**Diagrama de componentes**

**¿Qué función cumple un diagrama de componentes?**

Proporciona una visión general del sistema, sus relaciones y dependencias. Da al desarrollador información sobre si el sistema funciona de forma coherente. Se asignan tareas y funciones y sus interfaces dentro del sistema.

Los diagramas de componentes son más comprensibles ya que expresan los datos de manera sencilla. Permite visualizar de manera clara bloques modulares utilizables en varios puntos de la arquitectura.

**Conceptos básicos de un diagrama de componentes:**

Las partes esenciales utilizados en este diagrama son los:

* **Componentes:** (representa un módulo del sistema, puede ser un “concepto” funcional con todos los archivos que lo componen por detrás ).
* **Artefacto:** (unidad física de información, puede ser un archivo que lo compone).
* **Interfaz requerida:** (gancho abierto, recibe funciones o datos ).
* **Interfaz proporcionada:** (círculo, proporciona funciones o datos).
* **Puertos:** (conexión hacia una parte externa del sistema).
* **Relación:** (conceptos o archivos relacionados).
* **Relación de dependencia:** (concepto o archivo que depende de otro).

**Componentes del proyecto:**

Los componentes representados en el diagrama son desde el inicio de sesión tanto por una cuenta creada en la base datos, como por google, se visualiza el portal noticiero con su artefactos que lo componen e incluso se observan las acciones posibles a realizar en la página del administrador.

**Actor:** Actor externo (usuario) (**agente externo**) “lleva puerto”

**Componente 1:** Administrador de sesión,

**Componente 2:** Base de datos (A2022 Mariqueo) (**agente externo**) “lleva puerto”

**Componente 3:** Usuario

**Componente 4:** Api Google (**agente externo**) “lleva puerto”

**Componente 5:** Portal Noticiero

**Componente 6:** Pagina Administrador

**Componente 7:** Noticia

**Componente 8:** Mensajeria

**Explicación del Diagrama:**

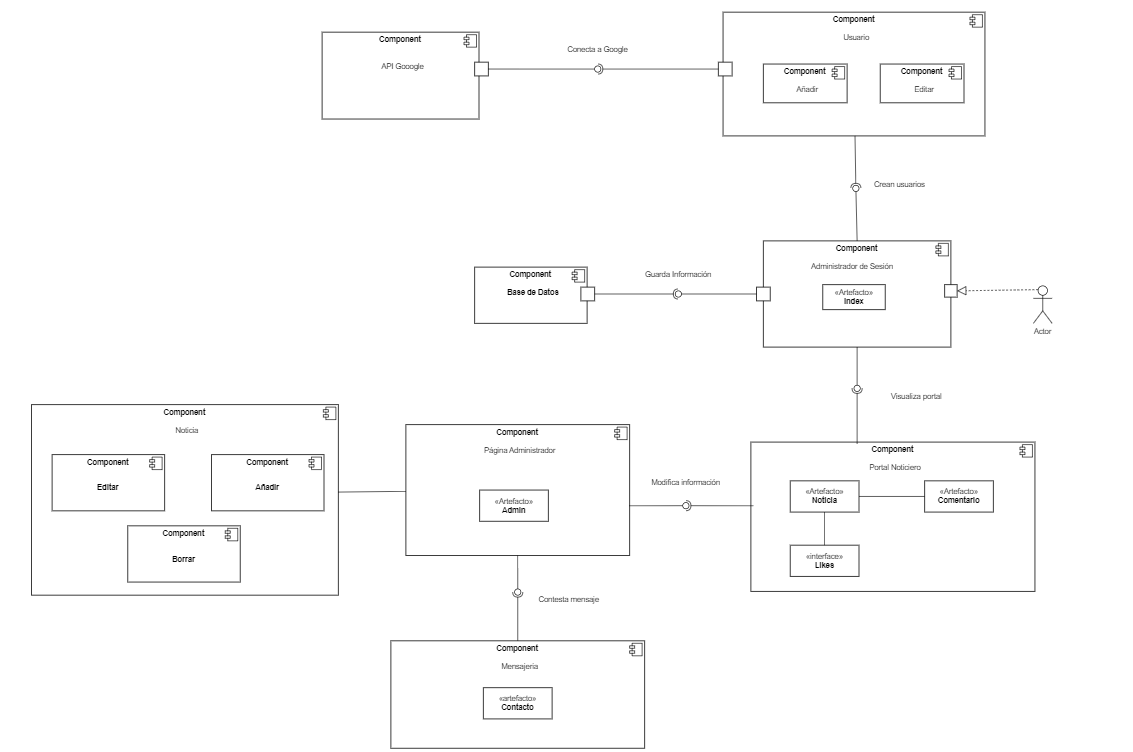
Un usuario a través de su cuenta (ya sea google o normal) al ser un agente externo se conecta por un puerto externo este actor externo (usuario) depende de que exista nuestro administrador de sesión para loguearse, posteriormente se evidencia la base de datos quien recibe datos (base de datos también es un agente externo pues no es parte del sistema) por lo que tiene una interfaz requerida (recibe información).

