

Hasquitz

CFGS Desarrollo de aplicaciones web

Alejandro Ramírez Jiménez

FECHA	Cambios
19/04/2019	Creación del documento y realización hasta el punto 3.6

Índice

1.- Introducción

2.- Estudio de Viabilidad

2.1.- Alcance del proyecto

2.2.- Descripción del Sistema Actual

2.3.- Descripción del Sistema Nuevo

2.4.- Identificación de Requisitos del Sistema

2.4.1.- Requisitos de información

2.4.2.- Requisitos funcionales.

2.4.3.- Otros Requisitos

2.5.- Descripción de la solución

2.6.- Planificación del proyecto

2.6.1.- Equipo de trabajo

2.6.2.- Planificación temporal

2.7.- Estudio del coste del proyecto

3.- Análisis del Sistema de Información

3.1.- Identificación del entorno tecnológico

3.2.- Modelado de datos

3.2.1.- Modelo Entidad-Relación

3.1.2.- Esquema de la base de datos

3.1.3.- Datos de prueba

3.3.- Identificación de los usuarios participantes y finales

3.4.- Identificación de subsistemas de análisis

3.5.- Establecimiento de requisitos

3.6.- Diagramas de Análisis

1.INTRODUCCIÓN

Consistirá en una aplicación que te permitirá crear tus propios test tan populares hoy en día como los de BuzzFeed (aunque no dan la posibilidad de hacerlos tú mismo).

Los explicaré brevemente, son test bastante particulares con títulos atrayentes, por ejemplo: “Descubre cómo de cerca estas de ser un mendigo según tu forma de ducharte” o un gran clásico como “Qué Pokémon inicial eres” o por supuesto “Responde estas 5 preguntas y sabrás con que Supernena tienes más afinidad”.

Son test cuyo propósito principal es ser absurdo y no tener una relación y veracidad demasiado grande, pero igualmente algo tiene además de que consiguen ser adictivos.

En la aplicación tendremos una gestión de usuarios, gestión de test y sistema de clasificación y puntaje de test. Lo principal de esta aplicación es que los usuarios puedan crear sus propios test mediante una interfaz intuitiva y veraz, en la que diseñen las preguntas, las respuestas, y soluciones, y que sean capaces de configurar cada respuesta y la afinidad que tiene con cada solución para así ir concretando poco a poco la solución en la que acabará la persona que haga el test.

También incluirá posibilidad de seguimiento.

2. ESTUDIO DE VIABILIDAD

2.1. Alcance del proyecto

El objetivo general del proyecto será la realización de una página web.

En primer lugar, abordaremos el análisis del problema y en función de los requerimientos obtenidos se elaborará el diseño de la solución más óptima. A continuación, se procederá con la fase de codificación y finalmente la subida de la web a un dominio.

La aplicación está destinada a un público joven, con un hambre voraz de la curiosidad y en búsqueda del entretenimiento, a su vez, que pueda convertir en algo que compartir con sus amistades.

La duración estimada del proyecto es de 3 meses teniendo como plazo límite el día 19 de junio de 2019, fecha en la que la versión definitiva de la web debe estar totalmente terminada.

2.2. Descripción del sistema actual

El sistema actual, más representativo sería el de la web BuzzFeed, en el que tienen diferentes departamentos según el país.

En cada departamento contratan ciertas personas denominadas “Staff” que son los que se encargan de realizar los quizzes para los visitantes de la web, es decir, se encarga la propia web de hacer los quizzes y los visitantes no tienen ninguna forma.

En estos, simplemente entras, buscas, y lo haces, no tiene sistema de clasificación ni poder seguir usuarios ni nada.

2.3. Descripción del sistema nuevo

El nuevo sistema que planeo implementar, será una web, mitad lo que sería BuzzFeed, mitad red social, en la que pueda haber interacción entre los propios usuarios, puedan votar y en cierto punto puedan autogestionarse. Y lo más importante, posean una interfaz con la que puedan crear sus propios Quiz y publicarlos.

