión definitiva:



-. Por último, corregimos los errores que puedan hacer que el prototipo no sea funcional y ajustamos los archivos para poder visualizar la página:



-.Este será la versión definitiva del prototipo.

<https://github.com/kebranta69/ProyectoEntornos.git>

El seguimiento del repositorio está dentro del mismo junto a toda la página, si no carga el css, descomprime el archivo, es el único fallo que he encontrado.

3º Trimestre:

- Posteriormente hemos seguido avanzando en el proyecto y hemos iniciado con la implementación de java y el funcionamiento de la caja negra y la caja blanca:

- En la caja negra, Ángel se ha encargado de la implementación del método y el trabajo con junit en NetBeans, escogiendo los casos de uso más convenientes:

```
public class Reporte {  
    /**  
     * La clase reporte almacenar todos los datos recogidos del usuario que rellena el formulario de reportes.  
     * Declaramos las variables necesarias para almacenar cada información proporcionada por el usuario:  
     * Variable nickReportado de tipo String que almacenar el nick del usuario que va a ser reportado.  
     * Creamos un ArrayList de tipo String que almacenar todos los motivos por los que el usuario considera que el otro jugador debe ser reportado  
     * Variable comentario de tipo String que almacenara el comentario que puede proporcionar el usuario.  
     * Creamos otro ArrayList para almacenar las fotos que puede aportar el usuario como pruebas.  
     */  
    private String nickReportado;  
    private ArrayList<String> motivos = new ArrayList<String>();  
    private String comentario;  
    private ArrayList<BufferedImage> imagenes = new ArrayList<BufferedImage>();  
  
    public String getNickReportado() {  
        return nickReportado;  
    }  
  
    public void setNickReportado(String nickReportado) {  
        this.nickReportado = nickReportado;  
    }  
  
    public ArrayList<String> getMotivos() {  
        return motivos;  
    }  
  
    public void setMotivos(ArrayList<String> motivos) {  
        this.motivos = motivos;  
    }  
  
    public String getComentario() {  
        return comentario;  
    }  
}
```

```

public void setComentario(String comentario) {
    this.comentario = comentario;
}

public ArrayList<BufferedImage> getImagenes() {
    return imagenes;
}

public void setImagenes(ArrayList<BufferedImage> imagenes) {
    this.imagenes = imagenes;
}

/**
 * Creamos el método formularioValido que dará por válido el formulario que tenga el nick del jugador reportado
 * y al menos un motivo para que así cuando el formulario llegue al moderador solo tenga que verificar que el nick existe
 * y que los motivos y las pruebas son suficientes para banear al jugador. Con este método todos los formularios que no contengan
 * nick o motivos para el ban serán excluidos automáticamente.
 * @return
 */
public boolean formularioValido(){
    boolean valido = true;
    if(nickReportado == null){
        valido = false;
    }
    if(motivos == null){
        valido = false;
    }
    return valido;
}

public static void main(String[] args){
    Reporte Freporte = new Reporte();
    System.out.println(Freporte.formularioValido());
}

