**COLEGIO SAGRADA FAMILIA – EL PILAR**

|  |  |
| --- | --- |
|  |  |

****

**PROYECTO FIN DE CICLO**

“Writed & Directed By”

|  |  |
| --- | --- |
| **Vº Bº del Director del**  **Proyecto** | **AUTOR: Juan Carlos Fernández González**  **DIRECTOR: José Luis García Bellón**  **CURSO: 2022 / 2023** |

**NOTA MUY IMPORTANTE: Esta plantilla es sólo una guía orientativa. Un PFC no tiene por qué tener ni todas sus secciones ni sólo las que aparecen aquí. La plantilla debe ser adaptada a cada PFC (añadiendo secciones, quitando algunas o modificando los contenidos que aparezcan según cada caso particular), y el alumno debe consultar siempre a su director de PFC ante dudas acerca de cualquier tema.**

**NOTA: Este es un documento en constante evolución y de gran tamaño, por lo que no es raro que se puedan encontrar fallos. Si detectases algún fallo, te pedimos que colabores con los autores de esta plantilla y mejórala, enviando un correo con tus incidencias. También puedes colaborar aportando nuevas ideas, secciones, contenidos o cualquier cosa que creas que le haga falta a este documento para mejorar sus contenidos actuales. Envía un correo a la dirección anterior si crees que puedes tener algo que aportar. GRACIAS.**

**Este documento está pensado para ser impreso a doble cara, cada capítulo comienza en la cara de delante de la hoja.**

**IMPORTANTE:**

**A CONTINUACIÓN SE ESPECIFICA, PARA CADA CAPÍTULO, EL CONTENIDO OBLIGATORIO:**

- TEXTO SIN REMARCAR: TODOS LOS PROYECTOS

- TEXTO REMARCADO EN AMARILLO: SOLO AQUELLOS QUE REALIZAN UN PROGRAMA O DISEÑAN BASES DE DATOS

- TEXTO NARANJA: PROYECTOS RELACIONADOS CON REDES

EL RESTO DE CONTENIDO, NO ESPECIFICADO EN EL ÍNDICE (PORTADA, AGRADECIMIENTO, PÁGINAS EN INGLÉS,..) ES OBLIGATORIO.

Capítulo 1. Memoria del Proyecto 14

1.1 Objetivos y Alcance del Proyecto 14

Capítulo 2. Introducción 15

2.1 Estudio de la Situación Actual 15

2.2 Evaluación de Alternativas 15

2.2.1 Sistema 1 16

2.2.2 Sistema 2 16

Capítulo 3. Aspectos Teóricos 17

3.1 Concepto 1 17

3.2 Concepto 2 17

Capítulo 4. Planificación del Proyecto y Resumen de Presupuestos 19

4.1 Planificación 19

4.2 Resumen del Presupuesto 21

Capítulo 5. Análisis 22

5.1 Definición del Sistema 22

5.2 Requisitos del Sistema 22

5.3 Análisis de Inter5faces de Usuario 23

5.4 Especificación del Plan de Pruebas 25

5.5 Diseño de la Base de Datos 26

5.5.1 Descripción del SGBD Usado 26

5.5.2 Integración del SGBD en Nuestro Sistema 26

5.5.3 Diagrama E-R 26

Capítulo 6. Implementación del Sistema 27

6.1 Estándares y Normas Seguidos 27

6.2 Lenguajes de Programación 27

6.3 Herramientas y Programas Usados para el Desarrollo 27

6.3.1 Programa 1 27

6.3.2 Programa 2 277.4 Manual del Programador 29

6.4 Creación del Sistema 28

6.4.1 Problemas Encontrados y soluciones adoptadas. 28

Capítulo 7. Manuales del Sistema 29

7.1 Manual de Instalación 29

7.2 Manual de Ejecución 29

7.3 Manual de Usuario 29

7.4 Manual del Programador 29

Capítulo 8. Conclusiones y Ampliaciones 31

8.1 Conclusiones 31

8.2 Ampliaciones 31

Capítulo 9. Referencias Bibliográficas 32

9.1 Libros y Artículos 32

9.2 Referencias en Internet 32

Capítulo 10. Apéndices 35

**10.1 Contenido Entregado en el pendrive. 35**

**BORRAR ESTAS NOTAS EN LA DOCUMENTACIÓN FINAL**

Agradecimientos

Esta sección no es en absoluto obligatoria, pero es el lugar correcto para dedicar el proyecto a las personas/instituciones/empresas/… que se desee.

Resumen

El proyecto “Writed & Directed By” (W&DB) es una iniciativa para la creación de un repositorio multimedia en red con el cual poder acceder de manera remota a una colección personal de películas, con el objetivo de poder acercar el 7º Arte a todo aquél que quiera disfrutarlo. “W&DB” permite a quien cuente con una conexión de internet la posibilidad de crearse una cuenta gratuita y poder navegar por un amplio catálogo de películas, las cuales podrán ser descargadas sin tiempos de espera ni publicidad. Al no estar destinado a ningún cliente particular, sino que es una iniciativa propia, puedo permitirme el lujo de hacerlo totalmente gratuito gracias a todas las herramientas Freeware disponibles actualmente.

El proyecto es básico pero efectivo, contando solamente con dos virtualizaciones de Ubuntu Server 22.04 y un proveedor de servicio DynDNS he podido llevar a cabo el proyecto, con el uso de Apache y MySQL DB para desarrollar toda la estructura de datos y ficheros y PHP para el traspaso de información de un servidor a otro.

Los servidores operan como podría esperarse de cualquier servicio en red, el servidor Web recoge y devuelve los datos mientras que el servidor BB.DD se encarga del almacenamiento, respaldo y gestión de los mismos.

Un texto breve (una cara aproximadamente) que describa qué se ha hecho en el proyecto, sus principales objetivos, la utilidad que se le quiere dar, si está destinado a algún cliente real, aspectos sobre la tecnología usada y cosas similares que permitan hacerse una idea rápida del trabajo realizado.

Se trata de describir brevemente todos los aspectos más importantes del proyecto destacando en lo posible sus puntos fuertes para permitir comprenderlo fácilmente en una lectura rápida sin tener más referencias del mismo. Por tanto, no debe ser un texto demasiado largo ni complejo.

Palabras Clave

Cine, MySQL, Apache, PHP, automatización, Usuarios.

De 5 a 7 palabras[[1]](#footnote-1) clave que mencionen conceptos de capital importancia en el proyecto: Cosas que el proyecto manipula (Ej.: “Máquinas Expendedoras”, “Automóviles”), tecnologías usadas (Ej.: J2EE, RMI), utilidad del proyecto (Ej.: Gestión de Existencias), temática (Ej.: Historia Medieval, Aeronáutica) y cosas similares.

Si finalmente saliesen demasiados términos, conviene hacer una selección de los más relevantes para quedarse con el número indicado.

*Abstract*

Traducción al inglés del resumen anterior. Conviene hacerlo una vez se tenga la versión definitiva de dicho resumen. Se recomienda consultar al director del proyecto acerca de si considera adecuado que aparezca esta sección.

*Keywords*

Ídem a la sección anterior. Se recomienda consultar al director del proyecto acerca de si considera adecuado que aparezca esta sección.