El administrador de sesión además provee las funciones (interfaz ofrecida “círculo” ofrece función) tanto de añadir como de editar a un usuario.

El componente usuario también puede requerir (interfaz requerida) la funcionalidad de conectarse mediante una cuenta de google por lo que se conecta mediante puertos al componente externo API Google.

El componente portal noticiero, en su interior se pueden observar los artefactos de noticia, comentario y likes. Los 2 últimos relacionados con la noticia.

El componente Pagina Administrador está relacionado con las acciones de editar, añadir y borrar un noticia (componente noticia) Y Además el Administrador provee la funcionalidad (interfaz ofrecida “círculo”) de contestar (a través de un correo) un mensaje pues existe la posibilidad de que un usuario desee contactarse por cualquier motivo relacionado al “Portal de Noticias UCT”



**Diagrama de Integración de sistemas**

El diagrama de sistemas y protocolos ejemplifica el flujo de un usuario y sus datos y peticiones en un sistema web.

Los principales conceptos para entender este diagrama son:

**HTTP:**

HTTP son las siglas de Hypertext Transfer Protocol, es decir, Protocolo de Transferencia de Hipertexto.

**¿Cómo funciona HTTP?**

Cuando escribes una dirección web en tu navegador y se abre la página que deseas, es porque tu navegador se ha comunicado con el servidor web por HTTP. Dicho de otra manera, el protocolo HTTP es el código o lenguaje en el que el navegador le comunica al servidor qué página quiere visualizar.

**Protocolos TCP/IP:**

El Protocolo TCP/IP o protocolo de control de transferencia consiste en un acuerdo estandarizado sobre el que se realiza la transmisión de datos entre los participantes de una red informática.

**Puertos:**

Un puerto de red es una interfaz usada por cada protocolo para intercambiar información dentro y fuera de la red. Su finalidad es permitir que varios programas puedan enviar y recibir información a través de la red en simultáneo.

**Puerto 80:**

Protocolo de transferencia de hipertexto (HTTP) para los servicios del World Wide Web (WWW).

**Puerto 3306:**

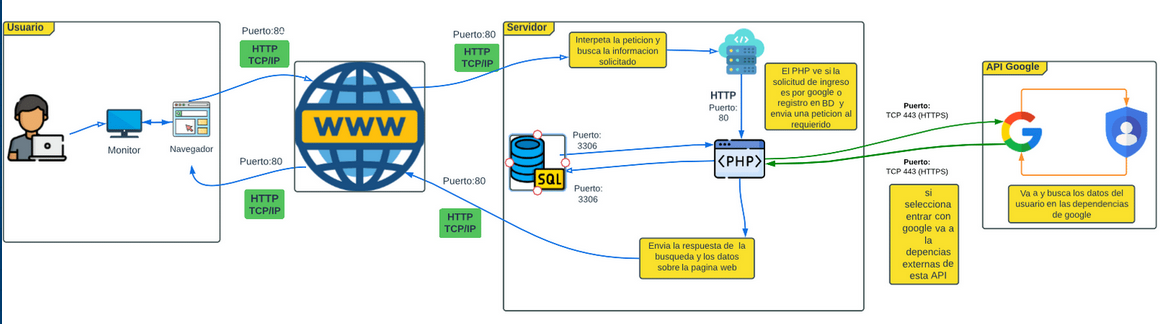
Puerto utilizado por el servicio de base de datos MySQL.