Se intentará dar solución a los diferentes problemas existentes de la siguiente forma:

Ficha de usuarios: La aplicación permitirá el registro de nuevos usuarios

Creación de Quizzes: La aplicación deberá otorgar una interfaz intuitiva para que los nuevos usuarios sin conocimiento alguno sean capaces de crear quizzes.

Buscador de usuarios: La aplicación deberá permitir poder buscar otros usuarios

Explorador de Quizzes: La aplicación deberá mostrar los quizzes que el usuario desee según las personas que sigue, o lo más populares.

Los datos recogidos serán almacenados en una base de datos interna de la aplicación y podrán ser exportados tanto en formato

La persistencia de los datos se hará mediante el sistema gestor de bases de datos relacional SQLite que permite bases de datos de hasta 2 TB de tamaño.

Como herramienta de desarrollo se usará el entorno Visual Studio Code, que proporciona terminales de comandos que funcionan bien con los lenguajes a utilizar. Este software es publicado de forma gratuita bajo licencia Apache 2.0 para plataformas Microsoft Windows, GNU/Linux y Mac OS X.

2.4. Identificación de requisitos del sistema

A continuación, se describen los requisitos identificados en el sistema clasificados en las siguientes categorías:

- Requisitos de información: describen qué información debe almacenar el sistema.
- Requisitos funcionales del usuario: definen la naturaleza del funcionamiento del sistema desde el punto de vista de la interacción con el usuario.
- Requisitos funcionales de la aplicación: definen la naturaleza del funcionamiento del sistema independientemente de la interacción con el usuario.

2.4.1 Requisitos de información del sistema

RI_01 - Usuarios	
Descripción	La base de datos almacenará la información de cada usuario que se registre en la web.
Datos específicos	Id, nombre, correo, contraseña y avatar
Volumen de información	Tantos como personas registradas haya. Puede aumentar constantemente.
Observaciones	

RI_02 - Quizzes	
Descripción	La base de datos almacenará la información de un Quiz creado por un usuario
Datos específicos	Id, creador, titulo, contenido(todas las preguntas,soluciones,respuestas y afinidades), estrellas, fecha de creación,publicado(si está público en la web), privado(si tiene valor, será la contra para acceder al quiz)
Volumen de información	Tantos como quizzes creados haya. Puede aumentar constantemente.
Observaciones	El campo contenido es un objeto del modelo Quiz que lleva las preguntas, soluciones, y posibles respuestas

RI_03 - Seguimientos	
Descripción	La base de datos almacenará la información de los seguimientos, que es cuando un usuario sigue a otro para ver sus quizzes en su timeline

Datos específicos	UsuarioOrigen,UsuarioDestino
Volumen de información	Tantos como seguimientos haya
Observaciones	

RI_04 - Votaciones	
Descripción	La base de datos almacenará la información los votos que ha recibido cada quizz
Datos específicos	Id, origen, quizz, cantidad(1-5)
Volumen de información	Tantos como votos se hayan realizado.
Observaciones	El campo quiz es una referencia al código del Quizz

RI_05 - Moderación	
Descripción	La base de datos almacenará información sobre las acciones de moderación sobre los Quiz
Datos específicos	quizz,usuario,decisión
Volumen de información	Tantos como quizzes estén pendientes de publicación
Observaciones	Decisión reflejar si el usuario que ha moderado, lo ha aprobado o rechazado

RI_05 - Notificaciones	
Descripción	La base de datos almacenará información sobre las notificaciones que los usuarios reciban y si las han leído
Datos específicos	id,usuario,mensaje,leido
Volumen de información	Tantos como notificaciones hayan recibido los usuarios
Observaciones	Las notificaciones que existen por ahora son si les han publicado o borrado un Quiz

2.4.2. Requisitos funcionales del sistema

RF_01 – Crear nuevo usuario	
Descripción	La web permite que un visitante pueda registrarse como un nuevo usuario.

RF_02 – Iniciar Sesión	
Descripción	La web permite que un visitante inicie sesión con su perfil.

