

MEMORIA FINAL DE PROYECTO



CICLO FORMATIVO DE GRADO SUPERIOR

DESARROLLO DE APLICACIONES WEB

AUTORES

JORGE RABADAN Y RICARDO ZORITA

TUTOR

JOSE LUIS GALLEGO



INDICE

1 INTRODUCCIÓN		4
2 ALCANCE DEL PROYECTO		5
3 ESTUDIO DE VIABILIDAD		5
3.1 Estado actual del sistema	5	
3.2 Requisitos del cliente	6	
3.3 Posibles soluciones	6	
3.4 Solución elegida	6	
3.5 Planificación temporal de las tareas del proyecto [nuevo proyecto]	6	
3.6 Planificación de los recursos a utilizar	6	
4 ANÁLISIS		7
4.1 Requisitos funcionales	12	
4.2 Requisitos no funcionales	12	
5 DISEÑO		13
5.1 Estructura de la aplicación	13	
5.2 Componentes del sistema	13	
5.3 Arquitectura de la red	13	
5.4 Herramientas	13	
6 IMPLEMENTACIÓN		13
6.1 Entorno de implementación	14	
6.2 Tablas creadas	14	
6.3 Carga de datos	14	
6.4 Ficheros de configuración actualizados	15	
7 PRUEBAS		16
7.1 Casos de pruebas	16	
8 EXPLOTACIÓN		22
8.1 Planificación	22	
8.2 Preparación para el cambio	23	
8.3 Plan de formación	23	
8.4 Implantación propiamente dicha	38	
8.5 Pruebas de implantación	38	
9 DEFINICIÓN DE PROCEDIMIENTOS DE CONTROL Y EVALUACIÓN		39
10 CONCLUSIONES		40
11 FUENTES		41
11.1 Legislación		41
11.2 Bibliografía		42
12 ANEXOS		42

1. INTRODUCCIÓN

Este documento recoge el trabajo realizado para el **módulo de Proyecto** del CFGS en Desarrollo de Aplicaciones Web.

Este módulo profesional complementa la formación establecida para el resto de los módulos profesionales que integran el título en las funciones de **análisis** del contexto, **diseño** o **desarrollo** del proyecto y organización de la **ejecución**.

La fase de alcance y análisis previo del proyecto determina qué queremos hacer y cómo lo desarrollaremos incluyendo diferentes etapas:

- El estudio de viabilidad identifica la necesidad y los requisitos del cliente, aporta las posibles soluciones y desarrolla la opción más viable. Además, aquí se incluye una planificación temporal de las tareas a realizar para el desarrollo del proyecto.
- La fase de análisis describe la aplicación tanto desde el punto de vista práctico (las funcionalidades que se podrán ejecutar) como desde el punto de vista no funcional (uso, accesibilidad, compatibilidad, escalabilidad, etc.)
- La etapa de diseño describe tanto la estructura de la aplicación como los componentes y herramientas del sistema (programación, gestión de datos, ORM, MVC y despliegue).
- En la implementación de la aplicación se describe el entorno usado incluyendo ejemplos de carga de datos, ficheros de configuración, implementaciones de código, etc.
- La realización de pruebas es un paso que se realiza a lo largo de toda la fase de desarrollo e implementación a través de diferentes técnicas. En este apartado se proporcionan ejemplos de diferentes casos de prueba.

- La fase de explotación se corresponde con la presentación de este proyecto, aquí se explican los recursos y tareas necesarias para su implementación.

2. ALCANCE DEL PROYECTO Y ANÁLISIS PREVIO

El propósito que se pretende conseguir en este proyecto es la creación de un blog, en el cual los usuarios se podrán dar de alta y realizar publicaciones.

El usuario dado de alta tendrá acceso a todo lo que se pueda hacer dentro del blog, como, por ejemplo, crear nuevos posts, responder al post que están creados, etc.

El desarrollo de este proyecto se ha llevado a cabo en varias fases: estudio de viabilidad, análisis, diseño, implementación y pruebas, y explotación o ejecución. A continuación, se detallan las actividades/tareas/procedimientos de cada una de estas fases.

3. ESTUDIO DE VIABILIDAD

En esta fase se considera si el proyecto se puede realizar teniendo en cuenta las circunstancias internas y externas de la empresa, las diferentes soluciones posibles y los recursos de los cuales se dispone.

Para ello se hace una valoración del estado actual del sistema y de los requisitos del cliente, se presentará un estudio de soluciones alternativas y la solución elegida por el cliente.

3.1 Estado actual del sistema

La creación del sistema parte de cero, se creará la aplicación web desde el principio.

En la actualidad este proyecto se encuentra en fase de aceptación por parte del cliente para su posterior puesta a producción.

3.2 Resumen de requisitos del cliente

El cliente quiere una web en la se permita a los usuarios registrarse, para que puedan crear, modificar y borrar publicaciones en el blog. Además, contará con la opción de poder valorar las publicaciones y realizar votos en relación con el tema.

3.3 Posibles soluciones

Existen muchas aplicaciones web relacionadas con los blogs, pero ninguna relacionado con el inglés y en la que te permite buscar profesores para aprender.

3.4 Solución elegida

La solución es crear un web desde cero en la que se cumplan todos los requisitos del usuario llamada Correct English.

3.5 Planificación temporal de las tareas del proyecto CORRECT ENGLISH

El tiempo de realización de este proyecto es de 3 meses. El proyecto finalizará el 17 de diciembre del 2018 y se utilizarán todos los medios y recursos posibles para cumplir los plazos de entrega.

Durante el desarrollo se dividirán las tareas, pero si fuese necesario cada miembro trabajará en una u otra tarea.

Jorge Rabadán realizará la parte de Back-end y Ricardo Zorita se centrará en el lado Front-end.

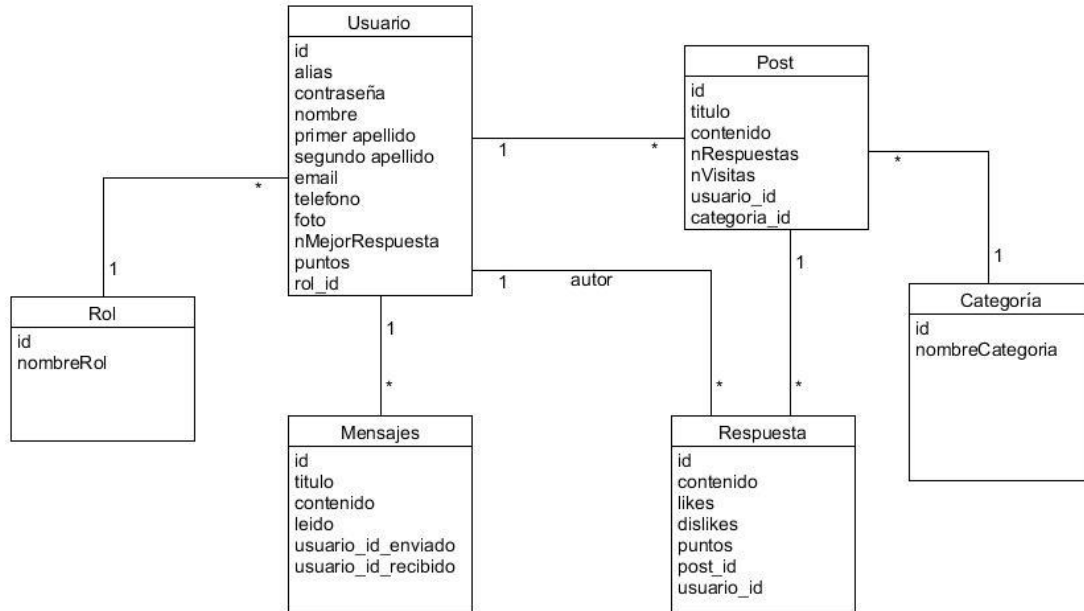
3.6 Planificación de los recursos a utilizar