**Explicación del Diagrama:**

Un usuario a través de su navegador ingresa al Portal UCT, lo que se traduce como una petición **HTTP** a través de **WWW** y el puerto 80. La respuesta a esta petición resulta en la visualización del contenido inicial **index.php**.

El usuario ingresa sus datos e intenta loguearse, al hacer esto está haciendo una petición al **servidor** el cual por medio del lenguaje de programación **PHP** se comunica con la base de datos **MYSQL** (**Puerto 3306**). La base de datos envía una respuesta y el lenguaje **PHP** la recibe, viajando de vuelta al usuario en forma de confirmación (Accediendo al **Home**) o de negación por información de inicio de sesión incorrecta.

Este ejemplo muestra sólo uno de los tantos casos de comunicación en nuestro proyecto.



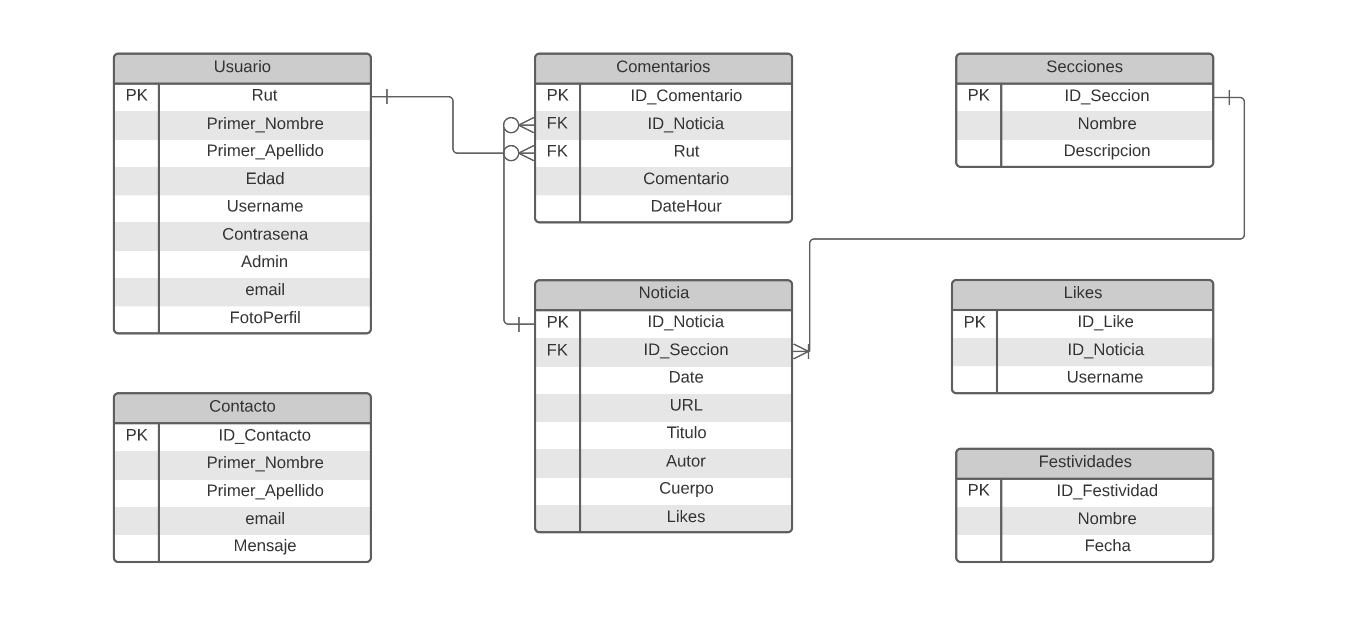
**Diagrama MER (NO FN)**

El diagrama MER se encarga de mostrar las relaciones entre las distintas entidades de un proyecto, a la vez este puede ser utilizado para una fiel representación de una base de datos. En este caso el diagrama fue actualizado en los siguientes aspectos:

**Tabla Contacto:** Esta tabla tiene la función de recibir y almacenar todas las solicitudes de contacto ingresadas a través del Portal.