Índice General

NOTA: No debemos olvidarnos de generarlo cuando se cierre la documentación, puesto que en caso contrario pueden quedar referencias mal actualizadas o texto erróneo (tanto éste como el de figuras). **BORRAR ESTA NOTA EN LA DOCUMENTACIÓN FINAL**.

[Capítulo 1. Memoria del Proyecto 18](#_Toc132298799)

[1.1 Objetivos y Alcance del Proyecto 18](#_Toc132298800)

[Capítulo 2. Introducción 19](#_Toc132298801)

[2.1 Estudio de la Situación Actual 19](#_Toc132298802)

[2.2 Evaluación de Alternativas 19](#_Toc132298803)

[2.2.1 Sistema 1 20](#_Toc132298804)

[2.2.2 Sistema 2 20](#_Toc132298805)

[Capítulo 3. Aspectos Teóricos 21](#_Toc132298806)

[3.1 Concepto 1 21](#_Toc132298807)

[3.2 Concepto 2 21](#_Toc132298808)

[Capítulo 4. Planificación del Proyecto y Resumen de Presupuestos 22](#_Toc132298809)

[4.1 Planificación 22](#_Toc132298810)

[4.2 Resumen del Presupuesto 23](#_Toc132298811)

[Capítulo 5. Análisis 24](#_Toc132298812)

[5.1 Definición del Sistema 24](#_Toc132298813)

[5.2 Requisitos del Sistema 24](#_Toc132298814)

[5.3 Análisis de Interfaces de Usuario 25](#_Toc132298815)

[5.4 Especificación del Plan de Pruebas 27](#_Toc132298816)

[5.4.1 Pruebas Unitarias 27](#_Toc132298817)

[5.4.2 Pruebas de Integración 27](#_Toc132298818)

[5.4.3 Pruebas del sistema 27](#_Toc132298819)

[5.5 Diseño de la Base de Datos 28](#_Toc132298820)

[5.5.1 Descripción del SGBD Usado 28](#_Toc132298821)

[5.5.2 Integración del SGBD en Nuestro Sistema 28](#_Toc132298822)

[5.5.3 Diagrama E-R 28](#_Toc132298823)

[Capítulo 6. Implementación del Sistema 29](#_Toc132298824)

[6.1 Estándares y Normas Seguidos 29](#_Toc132298825)

[6.2 Lenguajes de Programación 29](#_Toc132298826)

[6.3 Herramientas y Programas Usados para el Desarrollo 29](#_Toc132298827)

[6.3.1 Programa 1 29](#_Toc132298828)

[6.3.2 Programa 2 29](#_Toc132298829)

[6.4 Creación del Sistema 30](#_Toc132298830)

[6.4.1 Problemas Encontrados y soluciones adoptadas. 30](#_Toc132298831)

[Capítulo 7. Manuales del Sistema 31](#_Toc132298832)

[7.1 Manual de Instalación 31](#_Toc132298833)

[7.2 Manual de Ejecución 31](#_Toc132298834)

[7.3 Manual de Usuario 31](#_Toc132298835)

[7.4 Manual del Programador 31](#_Toc132298836)

[Capítulo 8. Conclusiones y Ampliaciones 33](#_Toc132298837)

[8.1 Conclusiones 33](#_Toc132298838)

[8.2 Ampliaciones 33](#_Toc132298839)

[Capítulo 9. Referencias Bibliográficas 34](#_Toc132298840)

[9.1 Libros y Artículos 34](#_Toc132298841)

[9.2 Referencias en Internet 34](#_Toc132298842)

[Capítulo 10. Apéndices 37](#_Toc132298843)

[10.1 Contenido Entregado en el pendrive. 37](#_Toc132298844)

Índice de Figuras

[Figura 4.1. Ejemplo de diagrama de Gantt 19](#__RefHeading___Toc346385392)

[Figura 4.2. Ejemplo de cronograma 20](#__RefHeading___Toc346385393)

[Figura 5.6. Boceto de una interfaz 26](#__RefHeading___Toc346385394)

1. Memoria del Proyecto

Esta memoria resume los pasos llevados a cabo para la creación del proyecto “W&DB”, para obtener información técnica sobre cada apartado consulte su correspondiente sección de esta documentación.

El primer paso fue hacer el diagrama para previsualizar la labor a llevar a cabo, una vez hecha una primera [estructura de distribución de trabajo](#planificación) comencé los preparativos.

Primero se instalaron dos maquinas gemelas en lo referente a Hardware, ambas sobre Ubuntu Server 22.04, se configuró su red habilitando el enmascaramiento NAT para permitir a ambas maquinas trabajar en una subred privada y acceder a internet simultáneamente.

El concepto de memoria de un proyecto es, en esencia, un resumen del proyecto para personas que desconozcan o no posean conocimientos avanzados de la naturaleza del proyecto y/o sus tecnologías, o incluso no posean conocimientos específicos de informática. Por tanto, debemos orientar la de manera que cualquier persona pueda entender que se ha hecho durante todo el proyecto.

* 1. Objetivos y Alcance del Proyecto

1. Introducción
   1. Estudio de la Situación Actual

Este proyecto puede compararse con el sistema de servicio que ofrecen ciertas empresas como Netflix, HBO o Disney y Amazon en sus respectivos Disney+ y Amazon Prime Video. Al igual que en las plataformas mainstream de streaming, con “W&DB” puedes crearte una cuenta y navegar por un catálogo de películas para ver aquella que más te apetezca.

Se buscaron distintas formas de hacer este mismo proyecto, pero se concluyó que hacerlo desde cero, de forma totalmente “artesanal” sería lo mas justo y lógico para poder adaptar el proyecto a unas necesidades a la carta.

Una de las ventajas que incluye el proyecto es el uso de lenguajes comunes, cualquier persona con algo de experiencia en Bash (linux), PHP y MySQL puede entender, estudiar, analizar y clonar o mejorar este proyecto.

En esta sección deben identificarse y describirse sistemas similares al que se va a desarrollar, estableciendo una comparación entre lo que ofrecen estos sistemas y lo que pretendemos lograr con el proyecto, para de esta forma diferenciar nuestro desarrollo de lo ya existente.

No tienen por qué ser sistemas que hagan lo mismo que el nuestro, sino que pueden ser sistemas que contengan funcionalidad en común con una parte significativa o bien que estén orientados a un conjunto de potenciales usuarios similar.

En esta sección también es adecuado evaluar las posibles herramientas o lenguajes de programación utilizables para el proyecto y determinar cuál (o cuales) se adaptan mejor a nuestras necesidades concretas (de forma justificada).

Conviene en general destacar los puntos en común y las principales discrepancias entre estos sistemas y el nuestro, con la idea de ver en qué sentido nuestro desarrollo supone una ganancia o mejora sobre ellos (también puede orientarse a resolver ciertos defectos de los mismos, mejorar algunas funciones para hacerla más completa, rápida o fácil de usar, etc.). Si los sistemas carecen de alguna funcionalidad que el nuestro va a incorporar, conviene también destacarlo (precisamente esto puede ser una de las principales aportaciones del mismo).

En general, conviene usar esta sección como un primer paso para “promocionar” las “bondades” de nuestro proyecto.

