

Instituto Puig Castellar

Proyecto / Credito de sintesi

Documento funcional

ALUMNO/GRUPO: Rubén Castillejos Cañete Izan Hernández Barbens

1. Introducción y contexto

Objetivos: explica de qué trata el proyecto y por que es necesario.

El proyecto consistirá en una página web donde a través de ella podrás crear un servidor de Minecraft. Y te preguntarás, ¿cómo se hará esto posible? Pues esto lo haremos de la siguiente manera: Nosotros tendremos tres máquinas virtuales donde la primera será el servidor de Minecraft, la segunda máquina será donde tendremos nuestra página web que tendrá nuestro logo personalizado, un menú con información básica sobre el juego y pequeños “pro tips” y los diferentes apartados básicos para poder crear un servidor de Minecraft como por ejemplo los mods que le querrás añadir a tu servidor(donde te tendrás que fijar si son compatibles o no) y en qué versión quieres tu servidor. Y por último la tercera máquina será para los backups/copias de seguridad, lo hacemos por si acaso. Sin guardar pasa cualquier cosa y perdemos todo lo que llevamos de proyecto.

Estamos pensando en más cosas que hacer por sí vemos que va todo correctamente, pero, primero tenemos que ver si nuestra idea principal funciona y aun nos queda tiempo para poder mejorar nuestro servidor y que pueda tener como por ejemplo un bot de telegram o por discord o más cosas que se nos pueden ir ocurriendo si vemos que vamos bien de tiempo con el proyecto.

Describe:

- **El problema o necesidad que se quiere resolver.**

Uno de los problemas que de momento tenemos o la duda es si nos va a aguantar nuestro servidor y queremos comprobar si nuestro servidor podrá aguantar los mods que le pondremos.

- **Quién será el usuario o cliente final.**

El cliente final seremos nosotros para comprobar si todo lo que hemos creado funciona a la perfección. También en un futuro si vemos que ha sido un buen trabajo lo que podremos hacer es subir tanto nuestro servidor e intentar que la gente utilice más nuestra página web y creen su servidor de Minecraft para poder jugar.

- **Qué solución se propone y con qué propósito.**

La solución que podemos proponer es ir viendo poco a poco a ver como avanza el proyecto y con el propósito de si podemos realizar tanto un buen servidor, una buena página web y que nos salgan bien las copias de seguridad para no perder nada durante el proceso.

2. Anàlisis de requisitos

2.1. Requerimientos funcionales (RF)

Què se debe hacer el sistema.

Enumeración de las funciones principales, numeradas como RF1, RF2, etc.

Codigo	Descripción del requisito funcionales
RF1	El sistema permitirá que se registren nuevos usuarios.
RF2	Los usuarios podrán iniciar sesión y gestionar su perfil.
RF3	El sistema mostrará las reservas activas del usuario.
RF4	Los usuarios podrán encender su servidor en la página web sin problema y también apagarlo
RF5	Poner casi todos las versiones de minecraft
RF6	Poner copias de seguridad
RF7	Que los usuarios pueden escoger que mods quieren para su server

RF8	Hacer contenedores para separar las partidas de los usuarios.
-----	---

2.2. Requisitos no funcionales (RNF)

Com s'ha de comportar el sistema.

Inclou aspectes com rendiment, seguretat, compatibilitat o facilitat d'ús.

Codigo	Descripció del requisit no funcional
RNF1	Los servidor de minecraft se puedan gestionar desde el móvil
RNF2	Que las características de los jugadores se puedan ver en un grafana

2.3. Restricciones

Condiciones o limitaciones del proyecto.

- Lenguajes o tecnologías obligatòries:

Herramientas del servidor hosting:

Ansible es para toda la configuración del server

Herramientas para el servidor web: html, css, mysql

- Recursos disponibles (tiempo, equipo, materiales):

Tres máquinas virtuales una para el servidor de minecraft, para la web y las copias de seguridad

El equipo es un ordenador de 32 RAM del instituto

- Dependencias o limitaciones técnicas:

Potencia para los servidores

3. Anàlisis de usuarios y roles

Objetivos: Identificar quién utilizará el sistema y qué podrán hacer.

Describe los diferentes tipos de usuarios, sus necesidades y sus permisos.