RF_03 – Modificar perfil	
Descripción	La web permite que un usuario pueda modificar su nombre,avatar, y contraseña

RF_04 – Crear Quiz	
Descripción	La web permite que un visitante registrado pueda crear Quizzes.

RF_05 – Borrar Quiz	
Descripción	La web permite que un visitante registrado pueda borrar un Quiz que él haya creado.

RF_06 – Realizar Quiz	
Descripción	La web permite que un visitante registrado pueda realizar un Quiz y obtener su resultado.

RF_07 – Votar Quiz	
Descripción	La web permite que un visitante registrado que haya realizado un quiz pueda puntuar el quiz que ha hecho.

RF_08 – Seguir/Dejar de seguir a un usuario	
Descripción	La web permite que un visitante registrado pueda seguir a otros usuarios y ver sus quizzes en la timeline

RF_09 – Moderar Quiz	
Descripción	La web permite que un visitante registrado moderar un Quiz decidiendo si lo aprueba o rechaza, aunque no se publicará hasta que el administrador lo decida.

RF_10 – Privatizar/Desprivatizar Quiz	
Descripción	La web permite que un visitante registrado pueda poner en privado un Quiz del que sea creador

2.4.3 Otros requisitos

Hosting
El servicio de hosting contendrá la aplicación y la base de datos

Dominio
Se tendrá que contratar un dominio para poder acceder a los servicios alojados en el hosting.

2.5.- Descripción de la solución

Se utilizará NodeJS, Angular, y Express(API REST) para la aplicación en sí.

La solución propuesta es en un hosting, tener una máquina ejecutando nuestra aplicación y a la vez la base de datos(MARIADB). Además de esto usaremos el servicio de Storage de Firebase de Google para las imágenes de los perfiles y quizzes.

2.6. Planificación del proyecto

2.6.1. Equipo de trabajo

La aplicación tiene como objetivo servir de proyecto fin de curso para el CFGS Desarrollo de Aplicaciones Web impartido en el IES Julio Verne de Sevilla. Todo realizado por Alejandro Ramírez Jiménez.

2.6.2. Planificación temporal

Se ha fijado un plazo de 2 meses y medio para la realización del proyecto desde la entrega del informe de anteproyecto inicial hasta la entrega y defensa del proyecto terminado.

Durante este periodo el proceso de desarrollo se ha descompuesto en distintas tareas programadas en el tiempo según el modelo como se describe a continuación

Fecha	Estado
22/03/2019 a 24/03/2019	Diseño y creación de la base de datos y el modelo de datos y comienzo del desarrollo web
24/03/2019 a 19/04/2019	Todos los objetivos principales del desarrollo realizados y comienzo de la documentación y análisis
19/04/2019 a 21/04/2019	Documentación y análisis terminado

Los principales hitos durante el proceso de desarrollo son los siguientes:

Fecha	Hito
22/03/2019	Entrega de la documentación de anteproyecto
26/04/2019	Entrega de la documentación de análisis
¿?	Entrega del proyecto

2.7. Estudio del coste del proyecto

Para la estimación del coste del proyecto se emplea la técnica LDC o `Líneas de código` basada en el tamaño del producto. Esta técnica trata de definir el tiempo y el costo del proyecto en base a la cantidad de líneas de código, el costo por línea y la tasa de desarrollo de líneas de código en función del tiempo.

Para la estimación es imprescindible hacer una descomposición funcional del proyecto y en este caso se toma como referencia la definición de subsistemas identificados en función de los requisitos funcionales del sistema que se detallan en el apartado 3.4 del documento

El valor esperado de líneas de código (E) se obtiene como una media ponderada de las estimaciones óptima (A), más probable (M) y pesimista (B).

$$E = (A + 4M + B) / 6$$

/6

Función	Optimista	Más probable	Pesimista	Esperado	Precio por línea	Costo
Gestión de usuarios	700	900	1000	800	,50	883 €
Creación Quizz, realización y puntuación	1300	1500	1600	1400	,50	1450 €
Seguimiento	200	400	500	300	,50	383€
Base de datos y API REST	800	1000	1100	900	,50	924€
Interfaz del usuario	700	900	1000	800	,50	950€

3. ANÁLISIS DEL SISTEMA DE INFORMACIÓN (ASI)

3.1. Identificación del entorno tecnológico

A continuación, se definen en detalle los distintos elementos de la infraestructura técnica que dará soporte al desarrollo de la aplicación.