Para el correcto funcionamiento de la aplicación será necesario un equipo y un servidor de aplicaciones para desplegar la aplicación.

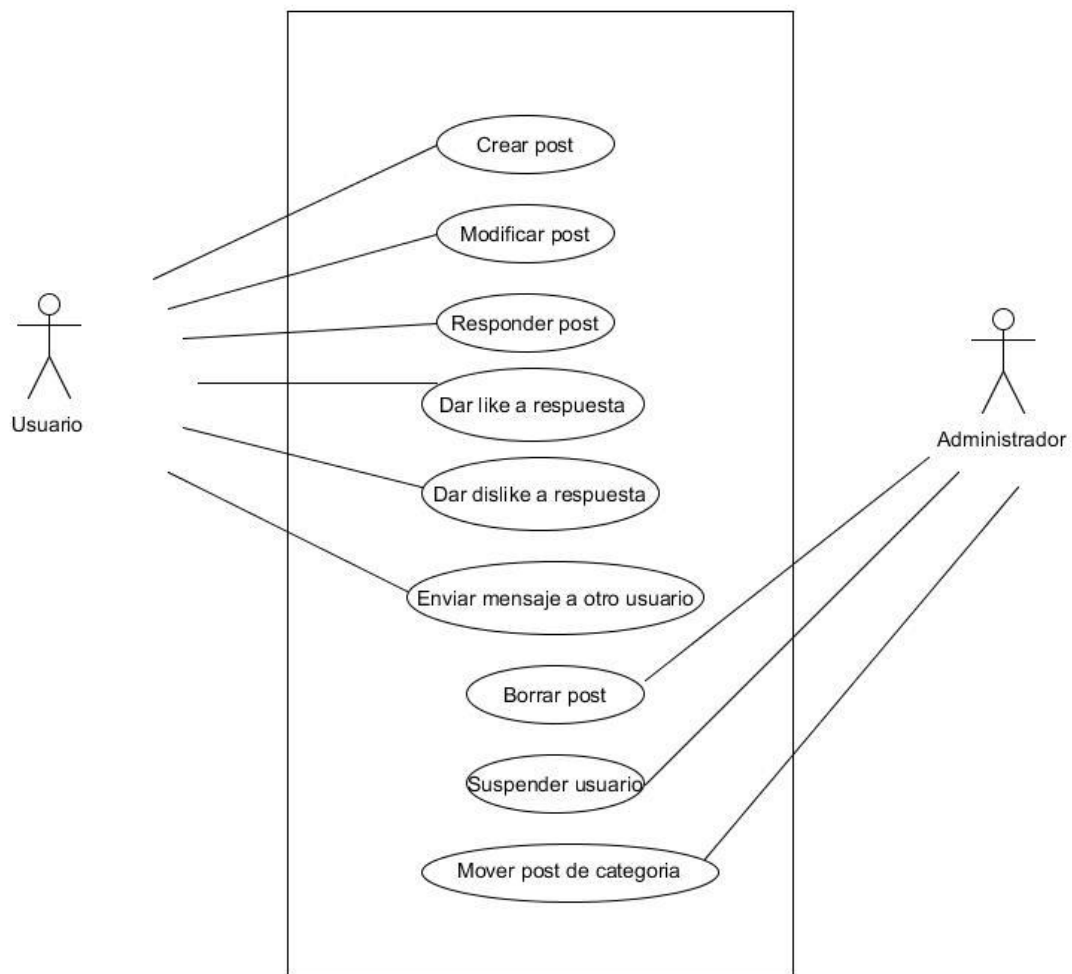
El equipo deberá tener instalado como mínimo Windows 7 y un paquete como XAMPP, que es el que permitirá tener la base de datos y el servidor de aplicaciones.

4.ANÁLISIS

En esta fase se establecerán los requisitos del sistema



*Diagrama de clases



*Diagrama de casos de uso

- **Caso de uso 001:** Crear post
 - Actor principal: Usuario
 - Precondiciones:
 - No hay ningún post creado en la BBDD.

Escenario de éxito:

1. Usuario registrado accede a la aplicación
2. El usuario indica mediante el botón “Crear post” que desea crear un nuevo post.
3. El usuario rellena el formulario y confirma la acción.
4. El sistema creará el post y mostrará la página principal de la aplicación.

- **Caso de uso 002:** Modificar post
 - Actor principal: Usuario
 - Precondiciones:
 - El usuario deberá de crear algún post.

Escenario de éxito

1. Usuario registrado accede a la página de editar.
2. El usuario indica mediante el botón “editar” que desea editar el contenido de un post.
3. El usuario rellena el formulario y confirma la acción.
4. El sistema muestra la página donde se encuentran todos los posts.

- **Caso de uso 003:** Responder post
 - Actor principal: Usuario
 - Precondiciones:
 - Deberá de estar creado algún post en la BBDD.

Escenario de éxito

1. Usuario registrado accede a la página de responder.
2. El usuario indica mediante el botón “responder” que desea responder a un post.
3. El usuario rellena el formulario y confirma la acción.
4. El sistema creará la respuesta y mostrará la respuesta realiza a ese post.

- **Caso de uso 004:** Dar “me gusta” a una respuesta

- Actor principal: Usuario
- Precondiciones:
 - Deberá de existir algún post para poder valorarlo.

Escenario de éxito

1. Usuario registrado accede a la página de una publicación.
2. El usuario indica mediante el botón “me gusta” que desea valorar esa publicación.
3. El sistema realizará la acción de dar “me gusta”.
4. El sistema muestra la publicación con las valoraciones.

- **Caso de uso 005:** Dar “no me gusta” a una respuesta

- Actor principal: Usuario
- Precondiciones:
 - Deberá existir algún post para poder valorarlo.

Escenario de éxito

1. Usuario registrado accede a alguna publicación.
2. El usuario indica mediante el botón “no me gusta” que desea valorar esa publicación.
3. El sistema realizará la acción de dar “no me gusta”.
4. El sistema muestra la publicación con las valoraciones.