```

```

public class ReporteTest {

    public ReporteTest() {
    }

    @Test
    public void testFormularioValido() {

        Reporte Freporte = new Reporte();

        Freporte.setNickReportado("Player");

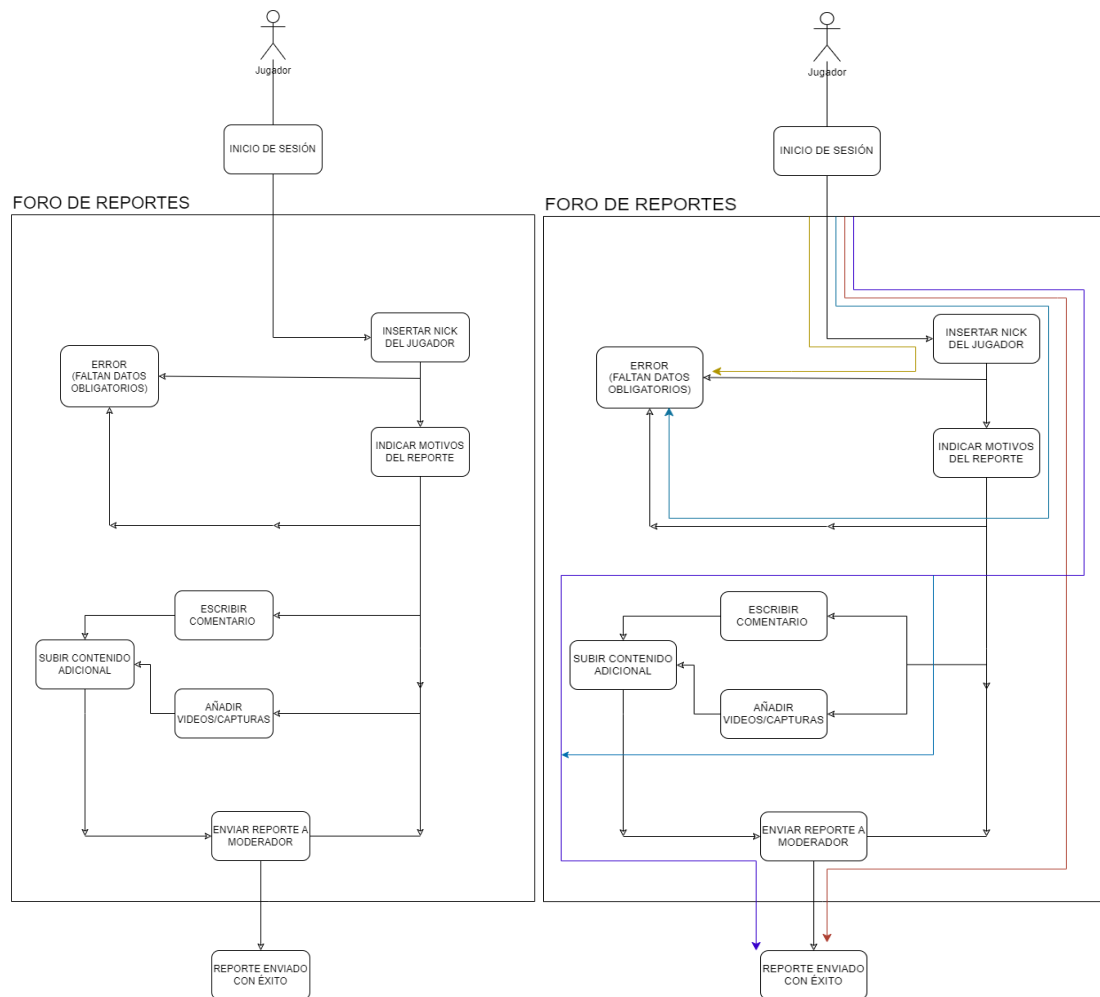
        Freporte.getMotivos().add("Palabras malsonantes");

        Freporte.getMotivos().add("Comentarios racistas");

        Freporte.setComentario("El jugador ha estado insultando durante toda la partida.");
        assertTrue(Freporte.formularioValido());
        System.out.println(Freporte.formularioValido());
    }
}

```

-. Mientras que Emilio se ha hecho cargo del grafo de flujo, los caminos básicos y la caja de pruebas; las tareas de caja blanca, aparte de subir los materiales a github como se especifica:



Name	Last commit message
..	
Manual_1.0	Ejercicio probando,probando...
Manual_2.0	Ejercicio probando,probando...
Manual_3.0	Ejercicio probando,probando...
Manual_Versión_Final	Ejercicio probando,probando...
Probando Probando... 1º Parte	Ejercicio probando,probando...
css	Ejercicio probando,probando...
imagenes	Ejercicio probando,probando...
.gitattributes	Ejercicio probando,probando...
Implementación de la parte 1.java	Ejercicio probando,probando...
SeguimientoRepositorioGithub.pdf	Ejercicio probando,probando...
index.html	Ejercicio probando,probando...
manual.html	Ejercicio probando,probando...
Caja Blanca	Prueba
Caja Negra	Prueba

Con esta última parte, que dentro del enlace de github se puede apreciar con mucho más detalle todos los archivos, queda por terminada la implementación de java dentro de nuestro proyecto y con esto finalizado.