3.1.1 Hardware

Se requiere una única estación de trabajo para las tareas de desarrollo, como mínimo, estas características.

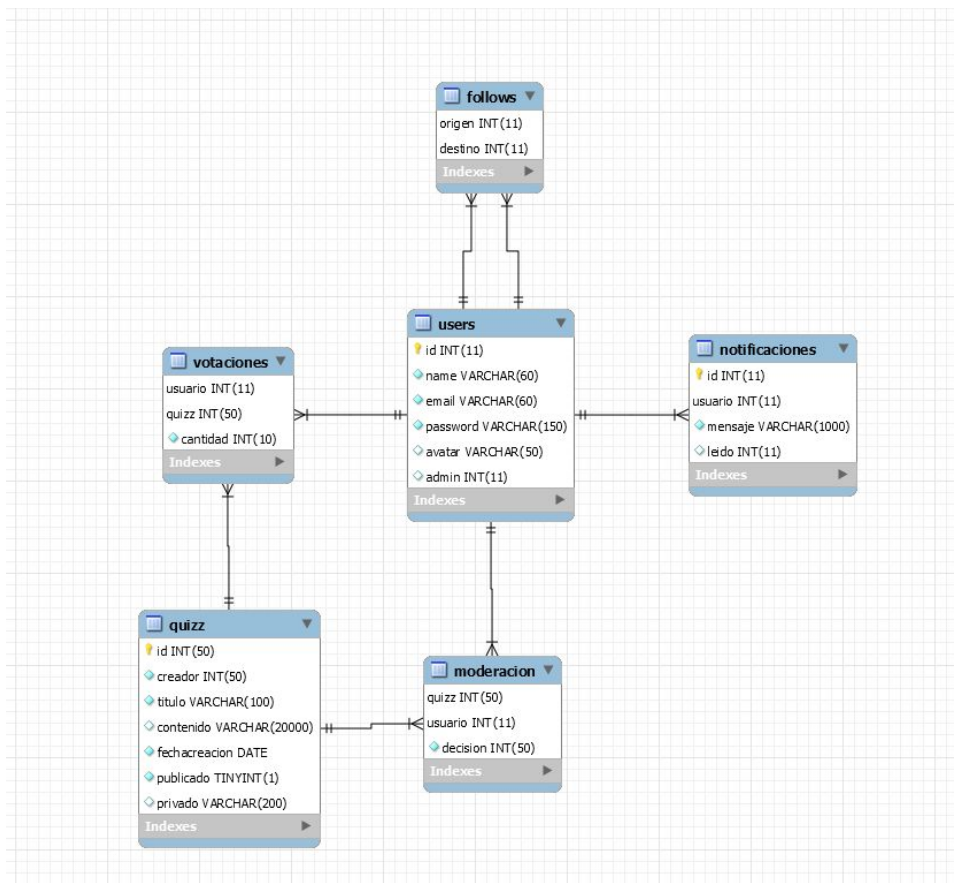
- 8 GB RAM
- I5
- 500GB almacenamiento

3.1.2 Software

- Sistema operativo Windows 10 Home 64 bits
- Visual Studio Code
- NodeJS
- MariaDB

3.2. Modelado de datos

3.2.1. Modelo Entidad-Relación



3.1.2. Esquema de la base de datos y datos de prueba

Se adjunta un fichero llamado buzzflix.sql con la creación de la base de datos.

Se adjunta un fichero llamado datos.sql con la inserción de los datos de prueba.

3.3. Identificación de los usuarios participantes y finales

La aplicación solo dispone de un tipo de usuario, el usuario registrado, que se podrá dar de alta libremente. Este tendrá acceso a todas las funcionalidades de la web explicadas.