- **Caso de uso 006:** Enviar mensaje privado a otro usuario

- Actor principal: Usuario
- Precondiciones
 - Deberá existir el usuario al que se quiere enviar el correo.

Escenario de éxito

1. El usuario que quiere mandar el mensaje se dirige a una publicación que haya realizado el destinatario.
2. El sistema muestra el perfil del usuario destinatario.
3. El usuario que envía el mensaje pulsa en el botón “enviar mensaje” que aparece en el perfil del destinatario

4. El sistema muestra un formulario para rellenar con el contenido del mensaje.
5. El usuario que envía el mensaje es redirigido a la página principal.
6. El destinatario del mensaje accede a su perfil donde le aparecerá una bandeja con todos los mensajes.
7. El sistema muestra la lista de todos los mensajes que tiene ese usuario.

- **Caso de uso 007:** Borrar post

- Actor principal: Administrador.
- Precondiciones:
 - Deberá de existir alguna publicación.

Escenario de éxito.

1. El sistema mostrara todas las publicaciones.
2. El administrador seleccionara la publicación que quiere borrar.
3. El sistema borrar la publicación seleccionada.

El sistema mostrara las publicaciones.

- **Caso de uso 008:** Suspender a un usuario

- Actor principal: Administrador.
- Precondiciones:
 - Deberá existir el usuario al que se quiere suspender.

Escenario de éxito.

1. El sistema mostrara el perfil del usuario.
2. El administrador selecciona la opción "suspender usuario"
3. El sistema suspenderá al usuario seleccionado.
4. El sistema mostrara la página principal.

- **Caso de uso 009:** Mover post de categoría:

- Actor principal: Administrador
- Precondiciones:
 - Deberá de existir algún post para poder moverlo.

Escenario de éxito

1. El administrador accederá a la respuesta que quiere mover.
2. El administrador le dará al botón con el icono “mover”.
3. El sistema realizará la acción de mover la respuesta a la categoría seleccionada.
4. El sistema mostrará esa publicación con todas sus respuestas en otra categoría.

4.1 Requisitos funcionales

Los requisitos funcionales de la aplicación son:

- El administrador podrá realizar todas las operaciones de la aplicación.
- Los usuarios registrados podrán crear nuevas entradas al blog, responder y añadir valoraciones.
- El resto de usuarios solo podrá ver los posts que hay creados en la aplicación.

4.2 Requisitos no funcionales

La aplicación tendrá un diseño funcional y amigable.

También deberá funcionar correctamente en cualquier navegador como por ejemplo Mozilla Firefox, Chrome, Internet Explorer ...

Al mismo tiempo la aplicación estará adaptada para todos los dispositivos como móvil, tablet...

5. DISEÑO

En esta fase se realiza una aproximación al diseño tecnológico de la aplicación.

5.1 Estructura de la aplicación

Como cualquier aplicación web se divide en Back y Front.

- Back-end: Sistema de la aplicación donde se realizan todas las acciones del servidor y se transfieren los datos a la base de datos.
- Front-end: Parte de la aplicación que procesa la información web.

5.2 Componentes del sistema / arquitectura de red

Los elementos necesarios para el proyecto serán, una base de datos donde almacenar la información, un servidor de aplicaciones para poder desplegar la aplicación en este caso se utilizará el paquete XAMPP.

5.3 Herramientas y tecnologías utilizadas.

Las tecnologías y herramientas utilizadas son:

- **Lenguaje**: Java.
- **Servidor de aplicaciones**: Apache Tomcat.
- **BBDD**: MYSQL.
- **ORM**: Hibernate.
- **Otras tecnologías**: JavaScript, CSS3, HTML5, Thymeleaf, JQuery y Bootstrap.

6. IMPLEMENTACIÓN

Partiendo del diseño, en esta fase se construye el sistema. Como se trata de una aplicación de nueva creación no habrá impacto en los usuarios y al ser implantada podrá ser utilizada por los usuarios de forma normal.

6.1 Implementación del modelo de datos

La creación de la base de datos con sus tablas se hará con el ORM Hibernate.

6.2 Carga de datos

Se extraerán los datos necesarios de la base de datos, los cuales se mostrarán en pantalla.

6.3 Configuraciones realizadas en el sistema

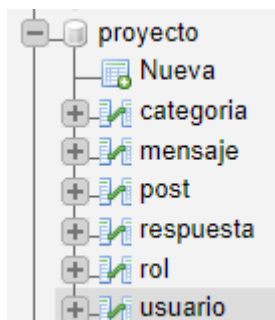
Las configuraciones que se han realizado en el sistema son las siguientes:

- En la base de datos crearemos una nueva estructura y un usuario con las respectivas credenciales que nos permitirá el acceso a la aplicación.
- La aplicación se configura desde el “application properties”, en el que se establecerán los requisitos necesarios al igual que también se reflejara la configuración para el envío de un mail de bienvenida al registrarse un usuario.

```
spring.datasource.url = jdbc:mysql://localhost:3306/proyecto
spring.datasource.driverClassName = com.mysql.jdbc.Driver
spring.datasource.username = root
spring.datasource.password =
spring.jpa.database-platform = org.hibernate.dialect.MySQLDialect
spring.jpa.show-sql = true
spring.jpa.hibernate.ddl-auto = update

spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=correctenglishforo@gmail.com
spring.mail.password=Proyecto1
spring.mail.properties.mail.smtp.starttls.enable=true
spring.mail.properties.mail.smtp.starttls.required=true
spring.mail.properties.mail.smtp.auth=true
spring.mail.properties.mail.smtp.connectiontimeout=5000
spring.mail.properties.mail.smtp.timeout=5000
spring.mail.properties.mail.smtp.writetimeout=5000
```

La estructura que tendrá la base de datos es la siguiente:



6.4 Implementaciones de código realizadas

La aplicación web está desarrollada en Java siguiendo una arquitectura del modelo MVC (Modelo, Vista, Controlador). El acceso a los datos desde la aplicación se ha hecho a través de Hibernate implementando JPA.

En la elaboración de las vistas se ha utilizado HTML5 y CSS3 como base, recogiendo los datos enviados con Thymeleaf. Utilizando JQuery y JavaScript para el manejo de datos y ejecución de código desde el lado cliente.

Con Bootstrap se han diseñado las vistas consiguiendo un resultado responsive para distintos dispositivos.