Rol	Descripciones	Permisos principales
-----	---------------	----------------------

Administrador	Gestionar usuario, hosting, copias de seguridad.	Puede dar de alta, baja, modificar el hosting y hacer copias de seguridad en resumen todos los premios y accesos a todo lo posible.
Usuarios	Utiliza la página web para jugar a los servidores	Podrán crear su propio servidor, reiniciar y apagar poner mods y versión del juego.
Visitante	Recuperar información sin interactuar con el sistema.	Sin permiso

4. Caso de uso / Escenarios de uso

Objetivo: mostrar cómo interactúan los usuarios en el sistema.

Selecciona de tres a cinco casos principales y describe los brevemente.

Codigo	Nombre del caso de uso	Actor principal	Descripción	Resultado esperado
CU1	Un registro nuevo	Nuevo usuario	Primero de todo, podrás acceder sin registrarte, pero, si quieres tener noticias y estar al tanto, crear servidores y poder ayudarnos a mejorar, tendrás que registrarte poniendo tus datos como tu correo, tu usuario que te pondrás y una contraseña	Una vez has hecho esos pasos, ya estarás registrado en nuestra página web.
CU2	Se inicia sesión	Usuario	Si ya tienes cuenta con nosotros, solamente tendrás que poner tu nombre de usuario/correo y tu contraseña.	Listo para navegar por nuestra página web y poder crear los servidores que quieras. O también puedes poner una recomendación para mejorar.
CU3	Crear servidor	Usuarios	Habrás que seleccionar en qué versión querrás crear el servidor. Una vez escoges la versión empiezas a añadir los mods que desees tener en tu server, por último miramos si son compatibles los mods entre si y solo quedará encender el sever.	Se creará tu servidor con lo que has escogido y si el pc aguanta el servidor podrás jugar tranquilamente a tu gusto, ya que podrás

				encenderlo y apagarlo tu.
CU4	Compatibilidad mods	Servidor	Antes de encender el servidor tendremos que mirar si los mods son compatibles entre ellos. Si son compatibles no saldrá ningún error y se podrá encender el servidor sin problemas, si en cambio, hay alguno de los mods que no son compatibles saltara un error diciendo que no son compatibles y tendrás que eliminar el mod o los mods que no son compatibles.	El resultado esperado es que se encenderá el servidor y funcionará todo correctamente.

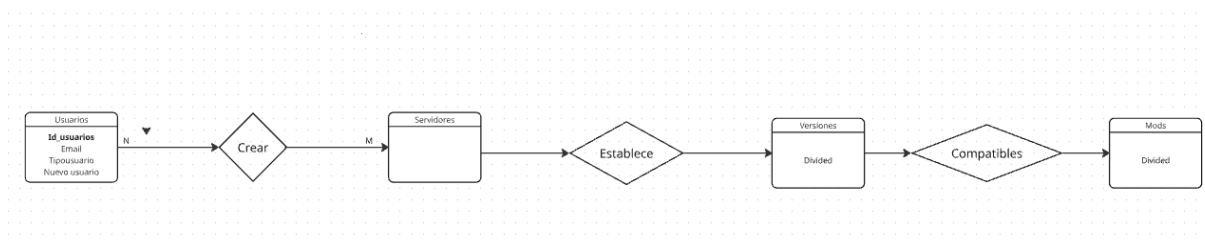
5. Modelo de datos o estructura de la información

Objetivo: representar la información que gestionará el sistema.

Incluye las entidades principales (tablas u objetos) y las relaciones entre ellas.

Ejemplo:

- **Entidades:** Usuario, Reserva, Recurso.
- **Relaciones:** un usuario puede tener varias reservas, cada reserva corresponde a un recurso.

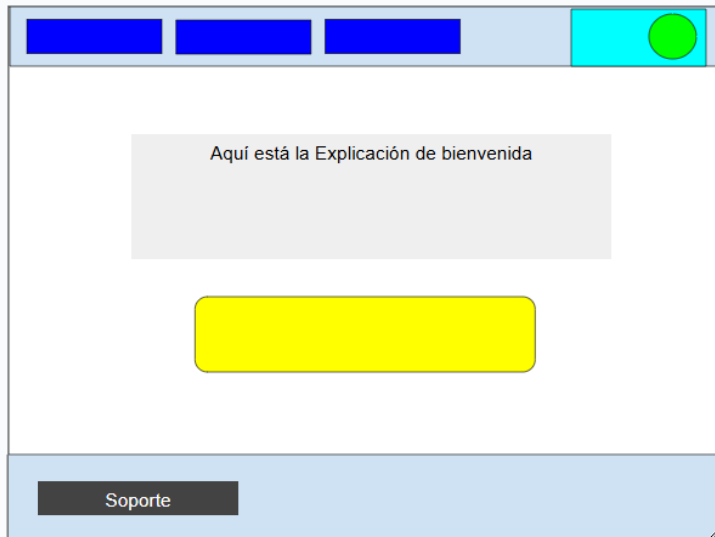


6. Diseño de la interfície

Objetivos: Visualiza la estructura y la navegación del sistema antes de desarrollarlo. Incluye bocetos o capturas de pantalla de las pantallas principales y una breve descripción de su función.