* 1. Evaluación de Alternativas

Originalmente se contempló desarrollar todo este proyecto sobre preinstalaciones y servicios de terceros ya creados, no es el método seguido ya que el objetivo de este proyecto es poner a prueba los conocimientos y habilidades del alumno, pero podría ser una opción más liviana respecto a carga de trabajo, ya que ahorraría tiempo entre los plazos de desarrollo, puesto que sería instalar y configurar en lugar de desarrollar totalmente de cero. Algunos de los servicios que podrían implementarse son DaCast o aprovechar las posibilidades de inserción disponibles en Google Drive

En esta sección se describirán, una por una, todas las alternativas estudiadas. Conviene estudiar 3 o 4 alternativas importantes, salvo que por algún motivo justificado se deba incluir un número menor o mayor de las mismas. En todo caso, siempre es conveniente cuidar de que en esta sección haya un conjunto de sistemas significativo. En función de lo dicho anteriormente, cada sistema podrá dividirse en tres secciones: “Descripción”, “Ventajas” e “Inconvenientes”, aunque es posible cualquier otra división que contenga los aspectos descritos, dependiendo de qué tipo de sistemas se estudien.

* + 1. Andersenlab

En un principio se contempló el uso de algún servicio de terceros como puede ser Andersenlab, una plataforma que ofrece todas las herramientas y preinstalaciones para la creación de un entorno de streaming, dependiendo el plan deseado para contratar podrías tener preinstalaciones completas del servidor web con una página inicial creada por defecto, espacio en servidores para almacenar películas, preinstalaciones de toda la infraestructura de red necesaria, ciertos parámetros de seguridad, etc. y a pesar de que, en esencia, no altera la naturaleza del proyecto se decidió desechar esta idea considerando que podría interpretarse como hacer “trampa”, ya que allanaría demasiado el camino de cara al objetivo final de este proyecto.

* + 1. Google Drive

Una posibilidad que también se contempló fue el uso de los servicios Drive como contenedor de almacenamiento y ventana de visualización, ya que Drive cuenta con múltiples opciones de integración podría ser viable utilizar la nube en el proyecto, pero limitaría el rango del mismo ya que no tendría manera de administrar o gestionar a un nivel profundo los sistemas de almacenamiento, por lo que esta idea también quedó desechada.

1. Aspectos Teóricos

Esta sección está destinada a describir brevemente aquellos conceptos, herramientas y tecnologías existentes que vamos a usar en nuestro proyecto. Conviene limitarse a un máximo de 2 hojas por cada uno de ellos aproximadamente (es una descripción, no un tutorial sobre ello), describiendo por qué y para qué usamos esto en nuestro proyecto pero sin hacer descripciones muy grandes de características y similares (para ello podemos referenciar enlaces Web, libros, etc. que las describan más detalladamente, que posteriormente podemos incluir en la bibliografía):

* En el caso de conceptos sobre los que trata el proyecto, debemos decir su origen, aplicación e importancia, para tener una idea de lo que se va a tratar. A modo de ejemplo, si usamos bases de datos orientadas a objetos debemos decir que son, que importancia tienen y describir brevemente su forma de trabajo. Si nuestro proyecto es sobre emisión de video en directo, debemos describir la técnica usada para ello y como funciona, etc.
* En el caso de tecnologías debemos decir su creador u origen, para lo que sirve y que aplicaciones ha tenido, para qué la vamos a usar, la versión que hemos empleado, etc.
* En el caso de herramientas debemos decir su fabricante y la utilidad que le vamos a dar dentro de nuestro proyecto.

Por último, debemos recordar describir de esta forma todo aquello que usemos en el proyecto y dar especial importancia a todo aquello que sea “novedoso”.

* 1. Concepto 1

Definición

* 1. Concepto 2

Definición

1. Planificación del Proyecto y Resumen de Presupuestos
   1. Planificación

El proyecto se divide en 8 semanas de desarrollo con 4 hitos:

* Las dos primeras semanas se centrarán en el desarrollo del servidor web, presentando los resultados el día 26 de abril de 2023.
* Las dos semanas siguientes el desarrollo se enfocará en la conectividad del servidor web, configurando un DDNS y redirigiendo el tráfico hacia el servidor web, los resultados serán presentados el 10 de mayo de 2023.
* Presuponiendo el cumplimiento de plazos, las semanas 5 y 6 se llevará a cabo el desarrollo del servidor de Bases de Datos, al igual que en el servidor web, se configurará el ecosistema de forma que pueda demostrarse su funcionamiento el día 24 de mayo de 2023.
* Las dos últimas semanas se enfocarán en interconectar todo el ecosistema para poder desplegar el servicio de forma totalmente operativa, contando con una ultima semana final para, con ayuda del tutor, poder solucionar problemas que no se hayan resuelto para entonces, contando con hacer una presentación final demostrando un funcionamiento pleno el día 8 de junio de 2023.

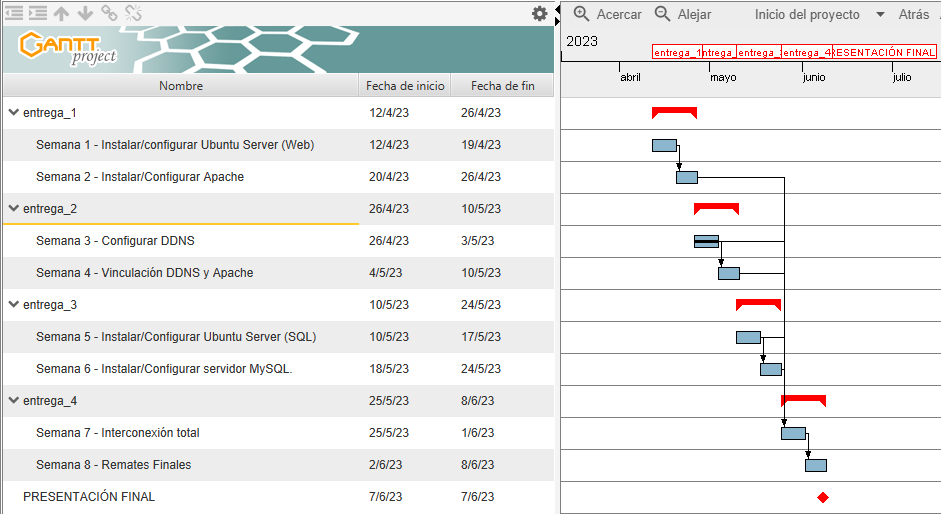


Figura 4.1. diagrama de Gantt del proyecto “W&DB”

* 1. Resumen del Presupuesto

Explicitar el presupuesto de cliente resumido que se ha previsto para la ejecución, de forma que pueda tomarse la decisión de proseguir o no con la ejecución de lo valorado. **Esta sección puede o no tener sentido en función del tipo de proyecto desarrollado. Por ello, se aconseja consultar al director del proyecto acerca de este tema antes de hacer nada relativo a la misma.**

* Debe incluir el coste total de la ejecución para la organización que ha de hacerse cargo de este proyecto.
* En este apartado debe tenerse especial cuidado en presentar las cifras de manera no ambigua, completa, sin costes ocultos y dando un total general desglosado por partidas.
* Habitualmente sólo se ponen los ítems, eliminando del presupuesto los subitems.
* Si aún así resultase excesivamente complejo, se puede resumir en grandes partidas de trabajo que tengan sentido práctico y económico para el cliente.
* Si se usase cualquier otro criterio para hacer este resumen del presupuesto, deberá justificarse adecuadamente. La siguiente tabla muestra un ejemplo de cómo se puede realizar el mismo.