En este trozo de código se puede observar los servicios que se pueden encontrar en el controlador, como se realiza el registro de un usuario y la persistencia de los datos en la base de datos.

```
@Controller
public class UsuarioController {

    @Autowired
    private UsuarioRepository repoUsuario;
    @Autowired
    private RolRepository repoRol;
    @Autowired
    private MensajeRepository repoMensaje;

    @Autowired
    Mail mail;

    @RequestMapping(value = "/registro", method = RequestMethod.GET)
    public String registro(ModelMap m){
        m.put("listaRoles", repoRol.findAll());
        m.put("view", "");
        return "views/_t/main";
    }

    @RequestMapping(value = "/registro", method = RequestMethod.POST)
    public String registroPost(@RequestParam("alias") String alias,
                               @RequestParam("contrasena") String contrasena,
                               @RequestParam("nombre") String nombre,
                               @RequestParam("primerApellido") String primerApellido,
                               @RequestParam("segundoApellido") String segundoApellido,
                               @RequestParam("telefono") String telefono,
                               @RequestParam("email") String email,
                               @RequestParam("sexo") String sexo,
                               ModelMap m) {

        Rol rolPorDefecto = (Rol) repoRol.getDefaultRol();
        Usuario u = new Usuario(alias, contrasena, nombre, primerApellido, segundoApellido, telefono, email, sexo, rolPorDefecto);
        try {
            if (repoUsuario.datosPerfil(alias) != null) {
                m.put("view", "_t/error");
            }
        } catch (Exception e) {
            m.put("alias", alias);
            m.put("view", "/usuario/crearPost");
            mail.sendEmail(email, "Bienvenido", "<h3>Bienvenido a Correct English.</h3> \n<p> Porfavor confirma tu direccion: http://localhost:8080</p>");
            repoUsuario.save(u);
            System.out.println(e.getMessage());
        }
        return "views/_t/main";
    }
}
```

*Ejemplo del controlador del Usuario.

7. PRUEBAS

Son muchas las pruebas que pueden realizarse en un proyecto para eliminar los posibles errores y garantizar su correcto funcionamiento. Los casos de prueba establecen las condiciones/variables que permitirán determinar si los requisitos establecidos se cumplen o no. A continuación, se detallan algunos de los casos de prueba que se ejecutarán para comprobar la correcta construcción de este proyecto.

7.1 Casos de pruebas

Caso de prueba 1 – Registro de un usuario

- **Descripción:** Registrar un usuario en la aplicación
- **Condiciones de ejecución:** Acceder a la aplicación e introducir los datos en el formulario de registro.
- **Entrada:** Alias, contraseña, nombre, primer apellido, email.
- **Resultado esperado:** Si los datos son correctos, se creará el usuario y se redirigirá a la pantalla principal.
- **Posible causa de error:** El usuario no introduce correctamente los datos.

Caso de prueba 2 – Login

- **Descripción:** Acceso con usuario registrado.
- **Condiciones de ejecución:** Introducir correctamente las credenciales.
- **Entrada:** Alias y contraseña.
- **Resultado esperado:** Si los datos son correctos se permitirá el acceso a la aplicación y se redirigirá a la página principal.

Caso de prueba 3 – Creación de una publicación

- **Descripción:** Crear una publicación en el blog.
- **Condiciones de ejecución:** Se necesita un usuario registrado.
- **Entrada:** título, contenido.
- **Resultado esperado:** si el título y el contenido superan la validación en el lado cliente se creará la publicación y se mostrará la publicación.
- **Posible causa de error:** los campos introducidos no cumplan con los requisitos.

Caso de prueba 4 – Modificación de una publicación

- **Descripción:** Permitir a propietario modificar sus publicaciones.
- **Condiciones de ejecución:** Se necesita al propietario de la publicación.
- **Entrada:** título y contenido.
- **Resultado esperado:** si el título y el contenido superan la validación se modificará la publicación y se mostrará el contenido modificado.
- **Posible causa de error:** los campos introducidos no cumplan con los requisitos.

Caso de prueba 5 – Envío de mensaje entre usuarios.

- **Descripción:** Permitir al usuario registrados él envío de mensaje privados.
- **Condiciones de ejecución:** Ambos usuarios deberán de estar dados de alta.
- **Entrada:** título mensaje y contenido.
- **Resultado esperado:** si los campos cumplen con los requisitos, se enviará un mensaje y se mostrará el perfil del destinatario.
- **Posible causa de error:** los campos no cumplan los requisitos.

Caso de prueba 6 – Actualizar datos del usuario.

- **Descripción:** Permitir a un usuario editar sus datos.
- **Condiciones de ejecución:** El usuario deberá de estar dado de alta en la aplicación
- **Entrada:** nombre, primer apellido, segundo apellido, email y teléfono.
- **Resultado esperado:** Si los campos cumplen con los requisitos y superan la validación en el lado cliente, actualizarán los datos del usuario.
- **Posible causa de error:** los datos introducidos no son correctos.

Caso de prueba 7 – Permitir cambiar contraseña

- **Descripción:** Permitir a un usuario cambiar su contraseña.
- **Condiciones de ejecución:** El usuario deberá de estar dado de alta en la aplicación.
- **Entrada:** contraseña actual, nueva contraseña, confirmación de nueva contraseña.
- **Resultado esperado:** Si los campos cumplen los requisitos, se cambiará la contraseña y se mostrará la página de perfil.
- **Posible causa de error:** las contraseñas nuevas no coincidan.

Caso de prueba 8 – Borrar cuenta de usuario

- **Descripción:** Posibilidad de borrar la cuenta
- **Condiciones de ejecución:** El usuario deberá de estar dado de alta en la aplicación.
- **Entrada:** confirmar contraseña.
- **Resultado esperado:** Si introducimos la contraseña correctamente la cuenta se borrará y también todo lo asociado a ella.

- **Posible causa de error:** La contraseña introducida no es correcta.

Caso de uso 9 – Dar valoración a una publicación

- **Descripción:** Tener la posibilidad de valorar una publicación
- **Condiciones de ejecución:** Deberá de estar creada la publicación la cual queremos valorar.
- **Resultado esperado:** Si la publicación esta creada se le sumara un punto.
- **Posible causa de error:** No hay ninguna publicación creada.

Caso de uso 10 – Dar valoración negativa a una publicación

- **Descripción:** Tener la posibilidad de valorar una publicación negativamente.
- **Condiciones de ejecución:** Ambos usuarios deberán de estar dados de alta
- **Entrada:** título, mensaje de respuesta.
- **Resultado esperado:** Si los campos cumplen los requisitos se enviará el mensaje de respuesta al destinatario.
- **Posible causa de error:** los campos no cumplan los requisitos.

Caso de uso 11 – Responder un mensaje privado

- **Descripción:** Tener la posibilidad de responder un mensaje privado estando registrados.
- **Condiciones de ejecución:** Deberá de estar creado la publicación la cual queremos valorar.
- **Resultado esperado:** Si la publicación esta creada se le sumara un punto.
- **Posible causa de error:** No hay ninguna publicación creada.

Caso de uso 12 – Suspender usuario

- **Descripción:** Tener la posibilidad de suspender a un usuario
- **Condiciones de ejecución:** Deberá de estar creado el usuario y tendremos que estar logueados como administradores.
- **Resultado esperado:** Si el usuario existe y hemos iniciado sesión correctamente se suspenderá al usuario.
- **Posible causa de error:** No hay ningún usuario disponible o no nos hemos loguado correctamente como administradores.