Para cada pantalla, indique:

- Nombre de la pantalla.



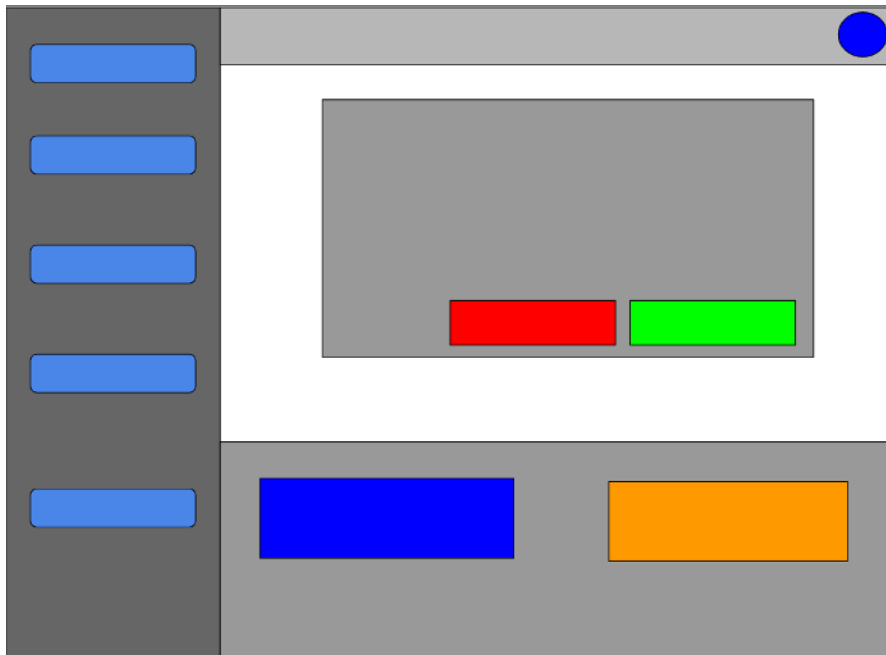
Diseño de inicio: La página contará con un mensaje de bienvenida que explique su propósito y funcionamiento. Los usuarios registrados dispondrán de privilegios adicionales, incluyendo acceso completo a todas las funciones: creación de servidores, elección de versiones, instalación de mods y uso del soporte para resolver dudas, reportar fallos o proponer mejoras. En cambio, los usuarios invitados (no registrados) sólo podrán visualizar el contenido básico de la página, ya que el resto de funciones permanecerán bloqueadas.

Este diagrama muestra un formulario de registro con un fondo gris y bordes redondeados. Incluye cuatro campos de texto verticales: 'Tu Nombre de usuario', 'Tu correo electrónico', 'La contraseña' y 'Repite la contraseña'. En la parte inferior hay dos botones: 'Cancelar' a la izquierda y 'Todo listo' a la derecha.

Registro de cuenta: Este pequeño "formulario" les saldrá a los usuarios cuando le den a la opción de registrarse. Primero de todo habrá que poner el nombre de usuario que te querrás poner, el correo electrónico el cual te vas a registrar y te

lleguen actualizaciones de nuestra página, la contraseña para poder iniciar sesión y ponerla de nuevo para confirmar que contraseña usarás. Una vez que hayas contestado todo se te dará a listo y eso significa que ya estás registrado.

- Funcionalitat principal.
Se usará para poder registrarte en nuestra página web y poder navegar por nuestra introducción con total normalidad.
- Casos d'ús relacionats.



- Nombre de la pantalla.

Diseño de gestión: Una vez registrado en nuestra página, podrás acceder al panel de gestión de tu servidor. Desde allí tendrás la posibilidad de encenderlo o apagarlo, elegir la versión que deseas utilizar e instalar los mods que prefieras. Nuestro objetivo es que todo el proceso sea lo más sencillo e intuitivo posible, de modo que incluso los usuarios con menos experiencia puedan adaptarse rápidamente a las funciones de creación y administración.