Los formatos de la taba anterior no son obligatorios, deben adaptarse a cada proyecto.

1. Análisis

Este apartado contendrá toda la especificación de requisitos y toda la documentación del análisis de la aplicación, a partir de la cual se elaborará posteriormente el diseño.

* 1. Definición del Sistema

Se trata de describir de nuevo el sistema, pero en lugar de repetir lo que ya hemos dicho de él, tenemos que constatar en este apartado hasta donde vamos a llegar en su construcción, es decir, qué límites vamos a poner en el desarrollo estableciendo qué se va a hacer y qué se va a omitir (en general, hasta donde se va a llegar). Podemos por tanto usar todo lo que hemos dicho en descripciones anteriores para ayudar a describir el alcance del sistema. Conviene dejar claro este apartado para así delimitar la labor de análisis y diseño que vamos a hacer a continuación y evitar así no describir aspectos que se han construido o describir cosas que finalmente no van a construirse.

En el caso de que quede claro implícitamente qué se va a hacer en el sistema, esta sección se puede omitir.

* 1. Requisitos del Sistema

El producto de esta sección se crea para su aprobación formal, es decir, que los potenciales clientes deben ver a partir de él las especificaciones completas del sistema. Además, esta sección construirá una base para solicitar cambios en los requisitos antes de avanzar más en la construcción del sistema.

Los requisitos del sistema se deben mostrar en una tabla como la que se presentará a continuación con ejemplos, ordenados por algún criterio lógico en función de a lo que se refieran. Por ejemplo, si tenemos usuarios tiene sentido agrupar todos los requisitos que tengan que ver con los usuarios.

Tampoco es necesario crear una única tabla para todos los requisitos, pueden crearse varias tablas que agrupen los requisitos que se refieran a una entidad.

Este apartado debe incluir también antes de la tabla de requisitos, si existe como tal, la **especificación textual** que el cliente nos proporcione sobre la aplicación, fruto de las reuniones que hayamos tenido con él o de las entrevistas que podamos haber llevado a cabo. Ha de tenerse en cuenta que esta información es la que usaremos para extraer los requisitos de la aplicación, por lo que no debe faltar sin contamos con ella.

|  |  |  |
| --- | --- | --- |
| **Código** | **Nombre Requisito** | **Descripción del Requisito** |
| R1.1 | Insertar Usuario | Se debe añadir un usuario al sistema una vez leídos y validados sus datos. |
| R1.2 | Leer Datos Usuario | Deben pedirse los datos completos de un usuario del sistema |
|  |  |  |

Esta tabla puede usarse para todos los **requisitos funcionales** (lo que la aplicación debe hacer), pero debemos especificar también otro tipo de **requisitos no funcionales**, de los que a continuación se muestran ejemplos típicos:

* **Requisitos de Usuario**: Si exigimos al usuario algún tipo de conocimiento previo para manejar la aplicación o alguna de sus partes, debemos especificarlo aquí.
* **Requisitos Tecnológicos**: Si el programa establece que debe funcionar con una versión concreta de un determinado programa o sistema, o bien en un entorno o sistema operativo concreto debemos también hacerlo constar.
* **Requisitos de Usabilidad**: Normas de usabilidad que la aplicación debe cumplir obligatoriamente (nosotros podemos especificar requisitos adicionales a los que el cliente solicite y desarrollarlos en la sección correspondiente).
* **Requisitos de Seguridad**: Si debemos implementar algún tipo de medida de seguridad en el sistema (encriptación de datos, etc.).
* **Requisitos de Tiempo de Respuesta**: Si el sistema debe proporcionar una respuesta en un tiempo acotado.
  1. Análisis de Interfaces de Usuario

A la hora de diseñar un interfaz de usuario, debemos cumplir con las normas de comunicación persona-máquina existentes, procurando que el interfaz sea usable, permita manejar el programa de manera eficiente y que no sea propenso a provocar errores en los usuarios. Esto debe hacerse así porque los diseños obedecen al resultado de hacer un diseñado centrado en el usuario, que simplemente nos lleva a simular en la pantalla el trabajo que realiza sobre una mesa o, en general, su entorno de trabajo existente hasta el momento. Un enlace que puede ser de ayuda a la hora de tomar determinadas decisiones a la hora de construir el interface de la aplicación es el siguiente: <http://www.ambysoft.com/essays/userInterfaceDesign.html>.

En esta sección debemos crear la especificación de las interfaces entre el usuario y el sistema a construir, incluyendo todos los diferentes tipos de pantallas que van a existir, los cuadros de diálogo o los informes que le proporcionarán al usuario.

En este apartado también es importante identificar posibles grupos de usuarios para así aplicar las pantallas a dichos grupos, así como detallar otros aspectos, como lo que vamos a incluir en las pantallas para cumplir con normas de accesibilidad y usabilidad.

Para los distintos tipos de pantallas, se debe hacer un esquema que muestre la disposición de las mismas, que permita identificar donde irá cada elemento y las diferentes zonas de trabajo. Se muestra un ejemplo con este dibujo:



Figura 5.6. Boceto de una interfaz

Otra posible opción para este apartado es diseñar ya las pantallas definitivas sin funcionalidad, solo para ver como quedarán en el producto final (es decir, crear un **prototipo**), lo que tiene la ventaja de poder enseñarle al cliente el aspecto de la aplicación desde un primer momento.

* 1. Especificación del Plan de Pruebas

En esta sección crearemos y diseñaremos el plan de pruebas de la aplicación y sus funciones, así como todos los mecanismos que utilizaremos para detectar errores y corregirlos ya en la fase de implementación.

Las pruebas contemplarán aspectos tanto de funcionalidad de la aplicación como de aspectos de los usuarios o clientes de la misma.

Se contemplarán hasta cinco tipos de pruebas:

* + 1. Pruebas Unitarias
* Una prueba unitaria es una forma de probar el correcto funcionamiento de un módulo de código, o en este caso una clase individual que cumple con una función concreta. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. A partir de los casos de uso, los escenarios y clases vistos anteriormente, debemos desarrollar pruebas unitarias que consideremos necesarias y especificar los resultados que se espera encontrar una vez ejecutada la operación sobre cada una de ellas. Es conveniente tabular estas pruebas para su aplicación posterior.
  + 1. Pruebas de Integración
* Las pruebas de integración comprenden verificaciones asociadas a grupos de componentes, verificando que éstos funcionan correctamente cuando estos son ensamblados para cumplir con una función concreta. Para ello, cada escenario debe probarse con el mayor número de entradas posibles (y relevantes) que sea posible, incluyéndose entradas con datos correctos y con datos incorrectos para probar que el sistema reacciona correctamente ante errores de los usuarios. Para elaborar estas pruebas debemos tener en cuenta las características de la aplicación.
  + 1. Pruebas del sistema
* Las pruebas del sistema son pruebas de integración del sistema construido completo, que permiten probar el conjunto de todo el sistema y que sus relaciones con otros sistemas que necesite son correctas, verificando así que todas sus especificaciones funcionales y técnicas se cumplen.
  1. Diseño de la Base de Datos

Esta sección es la indicada para describir todo lo relativo al sistema de gestión de bases de datos que vamos a usar en nuestra aplicación (si usamos alguno). Esto incluye hablar del sistema en sí, la versión utilizada, las clases empleadas al usarlo, como se integra en el sistema y el diagrama E-R de las tablas y relaciones creadas, todo ello en los apartados que se muestran a continuación.