Caso de uso 13 – Activar usuario

- **Descripción:** Tener la posibilidad de volver a activar a un usuario.
- **Condiciones de ejecución:** El usuario deberá de tener la cuenta suspendida.
- **Resultado esperado:** Si el usuario tiene la cuenta suspendida, se le activará de nuevo.
- **Posible causa de error:** El usuario no tenga la cuenta suspendida previamente.

Caso de uso 14 - Mover tema de categoría

- **Descripción:** Tener la posibilidad de mover una publicación de categoría siendo administradores.
- **Condiciones de ejecución:** Deberá de existir la publicación que se quiere mover.
- **Resultado esperado:** Si la publicación existe se moverá a la categoría seleccionada con todas sus respuestas.
- **Posible causa de error:** La publicación no exista o no exista la categoría a la que se quiere mover.

8. EXPLOTACIÓN

La implantación es la fase más crítica del proyecto ya que el sistema entra en producción, es decir opera en un entorno real, con usuarios reales.

8.1 Planificación

- Instalación del SO.
- Instalación del servidor Web.
- Instalación y configuración de XAMPP.
- Configuración de la base de datos (MySQL).
- Configuración del Application.properties para su correspondiente funcionamiento y unión con la base de datos.
- Despliegue de la aplicación.
- Testeo y comprobación del funcionamiento.
- Formación a administradores y usuarios para el uso correcto de la aplicación web.

8.2 Preparación para el cambio

Dado que es una Aplicación Web nueva no se contempla una mala acogida de los usuarios a este nuevo entorno.

Para poder desplegar la aplicación necesitaremos que el cliente configure las aplicaciones expuestas anteriormente o en su defecto que sea el propio cliente el que

nos autorice para realizar dichas configuraciones adaptándonos a los entornos que sean propuestos.

8.3 Manual de usuario

Para comenzar el manual hay que asegurarse de que la base de datos responde correctamente.

En la base de datos habrá que crear manualmente alguna tabla como puede ser la tabla “Roles” en la que hay que indicar dos roles, administrador con el ID 1 y usuario con el ID 2.

También se tiene que crear la tabla “Categoría” que tiene que tener la siguiente estructura.

id	nombre_categoria
1	redacciones
2	curriculums
3	cartas
4	otrosC
5	expresiones
6	gramatica
7	jerga
8	otrosD

Para crear un usuario administrador habrá que hacer lo siguiente:

En el RolRepositoryImpl hay que guardar el usuario como administrador haciendo: `querySetParameter(1, "Administrador")`; como aparece en la imagen. Una vez hecho esto el parámetro “Administrador” hay que dejarlo como “Usuario” para que los usuarios nuevos que registremos se creen con el rol Usuario.

```
@Override
public Rol getDefaultRol() {
    Query query = entityManager.createNativeQuery("SELECT r.* FROM proyecto.rol as r " +
        "WHERE r.nombre_rol like ?", Rol.class);
    //Esto es para crear un administrador
    //query.setParameter(1, "Administrador");

    query.setParameter(1, "Usuario");
    //Rol rol = new Rol ((String)query.getResultList().get(1));
    List<Rol> roles = query.getResultList();
    return (Rol)(roles.get(0));
}
```

Después de todo esto la aplicación funcionará correctamente:

1. Usuarios y acciones permitidas:

- Como administrador:

El administrador se encarga de:

- Mover un post de categoría, borrar una publicación y suspender un usuario.

1. Borrar una publicación

Para borrar una publicación tendremos que ser administradores. Para realizar la acción nos dirigimos a la publicación que queremos borrar. Para ello damos click en cualquier publicación y nos aparecerá el título y el propietario de la misma.



Publicacion de pepe - pepe

adsdas



Una vez le demos al botón el sistema borrar la publicación.

2. Suspender un usuario

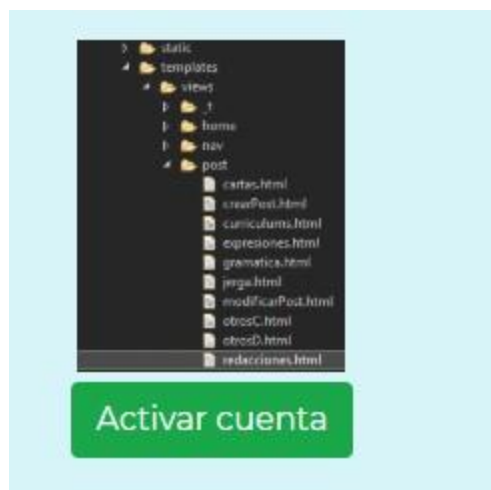
Para suspender a un usuario tenemos que ser administradores. Para llevar a cabo la acción nos dirigimos al perfil de un usuario dando a su nombre en cualquier publicación. Entonces debajo de su foro de perfil nos aparecerá un botón que dice “Suspender cuenta”



Cuando demos al botón nos saldrá un mensaje avisándonos que la acción se ha llevado a cabo correctamente. Entonces desde ese momento el usuario no podrá loguearse en la aplicacacion.

3. Activar cuenta de usuario

Para activar una cuenta de usuario tendremos que hacer todos los pasos iguales que para suspenderla, pero en este caso nos saldrá que un botón que dirá “Activar cuenta”.

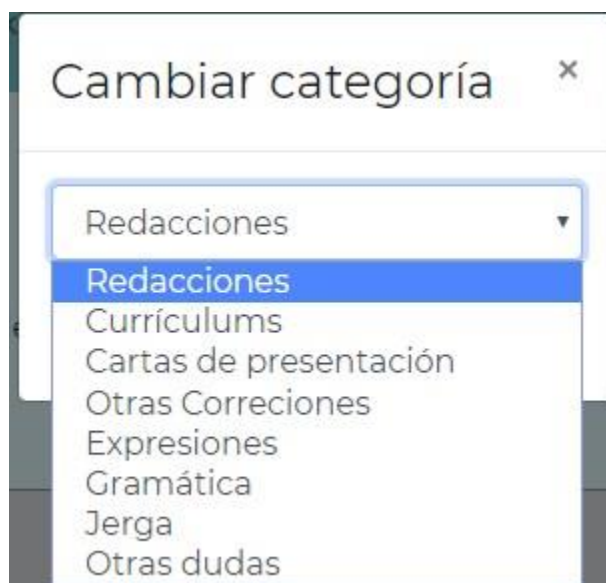


4. Mover publicación de categoría

Para mover un tema de categoría tendremos que tener la sesión iniciada como administradores. Una vez lo hayamos hecho nos dirigimos a la publicación y le damos al icono de mover.



Cuando le demos al botón nos aparecerá una ventana modal con un desplegable, en el que se nos mostrará todas las categorías disponibles.