- Funcionalidad principal
- Casos de uso relacionados

7. Planificación técnica

Objetivos: Planificar el desarrollo del proyecto.

Indica les tecnologies i eines que s'utilitzaran, i com s'organitzarà la feina.

- **Lenguajes y frameworks:**

Ansible: automatizar la configuración y gestión de servidores, permitiendo desplegar rápidamente nuevas instancias, mantenerlas actualizadas (con versiones de Minecraft, Spigot/Paper), gestionar plugins y configuraciones de forma estandarizada, y orquestar entornos multi-servidor, todo a través de archivos YAML (Playbooks).

- **Base de datos:**

Mysql: almacenar y sincronizar eficientemente grandes cantidades de datos de jugadores y plugins (inventarios, experiencia, registros, etc.) entre diferentes servidores, mejorando el rendimiento y permitiendo funcionalidades avanzadas en redes de servidores (networks), como compartir inventarios o economías, lo que reemplaza el almacenamiento local más lento por una base de datos centralizada y organizada.

Phpmyadmin: phpMyAdmin es una herramienta web que sirve para administrar las bases de datos (generalmente MySQL) que usan plugins o sistemas de tu servidor de Minecraft, permitiéndote ver y modificar información de jugadores (inventarios, posiciones, datos de plugins como protección, economía, etc.)

- **Herramientas de diseño o edición:**

WordPress: usa para servidores Minecraft para crear páginas web que se conectan al servidor, permitiendo mostrar el estado del servidor, gestionar su propio servidor de Minecraft donde los jugadores pueden jugar con su amigos poner mods la version de minecraft quieran y procesan las compras al conectarse, e incluso usar el inicio de sesión de Minecraft para acceder a tu web.

Html y Css: Aunque el servidor sea solo para activar mods, uso HTML y CSS para crear un Panel de Control personalizado y una Guía Visual de Instalación; con HTML organizó el listado de mods y sus requisitos para que los jugadores sepan qué descargar, y con CSS diseño una interfaz sencilla e intuitiva que les permite ver el estado del servidor (si está activo o en mantenimiento) sin tener que entrar a la consola técnica, haciendo que la gestión del servidor sea mucho más clara para todos.

- **División de tareas (si se trata de un grupo):**

Cronograma: (se puede incluir un diagrama de Gantt)

Mes 1 Preparación y bases del sistema

- Configurar el hosting donde se ejecutarán los servidores de Minecraft.
- Instalar el servidor web y preparar la base de datos.
- Crear la estructura para almacenar partidas, configuraciones, versiones y mods.
- Diseñar el panel web básico: acceso, usuarios y estructura inicial.
- Probar la conexión entre el panel y la base de datos.

Mes 2 Desarrollo del panel y conexión con los servidores

- Implementar la creación de partidas desde el panel web.
- Añadir opciones para seleccionar versiones del juego y cargar mods.
- Conectar el panel con el hosting para iniciar, apagar y reiniciar los servidores.
- Configurar el sistema que copia mundos, mods y configuraciones a cada servidor.
- Realizar pruebas internas de comunicación entre panel y hosting.

Mes 3 Gestión avanzada y contenedores

- Implementar contenedores o instancias aisladas para cada servidor de Minecraft.
- Configurar límites de recursos (RAM, CPU, puertos) por servidor.
- Mejorar el sistema de gestión de versiones y compatibilidad de mods.
- Añadir visualización de logs desde el panel.
- Optimizar la base de datos para guardar datos de jugadores y partidas.

Mes 4 Pruebas finales, optimización y lanzamiento

- Realizar pruebas completas: creación de partidas, carga de mods, arranque y apagado.
- Corregir errores en contenedores, rutas, permisos y base de datos.
- Ajustar el rendimiento del hosting y del panel web.
- Documentar el sistema para administradores.
- Lanzar oficialmente el servicio.

8. Análisis de riesgos

Objetivo: Identificar posibles problemas y cómo se abordarán.

8.1. Identificación de riesgos

Ejemplos:

- Marca de tiempo o mala planificación.
- Problemas técnicos o incompatibilidades.
Uno de los problemas principales es la potencia del procesador para activar dos servidores a la vez; ya sea porque estén sobrecargados de mods pesados o porque haya muchos jugadores, el CPU puede saturarse al intentar procesar los cálculos de ambos mundos al mismo tiempo. Aunque disponemos de 32 GB de RAM, si le asignamos demasiada memoria a cada servidor, el sistema de limpieza de Java (Garbage Collection) puede causar tirones de lag, por lo que debemos repartir esos 32 GB de forma inteligente para que el procesador y el disco puedan manejar toda la carga sin colapsar.