* + 1. Descripción del SGBD Usado
    2. Integración del SGBD en Nuestro Sistema
    3. Diagrama E-R

1. Implementación del Sistema
   1. Estándares y Normas Seguidos

Descripción breve de los estándares y normas que hayamos usado en nuestra aplicación a la hora de desarrollar su código y si nos hemos ocupado de validar que esos estándares se cumplan efectivamente.

* 1. Lenguajes de Programación

Bash (Linux)

MySQL

PHP

Descripción de los lenguajes de programación usados, su versión y distribución, los módulos o complementos de los mismos empleados, etc.

* 1. Herramientas y Programas Usados para el Desarrollo

Descripción de todas las herramientas de desarrollo (Eclipse, Visual Studio, etc.), sistemas adicionales existentes, complementos y otros productos software que necesitemos para la implementación de nuestro sistema. Debemos dejar claro que versión usamos, para qué y cómo interactuará con nuestro sistema.

* + 1. Programa 1
    2. Programa 2
  1. Creación del Sistema

Todos los aspectos con los que nos hemos encontrado durante la implementación debemos describirlos aquí.

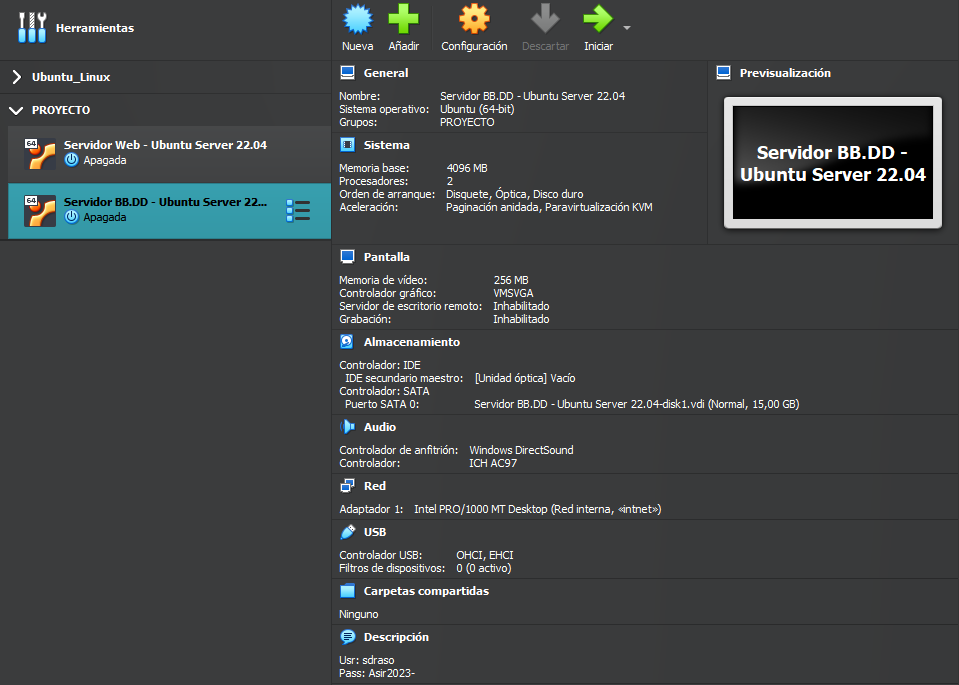
* + 1. Problemas Encontrados y soluciones adoptadas.

Enumeramos los problemas encontrados en el desarrollo y la solución que le hemos dado, si hubo alguno que merezca la pena destacar.

1. Manuales del Sistema
   1. Manual de Instalación

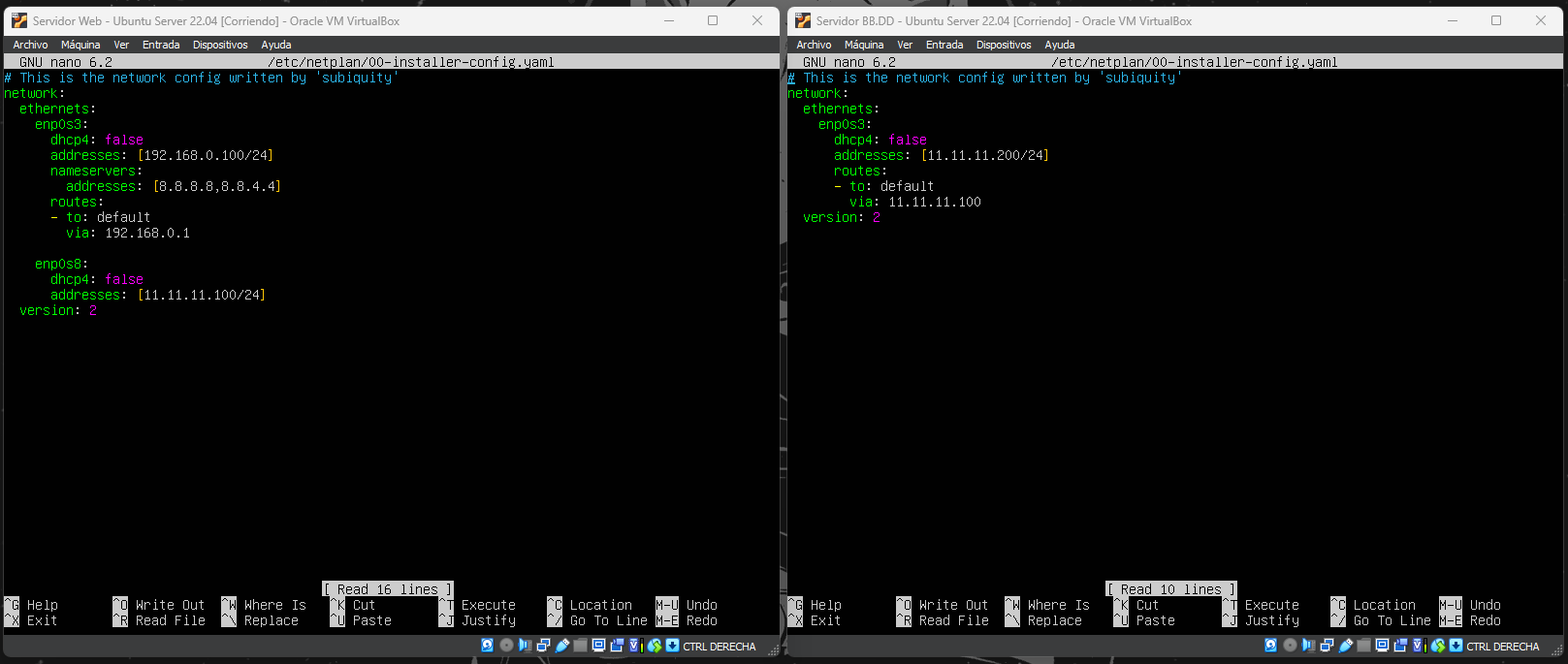
En este manual se explicará de forma detallada como llevar a cabo una réplica de este proyecto paso a paso, todos los pasos dados durante el desarrollo de “W&DB” han sido registrados aquí:

Comenzamos creando dos máquinas virtuales que, cumpliendo [los requisitos de sistema](#RequisitosDelSistema), alberguen dos entornos Ubuntu Server 22.04.