Una vez tengamos la categoría seleccionada le damos al botón “Confirmar”.

El sistema realizará la acción y mover la publicación con todas sus respuestas a la nueva categoría.

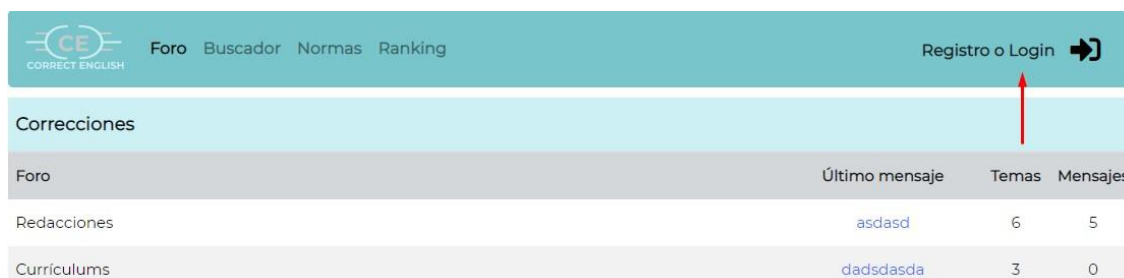
- Como usuario registrado:

Un usuario registrado podrá realizar varias funciones como son:

Crear publicaciones, enviar mensajes a otros usuarios, dejar valoraciones de las publicaciones realizadas, modificar sus propias publicaciones, modificar sus datos.

2. Registro en la página.

Para crearse una cuenta en la aplicación habrá que acceder como usuario anónimo, una vez dentro hay que hacer click en el símbolo de la flecha con la puerta en la parte superior derecha.



Correcciones			
Foro	Último mensaje	Temas	Mensajes
Redacciones	asdasd	6	5
Curriculums	dadsdasda	3	0

CORRECT ENGLISH – DAW

Se abrirá una ventana modal donde nos saldrá un formulario de registro, el cual habrá que rellenar.



A registration form modal with a close button (X) in the top right corner. It features two tabs: 'Inicio Sesión' (selected) and 'Registro'. Below the tabs are three input fields: 'Alias*', 'Contraseña*', and 'Confirmar contraseña*'. Each field has a placeholder text matching its label.

Una vez finalizada esta acción accederemos al correo que hemos facilitado al registrarnos y nos tiene que llegar un correo de bienvenida a la aplicación, una vez realizado este paso ya podremos entrar en la aplicación.

3. Modificar datos de la cuenta.

Para realizar esta acción deberemos de estar registrados previamente en la aplicación.

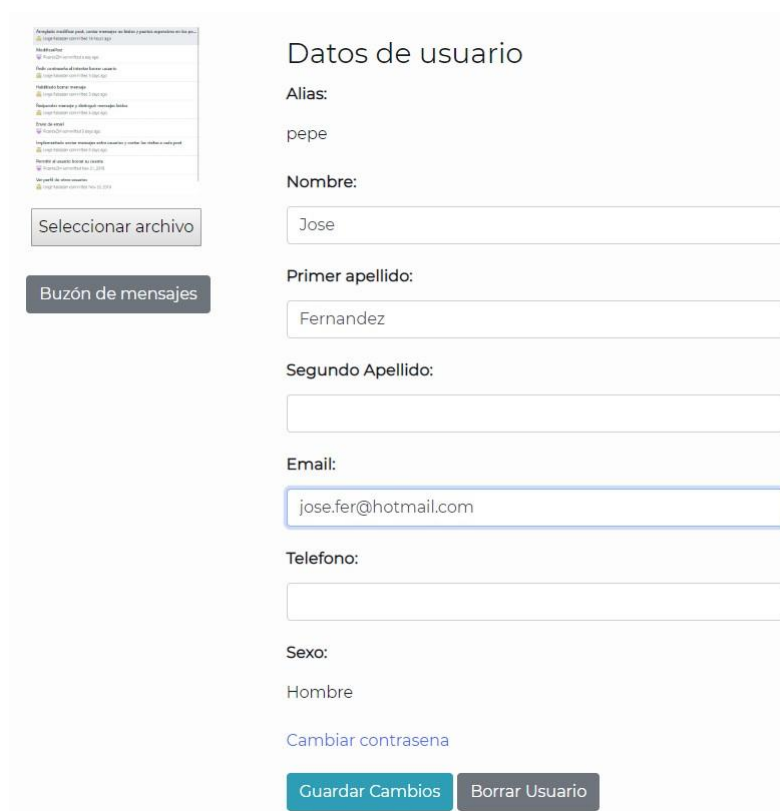
Una vez nos hayamos registrado y accedido a la página nos dirigimos a la parte superior derecha y hacemos click en el icono de la persona.



CORRECT ENGLISH - DAW

Cuando le demos click nos aparecerá nuestro perfil con los datos que se podrán modificar.

Para llevar a cabo los cambios rellenamos los datos que deseamos modificar y le damos al botón de “Guardar cambios”. Una vez hecho esto nos aparecerá el perfil con los nuevos datos ya modificados.



The image shows a web interface for a user profile. On the left is a sidebar with a list of links and two buttons: 'Seleccionar archivo' and 'Buzón de mensajes'. The main area is titled 'Datos de usuario' and contains the following fields and options:

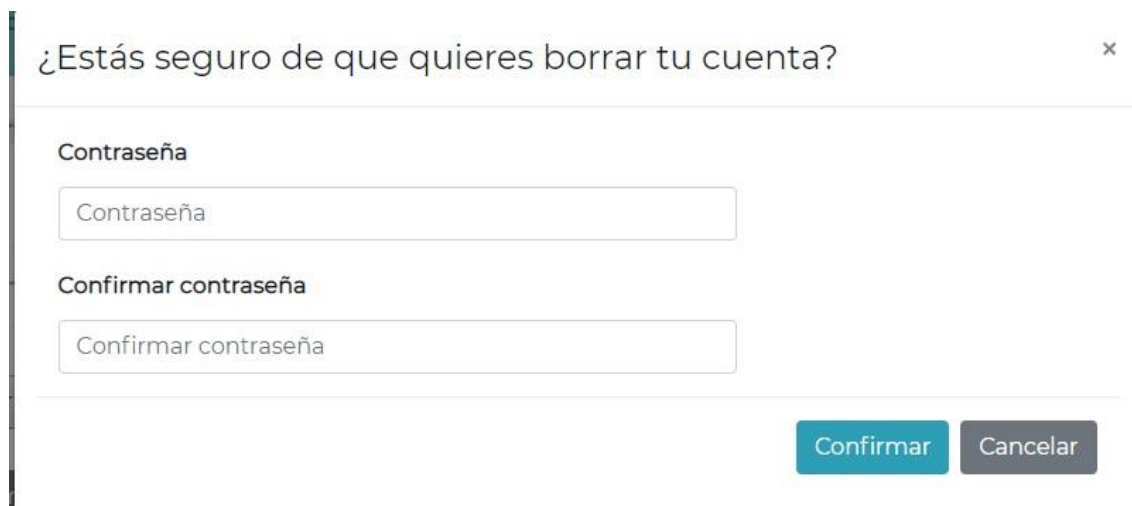
- Alias:** pepe
- Nombre:** Jose
- Primer apellido:** Fernandez
- Segundo Apellido:** (empty field)
- Email:** jose.fer@hotmail.com
- Telefono:** (empty field)
- Sexo:** Hombre
- [Cambiar contraseña](#)
- Buttons:** 'Guardar Cambios' and 'Borrar Usuario'

4. Borrar cuenta de usuario

Para borrar nuestra cuenta accederemos a nuestro perfil desde el icono de la persona, situado en la parte superior derecha.