**Tabla Likes:** Esta tabla funciona como un verificador de existencias, por ejemplo cada vez que un usuario clickee el botón de me gusta su nombre de usuario tanto la ID de la noticia en cuestión serán ingresadas a esta tabla. De esta forma se podrá verificar si el usuario logueado ya reacciono a esa publicación.

**Tabla Festividades:** La tabla festividades almacena fechas importantes a nivel Nacional y mundial, como lo puede ser la Navidad, Fiestas patrias, Aniversario UCT, Año nuevo, etc. Esta información es utilizada para personalizar la página en función de la fecha correspondiente.

****

**Diagrama MER (FN)**

La normalización es un proceso de organización realizado en las bases de datos con el fin de eliminar redundancias, proteger datos y mejor flexibilidad en el manejo de la misma.

Para normalizar nuestro MER se siguieron los siguientes métodos:

**Atomizar datos:** Esto consta en que cada valor de las tablas de la base de datos sean atómicas,de modo que no tengan datos en una misma tabla un ejemplo sería que en una columna que se llama apellidos tengo dos valores que diga apellido\_1,apellido\_2 en cambio al estar formalizada quedarían dos Columnas de apellidos.

**Primera forma normal:** Método que trata que a cada información se le asigne una celda única.

**Segunda forma normal:** En esta forma debe cumplir la primera forma normal como requisito y ahora todas la columnas de una tabla deben depender a una única clave primaria(esto se puede hacer dividiendo en otra tablas y asigna solo una clave a cada columna)

**Tercera forma normal:** Tal como la anterior las tablas deben estar en 1FN y 2FN, y deben quedar con dependencias funcionales, en cierto modo si una columna no depende de la clave primaria se debe sacar aparte en una subtabla o otra

**Diagrama de casos de uso**

El diagrama de casos de uso es una representación visual y que a través de relaciones muestra las acciones y capacidades de distintos entes en un sistema informático.

**Componentes del Diagrama:**

**Actores:**

Los actores son entidades externas que interactúan con el sistema. Puede ser una persona, otro sistema o una organización.

**Casos de uso:**

Los casos de uso son todas aquellas acciones que se encuentran disponibles en

un sistema, página web, aplicación, etc.

Además de las acciones principales se incluyen aquellas que dependen de estas o las que son opciones de las mismas.

**Relaciones:**

**Asociación:** Es el tipo de relación más básica que indica la invocación desde un actor a un caso de uso o viceversa. Dicha relación se denota con una flecha simple.

**Dependencia:** Es un tipo de relación en la cual un caso de uso depende de otro. El cual se denota con una flecha punteada.

**Include:**

Cuando relacionamos dos casos de uso con un “include” estamos diciendo que el primer caso de uso influye en el primero .Es decir, el segundo es esencial para que el primero cumpla su objetivo.

**Extend:**

Establece que un Caso de Uso puede ser extendido con algún comportamiento adicional definido en otro Caso de Uso.

**Explicación del diagrama:**

Como bien dice en el informe un diagrama de casos de uso es todo lo que el usuario puede ver en la página web, en este caso cuando el usuario ingresa al portal de noticias UCT lo primero que va a visualizar seria para que se registre o inicie sesión, luego de eso verá las noticias publicadas en la página y a su vez las distintas secciones en las cuales se clasifican estas noticias también en cada noticias se podrá ver los comentarios y “me gustas” dejados por los usuarios. También por el lado del administrador se puede publicar, editar y eliminar las noticias, por lo que esto solo lo va a poder ver el administrador y no el usuario común.