A continuación, configuraremos las interfaces de red de forma que el servidor Web y el servidor BBDD operen en una red local privada, programando NAT para que ambas maquinas puedan tener acceso a internet.

La configuración de red debería ser algo similar a esto:



Completada la configuración de red falta habilitar NAT, lo primero que debemos hacer es dirigirnos al fichero **/etc/sysctl.conf** y descomentar la siguiente línea:



Con esto habilitamos el reenvio de paquetes entre las tarjetas, una vez editado y guardado, podemos comprobar si todo ha sido correcto con el siguiente comando:



Una vez tenemos completado el NAT ya podemos instalar apache:

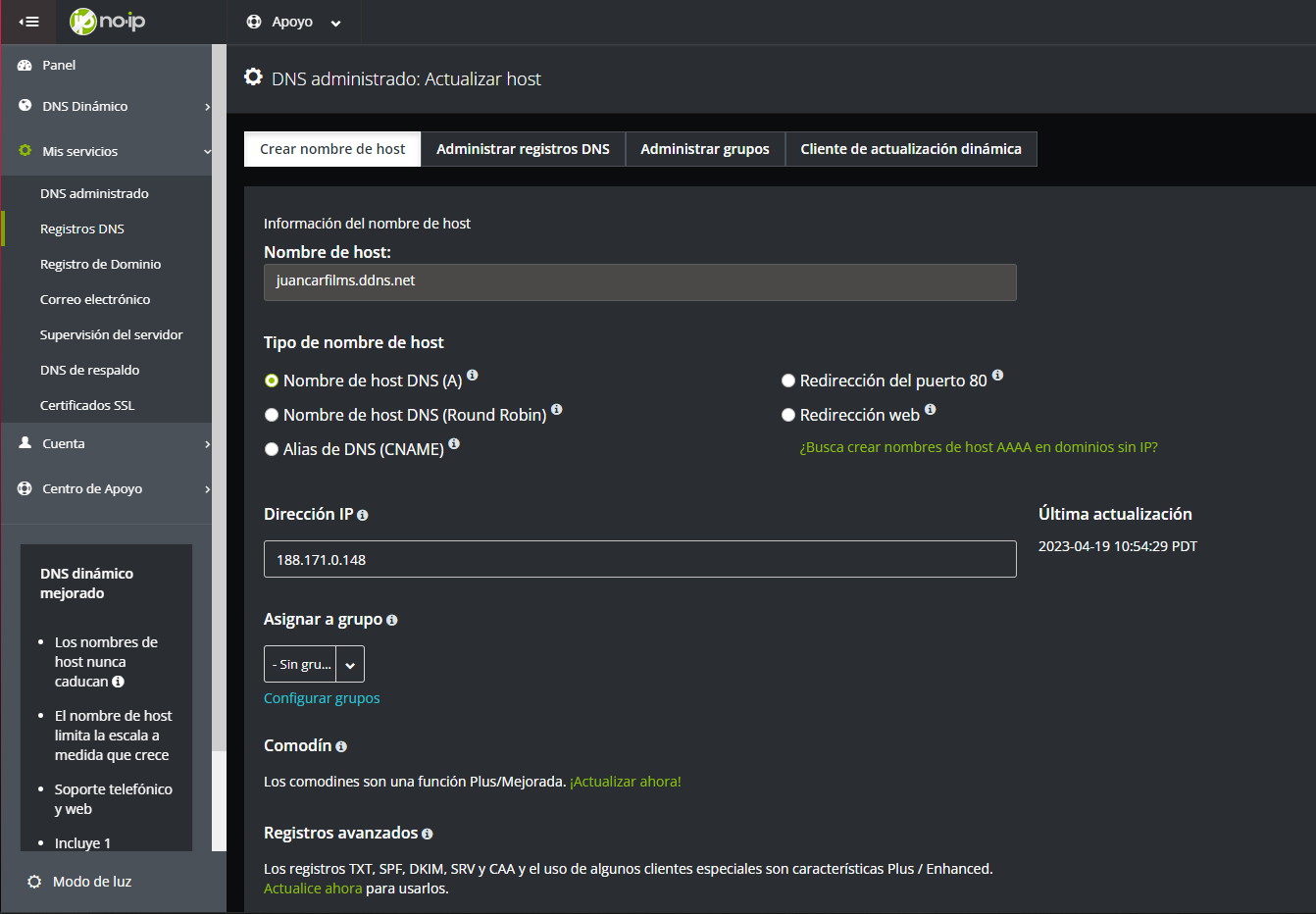


Si la instalación es correcta tendríamos que poder ver la web default de apache intentando acceder a través de un buscador a la IP de la máquina virtual:

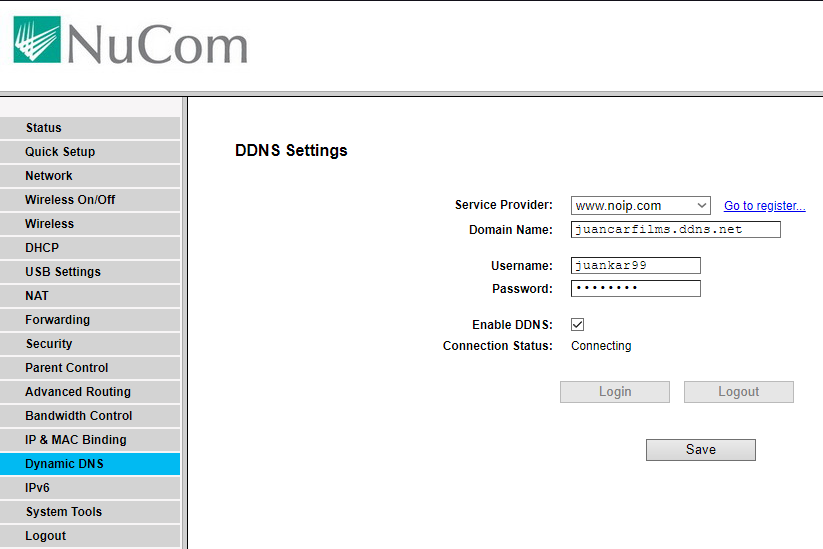
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

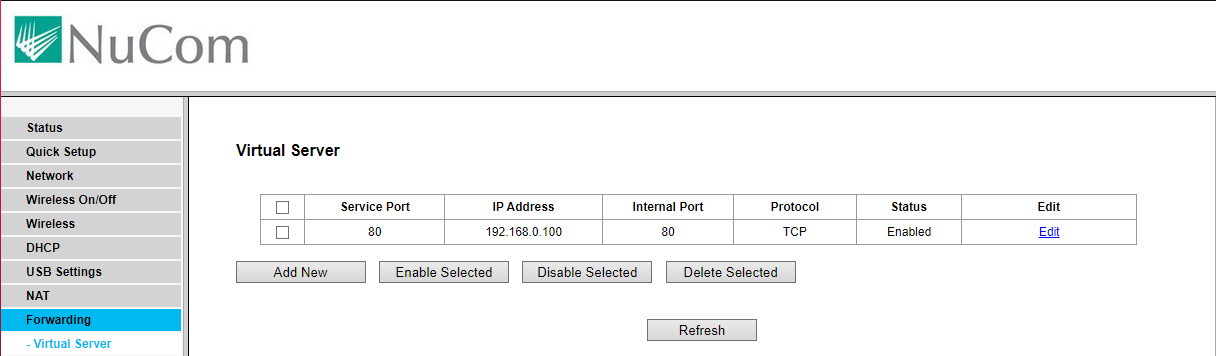
Una vez el sevidor Apache está funcionando necesitamos hacer que el tráfico entrante por el puerto 80 se redirija al Apache instalado en Ubuntu, para que se asocie el dominio “juancarfilms.ddns.net” a mi IP externa usaré un DNS Dinámico, utilizando un proveedor de servicio gratuito como NO-IP, una vez creada la cuenta en el servicio configuramos el DNS con los parámetros deseados:

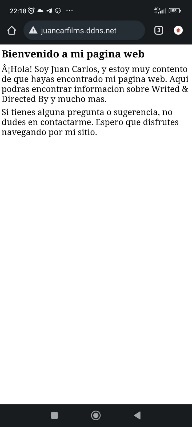


También es necesario habilitar el DNS Dinámico en la configuración del router, estableciendo la siguiente configuración:



Por último, para poder acceder al servidor Apache a través del dominio necesitaremos habilitar un “Virtual Server”, una función incluida en muchos routers con la cual podremos dar visibilidad externa a servicios montados en LAN, de esta manera hacemos el servidor Apache accesible desde WWW:



Reiniciamos el servicio con: **sudo systemctl restart apache2**

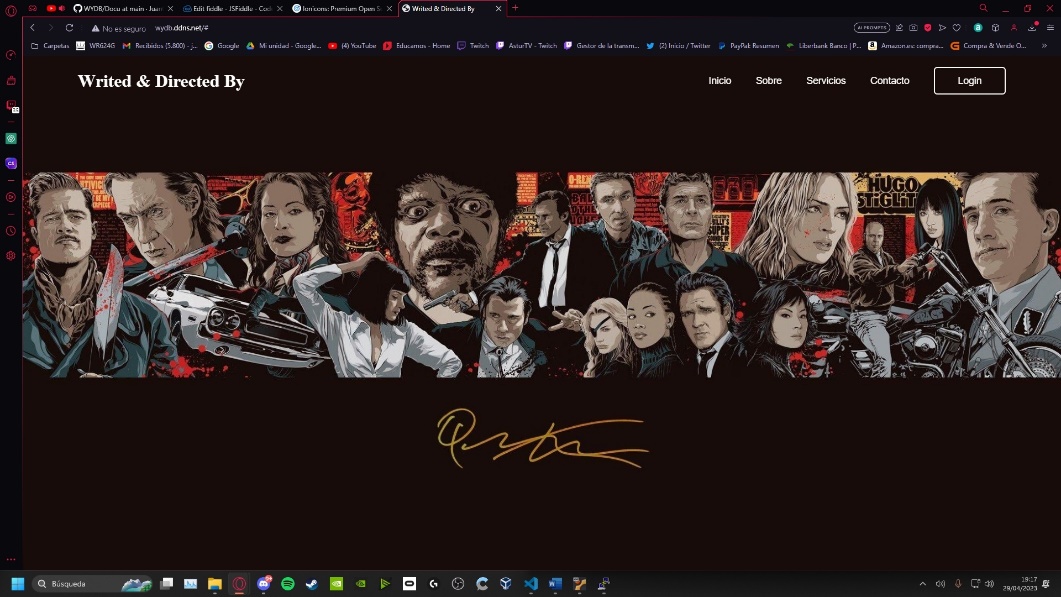
Esperados unos segundos ya se tendría que poder acceder desde internet, para hacer la prueba he intentado acceder al index provisional a través de mi móvil utilizando mi conexión de datos:

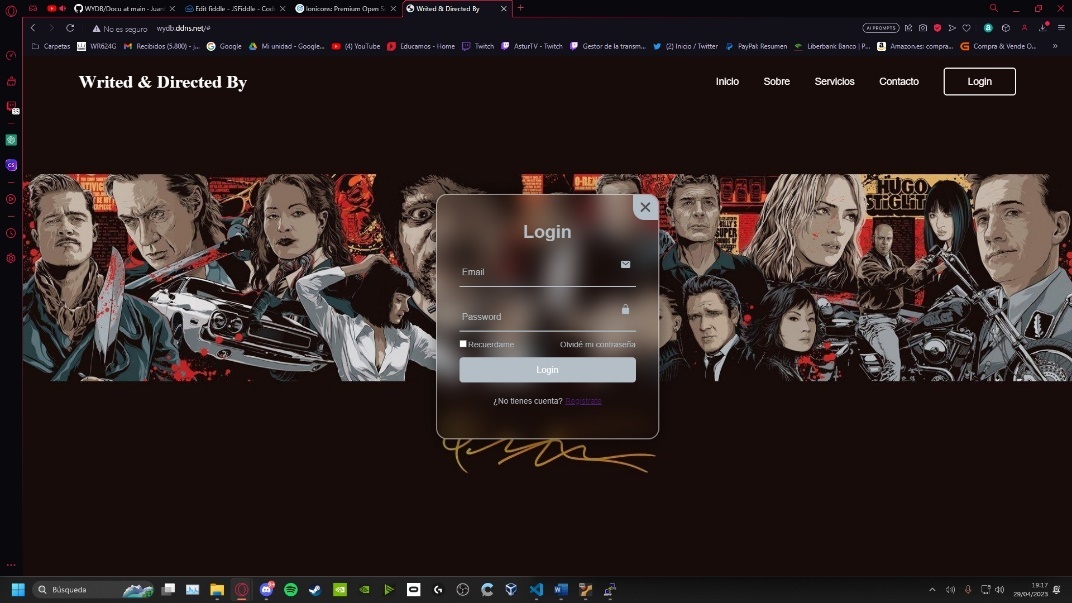
Habilitado el acceso al servidor, el siguiente paso que he llevado a cabo ha sido desarrollar un poco la interfaz del Log in, habilitando también una sección para crear la cuenta, todos estos datos redirigidos a ficheros .php para más tarde ser procesados y almacenados en la base de datos.