- Pèrdua de dades.
- Abandono de miembros del grupo.
- Cambios en los requisitos.

8.2. Valoració i resposta

Califique cada riesgo según su probabilidad e impacto, e indique cómo se mitigará.

Riesgo	Probabilidades	Impacto	Plan de prevención o contingencia
Falta de tiempo	Alta	Alta	Dividiremos las tareas y estableceremos los plazos intermedios.
Problemas técnicos	Media	Media	Pruebe las tecnologías antes de programar.
Pérdida de datos	Baja	Alta	Realiza copias de seguridad semanales.
Problemas de potencia	Media	Media	Optimizar los servidores
Conectar el hosting a la web	Alta	Alta	Para garantizar una buena conectividad entre el hosting y el servidor web, realizaremos varias pruebas de conexión para verificar su estabilidad y rendimiento.
Problemas de incompatibilidad de versiones o mods	Media	Alta	Los fallos surgen cuando la versión del servidor y la de los mods no coinciden, basta con usar todos los mods en la misma versión del juego y revisar el latest.log para identificar y corregir el mod que causa el conflicto.
En caso de fallo de la base de datos	Media	Media	En un servidor de Minecraft, los problemas de base de datos suelen hacer que no se guarden bien los datos de los jugadores. Normalmente pasa por una mala configuración o fallos de

			conexión, así que toca revisar credenciales, rutas y los registros para ver qué está fallando.
problemas en los contenedores	Alta	Alta	En los contenedores pueden surgir problemas cuando no están bien configurados, consumen más recursos de los previstos o se quedan sin permisos para acceder a archivos del servidor. Esto puede hacer que los mundos, plugins o procesos de Minecraft no arranquen bien, así que toca revisar límites de memoria, rutas, puertos y los registros del contenedor para ver qué está fallando.

9. Validación y créditos del éxito

Objetivo: defina cómo sabremos que el proyecto está funcionando correctamente.

- **Criterios de aceptación.**

Vamos a tener diferentes criterios de aceptación sobre nuestra página web y relacionado con el inicio a Minecraft y con nuestro servidor de prueba:

Sobre el inicio de sesión y registro

- El usuario puede registrarse con su correo y su contraseña, después no podrá registrarse otra vez con el mismo correo. Una vez te registres te llegara un mensaje a tu correo que pondrá lo siguiente **“Bienvenido/a a nuestra página web, espero que te guste nuestro servicio, que puedas crear todos los servidores que quieras con los mods que veas más guapos para tu server (Cuidado y observa si son compatibles todos los mods) y nos puedes enviar cualquier recomendación para mejorar nuestra página.”**. Y podrás navegar por nuestra página.

- Al iniciar sesión, si pones incorrectamente tu usuario o contraseña te saldrá un mensaje de error.

Introducción

- El apartado de introducción muestra la información correctamente desde cualquier dispositivo y no hay ningún error al verlo desde otro dispositivo como por ejemplo desde el móvil.

Versiones

- En este se muestra toda una lista de versiones disponibles en las cuales se pueda jugar.

- Que el usuario pueda seleccionar la versión que desee jugar.

- La versión que haya sido seleccionada por el usuario queda guardada para poder preparar todo y escoger los mods.

Mods y compatibilidad

- Que se muestre toda una lista de mods.

- Cada mod indica su nombre, la descripción de lo que hace el mod y en que versión es compatible.

- El sistema solo permite seleccionar los mods que son compatibles con la versión elegida.

- Si un mod no es compatible con otro mod, se mostrará un aviso al usuario de que no se podrá iniciar el servidor.

- El usuario puede añadir los mods que quiera en su servidor. Esto evidentemente no se puede repetir.

Iniciar servidor

- El usuario una vez ha seleccionado la versión en la que quiere jugar y los mods que ha elegido son compatibles entre sí, podrá iniciar el servidor y comenzar a jugar.

- El sistema mostrará en qué estado está, es decir, si está activo, si está inactivo o si ha dado un error. En el caso que haya un error se mostrará un mensaje explicando dónde está el error.

Comprobación del funcionamiento del servidor

- Que el servidor se inicie sin errores.

- Una vez se cierre el servidor, salga como activo.

- Que los usuarios se puedan conectar al servidor desde Minecraft, que los mods funcionen correctamente y que el mundo se cargue correctamente.