Cuando demos click en el icono nos aparecerá la información de nuestro perfil y para borrar nuestra cuenta le damos al botón que dice “Borrar cuenta”.

Cuando le demos nos saldrá una ventana de confirmación en la cual tenemos que introducir nuestra contraseña para poder borrar la cuenta.

A confirmation dialog box with a title bar that says "¿Estás seguro de que quieres borrar tu cuenta?". Inside the dialog, there are two input fields. The first is labeled "Contraseña" and the second is labeled "Confirmar contraseña". At the bottom right of the dialog, there are two buttons: "Confirmar" (in blue) and "Cancelar" (in grey).

Si introducimos bien la contraseña se nos borrar la cuenta y todo lo asociado a ella, es decir todas las publicaciones realizadas.

5. Cambiar contraseña

Para poder cambiar nuestra contraseña tendremos que acceder a nuestro perfil (dando click al icono de la persona situado en la parte superior derecha) una vez hayamos iniciado sesión.



Cuando accedamos a nuestra cuenta le daremos click en el enlace que poner "Cambiar contraseña".

Telefono:

Sexo:

Hombre

[Cambiar contraseña](#)



Ahí nos mostrará un formulario en el cual tenemos que poner la contraseña actual y la nueva contraseña confirmándola 2 veces. Una vez confirmada la acción ya podremos acceder a la página con la nueva contraseña.

Cambiar contraseña

Contraseña actual

Contraseña nueva

Confirmar nueva contraseña

Actualizar Contraseña

6. Crear publicación

Para crear una publicación deberemos de estar registrados en la aplicación.

Una vez registrados accedemos con la cuenta y en la página principal nos aparecerán los temas del foro, hacemos click en unos de ellos y nos saldrá un botón con la opción “Nuevo tema”.



Cuando demos a ese botón nos saldrá un formulario para crear el tema, lo rellenamos y el sistema nos mostrará la publicación que acabamos de crear.

A screenshot of a form titled 'Crear Post'. It has two input fields: 'Título:' and 'Mensaje:'. The 'Mensaje:' field is a larger text area with the placeholder text 'Escribe tu mensaje aquí'. At the bottom of the form, there are two buttons: 'Publicar nuevo tema' (in teal) and 'Cancelar' (in grey).

7. Modificar publicación

Para poder modificar una respuesta deberemos ser el propietario de la misma.

Para modificarla accedemos a la publicación realizada y le damos click. El sistema nos mostrará la publicación y si es nuestra nos aparecerá un botón para poder modificarla.

CORRECT ENGLISH - DAW

Le damos click al icono del lápiz con un recuadro y nos saldrá un formulario para modificar la publicación.



Publicacion sobre un tema - pepe

Contenido de la publicacion



Rellenamos los campos y le damos a “Guardar cambios”, entonces el sistema nos mostrará la respuesta ya modificada.

Modificar Post

Título:

Mensaje:

Guardar Cambios

Cancelar

8. Enviar mensaje a otros usuarios

Si queremos mandar mensajes a otros usuarios deberemos de acceder al perfil del propio usuario.

Redacciones	
Tema	Escrito por
asdasd	uuu
uuu	pepe

Para ello buscamos su nombre en cualquier publicación que haya hecho y le damos click. Entonces nos saldrá el perfil de ese usuario y nos aparecerá un botón que dice “Enviar mensaje” debajo de su avatar.



Buzón de mensajes

Datos de usuario

Alias:

pepe

Nombre:

Jose

Primer apellido:

Damos click en ese botón y nos aparecerá un formulario el cual tendremos que rellenar para enviar el mensaje.

Mensaje Privado

Título:

Mensaje:

Enviar Mensaje


Cuando hayamos hecho este paso el sistema mandará el mensaje y aparecerá en la bandeja de mensajes del usuario destinatario. Para ver los mensajes bastará con hacer click en “bandeja de mensajes” y nos aparecerá la lista con todos los mensajes los cuales si queremos podremos borrar una vez los hayamos leído.

9. Responder mensajes privados

Para responder a un mensaje recibido deberemos de estar logueados y acceder al buzón de mensajes desde nuestro perfil.






Una vez hemos accedido no dirigimos a la bandeja de mensajes donde nos aparecerán todos los mensajes que tenemos, entonces hacemos click en el mensaje que queremos responder. Se nos mostrará el contenido del mensaje y le daremos al icono de la flecha para responder. Se nos desplegará un formulario en la parte inferior para responder, una vez rellenado se enviará la respuesta al usuario que queremos. También está la opción de poder borrarlo dando al icono de la papelera.

Mensajes		
Título	De	Borrar
mensaje para pepe	uuu	

10. Ver perfil de otros usuarios.

Para ver el perfil de otros usuarios accedemos a cualquier categoría que tenga post creados. Una vez ahí se nos mostrarán todos los post que se han creado, entonces nos saldrá el nombre de usuario que ha creado el post, le damos click en su nombre y nos aparecerá el perfil de ese usuario.

Redacciones			
Tema	Escrito por	Respuestas	Visitas
tema de maria	 maria	1	23
tema de pepe	 pepe	4	202
tema de juan	 juan	0	6

11. Realizar búsqueda de publicaciones

Para realizar una búsqueda de alguna publicación bastara con dirigirnos a la barra de navegación superior y dar click en la opción que dice “Buscador”, entonces se nos desplegara una barra de búsqueda donde podremos buscar los post con el nombre que queramos.



12. Responder a publicaciones realizadas

Para poder responder a una publicación deberemos de estar registrados. Luego accederemos a una publicación y nos aparecerá un botón de “Responder”.



Tema de maria - maria

contenido tema maria

Luego nos saldrá un formulario para rellenar con la respuesta.