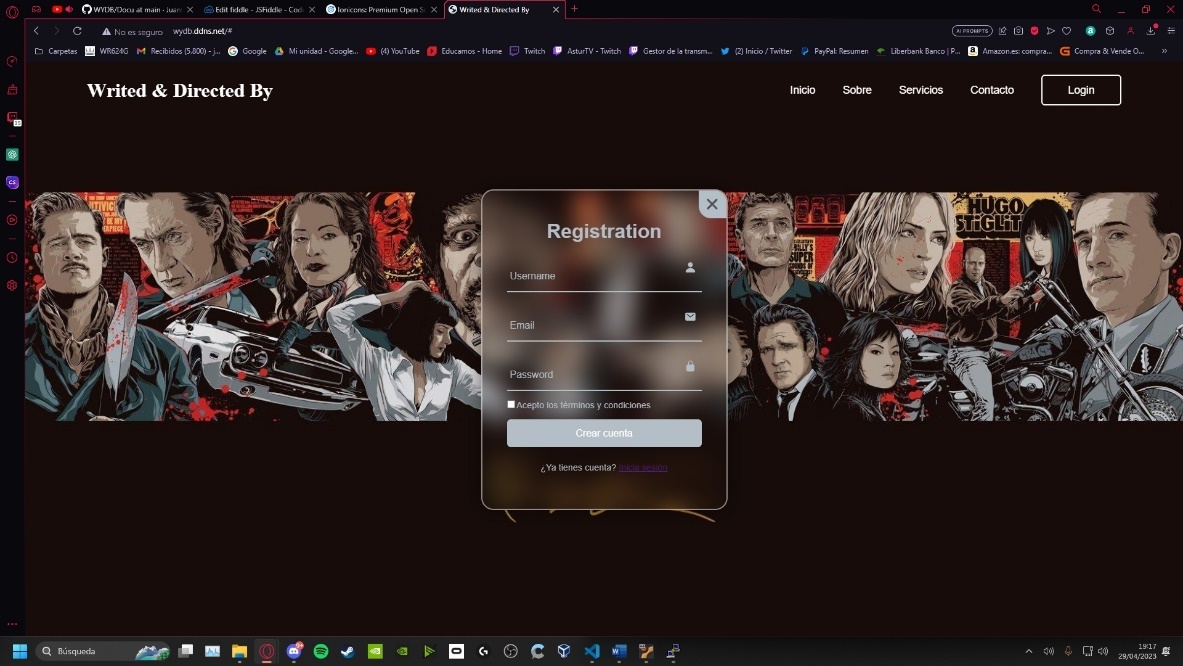
Todo los recursos utilizados en el desarrollo web está subido a [GitHub](https://github.com/JuanCAPlos/WYDB). Sientase libre de acceder a mirarlo.

Como “Diseño Web” es una de las habilidades de programación que menos tengo desarrolladas he tenido que buscar documentación sobre como combinar HTML, CSS y JavaScript, siguiendo el siguiente [video](https://www.youtube.com/watch?v=p1GmFCGuVjw) he conseguido desarrollar una interfaz bastante estética y completa, los emoticonos del login han sido extraidos e implementados de este [repositorio](https://ionic.io/ionicons). Este primer login está compuesto por HTML:

El resultado provisional es el siguiente:







Elaborar un manual que contemple todos los pasos necesarios para instalar nuestro sistema, incluyendo la instalación de otras herramientas o software cualquiera (sea o no comercial) necesario para que funcione. Debemos explicarlo todo paso a paso de forma clara y acompañarlo por capturas de pantalla adecuadas.

* 1. Manual de Ejecución

Este manual contemplará todos los pasos necesarios para el arranque de nuestro sistema, lo que es especialmente importante en caso de sistemas con clientes y servidores o distintos procesos que deban arrancarse independientemente.

Por otra parte, también debemos incluir procedimientos para parar adecuadamente la aplicación.

* 1. Manual de Usuario

El manual de usuario es algo muy importante debido a que es el documento que servirá a los usuarios de nuestro sistema para saber cómo funciona cada una de las partes de nuestra aplicación. Debemos pues describir cómo funcionan todas las opciones de las misma, que parámetros tiene, que cosas debemos hacer para que todas las operaciones funcionen correctamente y cualquier otro aspecto que consideremos oportuno para explicar el funcionamiento del sistema.

No debemos escatimar detalles en este manual ya que es la herramienta para que los usuarios comprendan nuestro sistema. También debemos hacer el mayor uso posible de capturas de pantalla para mejorar nuestras explicaciones.

* 1. Manual del Programador

En este manual debemos describir cualquier aspecto que pueda ayudar a otros programadores a ampliar, modificar o entender aspectos de la construcción de nuestra aplicación. Debemos por tanto hacer una descripción general de los distintos aspectos involucrados en la construcción del sistema que puedan ser más difíciles de entender y también describir los procedimientos necesarios para hacer ciertas ampliaciones que hayamos contemplado en el diseño del sistema (añadir nuevas entidades, nuevos atributos a entidades existentes, nuevos servicios que usen a los ya desarrollados, modificaciones en la interfaz, etc.).

1. Conclusiones y Ampliaciones
   1. Conclusiones

Conclusiones del sistema: Qué hemos elaborado, si los resultados están dentro de lo esperado, si hemos cumplido las expectativas, justificación de haber escogido las mejores opciones para cada uno de los aspectos del sistema, etc.

* 1. Ampliaciones

Cualquier labor de ampliación que tengamos contemplada en el sistema debe ser descrita aquí, mencionando en qué consiste, cómo ampliará el sistema, qué ventajas nos aporta y porqué no se ha incluido en el sistema diseñado, entre otros aspectos.

1. Referencias Bibliográficas
   1. Libros y Artículos

Videos:

Para conseguir que el DynDNS redirija el tráfico entrante por el puerto 80 al servidor Apache. [Video](https://www.youtube.com/watch?v=AOA06CPic08)

Durante este proyecto ChatGPT ha sido una herramienta muy pontente con la cual concentrar datos concretos que eran de mi interés. Muy lejos de ser una herramienta “tramposa” me ha parecido un motor de búsqueda elevada a la máxima potencia.

Libros y artículos usados de alguna forma durante el desarrollo del proyecto o su documentación.

**Formato sugerido:**

**[<PrimerApellidoAutor><DosUltimosDigitosDelAño>]**<Apellidos1, Nombre1; Apellidos2, Nombre2;…>. ”<Título del libro o Articulo>”. <Editorial o lugar de publicación>. <Año (4 cifras)>.

**Ejemplo:**

**[Redondo07]** Redondo L., J. Manuel; De Tal y Cual, Menganito. ”Ejemplo para la plantilla de PFC”. Universidad de Oviedo. 2007.

Si tenemos el ISBN, debemos también ponerlo al final.

* 1. Referencias en Internet

Páginas Web consultadas para cualquier aspecto relacionado con el desarrollo del sistema o su documentación.

**Formato sugerido:**

**[<PrimerApellidoAutor><DosUltimosDigitosDelAño>]**<Apellidos1, Nombre1; Apellidos2, Nombre2;…>. “<Título de la página Web>”. <URL>. <Año en el que se consultó (4 cifras)>.

**Ejemplo:**

**[Redondo07]** Redondo L., J. Manuel; De Tal y Cual, Menganito. “Título de la página Web de ejemplo”. www.unaurlcualquiera.com. 2007.

Si tenemos más datos que permitan localizar la información dentro de la página, podemos ponerla donde consideremos oportuno.

Esta referencia es real (se usa dentro del documento) y debe dejarse aquí siempre que usemos el cuestionario que la menciona en la sección de usabilidad.

**[Hassan08]** Hassan Montero, Y. “Guía de Evaluación Heurística de Sitios Web”. <http://www.nosolousabilidad.com/articulos/heuristica.htm>

1. Apéndices
   1. Contenido Entregado en el pendrive.

Debe contener estructura de directorios con la memoria en pdf, programas, presentación, máquinas virtuales utilizadas para la defensa,….

1. No necesariamente debe ser una única palabra, pueden ser varias. Por ejemplo, si el proyecto tratase sobre la gestión de calificaciones de Alumnos, una palabra clave válida podría ser “Expediente Académico”. [↑](#footnote-ref-1)