Rendimiento y carga

- El servidor se mantiene correctamente estable con varios jugadores, es decir, no se producen caídas del servidor constantes.

- **Proves previstos (funcionals, d'usuari, de rendiment).**

Pruebas funcionales:

- Los usuarios se pueden registrar correctamente y pueden iniciar sesión.

- Que se muestren todos los apartados de nuestra página web.

- Puede seleccionar la versión que él quiera para jugar.

- El sistema permite que los usuarios pueden añadir mods y eliminarlos.
- Los usuarios se pueden conectar al servidor.
- El servidor se inicie correctamente sin ningún fallo, con los mods que se han seleccionado y al reiniciar no haya ningún error.

Pruebas de usuario

- Que la web sea usable desde todos los dispositivos.
- La información de los apartados se vea alta y clara.
- Que el usuario sepa que su servidor está activo o encendido.
- Los usuarios pueden configurar el servidor sin ayuda externa.
- Que la navegación entre apartados sea clara.

Pruebas de rendimiento

- Los apartados cambian sin retrasos.
- La selección de las versiones y de mods sea rápida.
- Que la página se cargue rápidamente sin ningún problema.
- El servidor se mantiene estable con varios jugadores conectados al servidor.
- No hay lag excesivo durante el juego.
- El servidor durante los primeros 30 minutos funciona correctamente.

10. Conclusion

Objetivo: concluya el análisis y prepárese para la siguiente fase.

Resuma las decisiones principales:

- **Qué funciones tendrá el sistema.**

El sistema tendrá diferentes funciones como són las siguientes:

Estas son de gestión de usuarios

- Registrar a los nuevos usuarios
- Que permita iniciar sesión
- Que los usuarios puedan cerrar sesión
- Que éste valide las credenciales de cada usuario, es decir, si cada usuario está poniendo su id bien y su contraseña.
- Proteger el acceso a funciones avanzadas como por ejemplo mods y al servidor.

Versiones y mods

- Mostrar las versiones disponibles.
- Permite seleccionar una versión.

- Que el usuario cuando elija la versión se guarde seleccionada.
- Que la versión sea compatible con el servidor.
- Que se muestre toda la lista de mods disponibles.
- Entre los mods haya compatibilidad, evidentemente añadir mods, eliminarlos y que no se pueda poner mods duplicados.

Gestión servidor

- Que pueda iniciar el servidor.
- Que pueda detener el servidor o reiniciar.
- Se muestra el estado del servidor.
- Que muestre mensajes de error y logs básicos.

Seguridad

- Proteger todos los datos de los usuarios.
- Evitar configuraciones incorrectas.
- Copias de seguridad siempre por si acaso.
- **Qué tecnologías se utilizarán.**
 - HTML
 - CSS
 - Ansible
 - PHP
 - Miró: Programa para la creación de la base de datos lógica
 - PHPmyAdmin
- **Qué valor aporta el proyecto.**

Este proyecto nos aporta diferentes cosas como por ejemplo:

 - La creación de una página web (integrarnos más sobre este tema y saber hacer mejores páginas y más profesionales).
 - La creación de servidores de un videojuego.
 - A como gestionar un servidor web.
 - La creación de los contenedores para separar las partidas.

Indique los siguientes pasos.:

- **Prepare el entorno de desarrollo.**

Va a ser una página web donde se va a poder gestionar y crear diferentes servidores del famoso videojuego “**Minecraft**”. Los usuarios podrán escoger una versión con la que puedan jugar, y también, podrán añadir o no añadir mods a su servidor. Esto lo conseguirán hacer en nuestra página saldrán los apartados de versiones y mods. A partir de ahí escogeran lo que ellos quieran pero con una condición, tienen que ser compatibles el servidor con los mods y entre los mods.

Esto lo haremos posible gracias a HTML y CSS que nos ayudará para hacer nuestra página y que quede lo más profesional posible.

- **Cree la estructura inicial del repositorio y la base de datos..**

Nuestra base de datos es la siguiente:



Consta de 4 tablas que son los diferentes apartados o entidades de este proyecto. Tenemos usuarios, servidores, versiones y mods, las cuales usuarios y servidores es una relación N-M, servidores y versiones es 1-N y versiones y mods es 1-N.

- **Comience a implementar los casos de uso prioritarios..**

Los casos de uso prioritarios son los siguientes:

- Registro nuevo
- Inicio de sesión
- Creación de servidor
- Gestión de version y mods
- Compatibilidad de versión y de mods