Respuesta

Mensaje de respuesta:

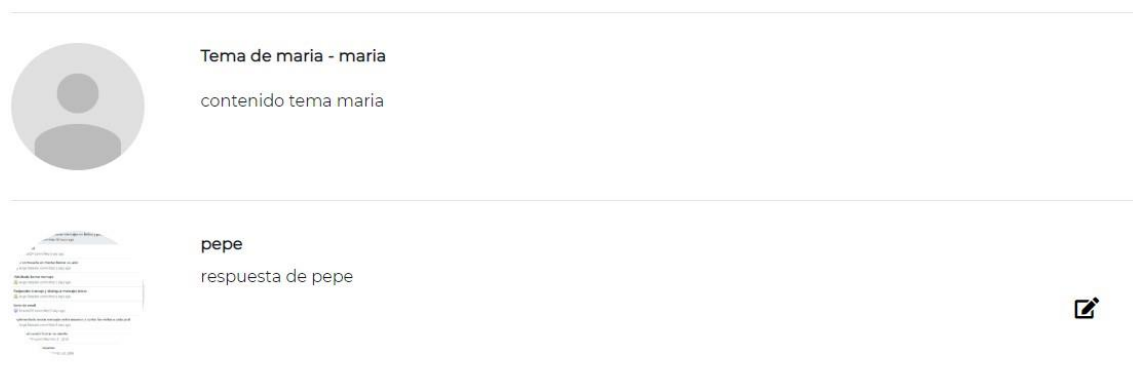
Escribe tu respuesta aquí

Responder

Cancelar

CORRECT ENGLISH – DAW

Una vez que hayamos realizado la respuesta nos saldrá el título de la publicación y las respuestas a la misma.



13. Dar “me gusta” y “no me gusta” a una respuesta

Para poder valorar una respuesta también tendremos que estar registrados. Una vez lo estemos accedemos a una publicación y veremos sus respuestas entonces le damos al icono de la mano hacia arriba para dar “me gusta” y el de la mano hacia abajo para “no me gusta”.



Cuando le demos solo lo podremos hacer una vez y le sumara o restara 1 punto al usuario.

14. Mostrar ranking

Para poder ver el ranking de todos los usuarios y sus respectivas puntuaciones accederemos desde la barra de navegación superior al apartado que poner “Ranking”.



Cuando le demos nos aparecerá el ranking con todos lo usuarios.

Posición	Alias	Nombre	Puntuación
1	pepe	gonzalez	154
2	juan	fernandez	51
3	maria	perez	8

15. Buscador de la página

Para realizar una búsqueda en toda la página bastara con que le demos click a donde poner buscador en la barra de navegación. Cuando le demos nos mostrará un campo para introducir lo que queramos buscar.



Una vez hayamos introducido lo que queramos buscar el sistema nos mostrara la búsqueda.

Mostrar

10

 registros

Buscar:

Búsqueda por "a"

Tema	Escrito por	Respuestas	Visitas
Tema moviles	uuu	1	8
dadsdasda	pepe	0	9

8.4 Implantación propiamente dicha

La prueba de que esta aplicación esta en producción se puede comprobar desde cualquier navegador accediendo a la url:

correct-english.sytes.net:8080

8.5 Pruebas de implantación

Comprobar que la aplicación se comunica correctamente con la BBDD.

Comprobar que el encaminamiento externo es correcto llegando a los servicios web.

9. DEFINICIÓN DE PROCEDIMIENTOS DE CONTROL Y EVALUACIÓN










A lo largo del ciclo de vida del proyecto se producirán cambios e incidencias que deberán controlarse y registrarse. En este proyecto se ha utilizado la aplicación Taiga.io (<https://taiga.io/>), para la gestión de tareas. En esta aplicación cada tarea se le asigna a un miembro del equipo, y son ellos los encargados de ir modificando el estado de esas tareas.



- Para el control de versiones se ha utilizado la plataforma GitHub (<https://github.com/>).

El enlace al repositorio es:
<https://github.com/Jorge96RM/Proyect>

También se ha utilizado la herramienta GitHub Desktop para una mejor visualización del historial de subidas.

Arreglado modificar post, contar mensajes no leídos y puntos suspensivos en los po...
 Jorge Rabadan committed 16 hours ago
ModificarPost
 RicardoZM committed a day ago
Pedir contraseña al intentar borrar usuario
 Jorge Rabadan committed 3 days ago
Habilitado borrar mensaje
 Jorge Rabadan committed 3 days ago
Redponder mensaje y distinguir mensajes leídos
 Jorge Rabadan committed 4 days ago
Envío de email
 RicardoZM committed 5 days ago
Implementado enviar mensajes entre usuarios y contar las visitas a cada post
 Jorge Rabadan committed 6 days ago
Permitir al usuario borrar su cuenta
 RicardoZM committed Nov 21, 2018
Ver perfil de otros usuarios
 Jorge Rabadan committed Nov 20, 2018

*Historial de versiones en GitHub Desktop.

10. CONCLUSIONES

En este proyecto se ha realizado el desarrollo de una aplicación web que servirá como plataforma para que los usuarios puedan debatir sobre todo tipo de temas.

Tras el análisis inicial se decidió crear una aplicación de cero, diseñando y creando la base de datos.

Para la realización de la aplicación se optó por Java como lenguaje de programación principal y una base de datos relacional como MySQL dónde organizar la información.

CORRECT ENGLISH – DAW

Siguiendo una arquitectura MVC para organizar correctamente las clases y funcionalidades de la parte Back-end. Se han unido los servicios con la base de datos utilizando Hibernate implementando JPA.

Para el desarrollo en el lado del cliente web se ha usado el lenguaje más común, JavaScript, con ayuda de jQuery. La parte “visual” se ha desarrollado con HTML5, CSS3 y Bootstrap, buscando un diseño amigable y compatible con distintos dispositivos.

Se considera que el proyecto final ha cumplido los requisitos solicitados por el cliente, desarrollando una aplicación fiable y funcional sin necesidad de que el usuario posea un nivel avanzado para su utilización.

11. FUENTES

11.1 Legislación

DAW

Enseñanzas mínimas: Real Decreto 686/2010, de 20 de mayo (BOE 12/06/2010)

http://pdf/IFCS03/titulo/RD20100686_TS_Desarrollo_Aplicaciones_Web.pdf

Currículo: [Decreto 1/2011, de 13 de enero \(BOCM 31/01/2011\)](#)

http://pdf/IFCS03/curriculo/D20110001_TS_Desarrollo_Aplicaciones_Web.pdf

Definición de procedimientos de control y evaluación:

- * <http://www.xperta.es/es/aquienvadirigido.asp>
- * <http://churriwifi.wordpress.com/2010/04/10/gestion-de-incidencias/>

11.2 Bibliografía

- <https://es.wikipedia.org>
- <https://es.stackoverflow.com/> y <https://stackoverflow.com/>
- <https://spring.io/>
- <https://github.com/>
- <https://www.thymeleaf.org/>
- <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
- <http://hibernate.org/>
- <https://jquery.com/>
- <https://datatables.net/>
- <https://developer.mozilla.org/es/docs/Web/JavaScript>
- <https://www.oracle.com/es/index.html>
- <https://www.w3schools.com/>