3.4. Identificación de subsistemas de análisis

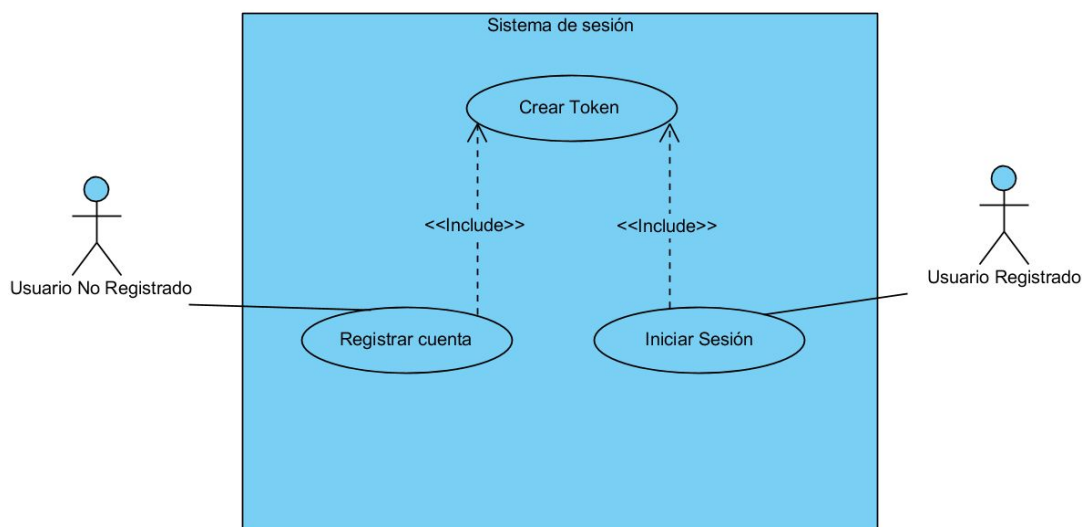
La aplicación estará dividida en los siguientes subsistemas:

- Subsistema de sesión: Agrupa las funcionalidades del registro e inicio de sesión de un usuario.
- Subsistema de Quizz: Agrupa las funcionalidades de creación, realización de Quizzes, su votación y su moderación.
- Subsistema social: Agrupa las funcionalidades de la visibilidad del perfil, edición, búsqueda de otros usuarios y el seguimiento.

3.5 Establecimiento de requisitos

Subsistema de sesión:

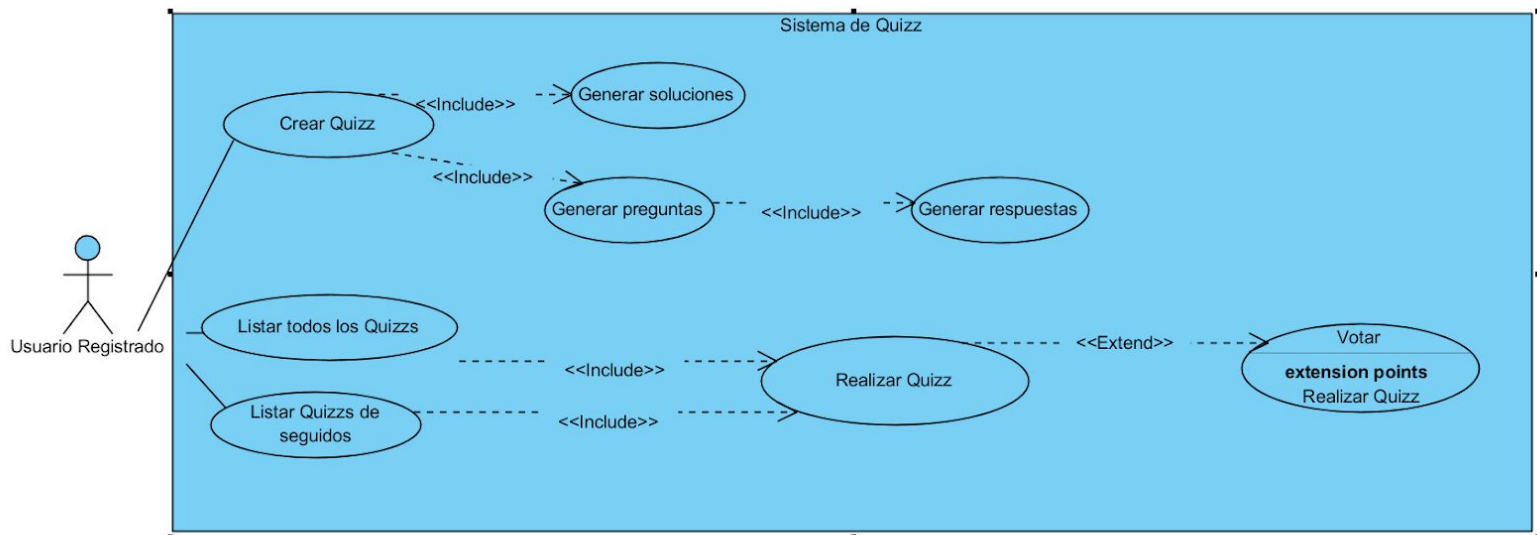
- Registrar: El visitante sin registrar podrá crearse una cuenta introduciendo sus valores y automáticamente será logueado por lo que se crea su token.
- Iniciar Sesión: El visitante introduce ciertos valores que si son correctos le proporcionarán un token y por ende acceso a la web.



Subsistema de Quizz:

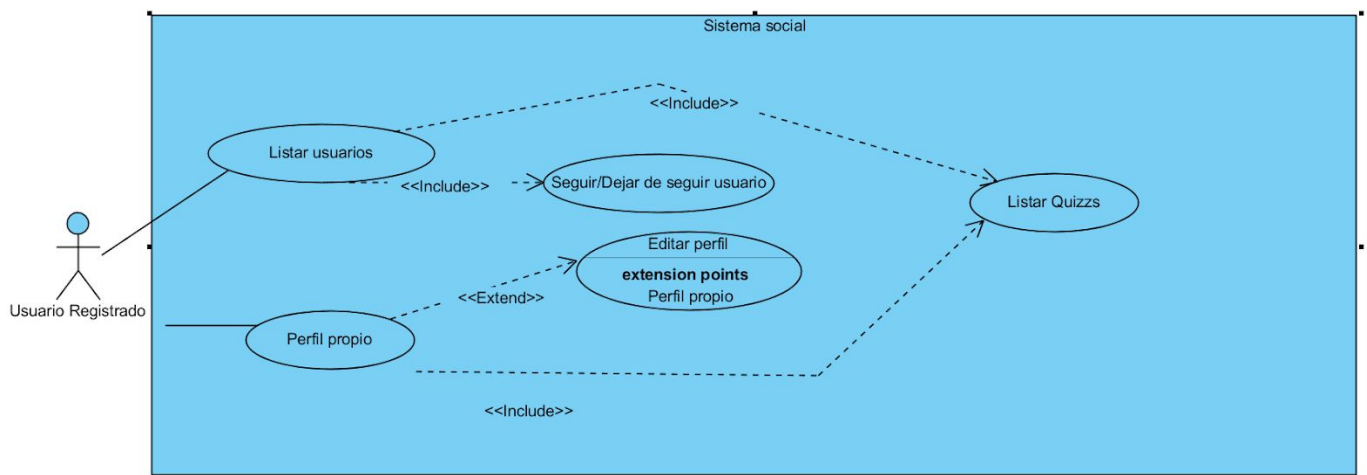
- Crear Quizz: Permite al usuario registrado, poder crear su propio quiz generando sus soluciones, preguntas, y respuestas.
- Listar todos los Quizzes: Permite al usuario registrado, poder obtener una lista con todos los quizzes de la web, ordenados por popularidad.
- Listar Quizzes de seguidos: Permite al usuario registrado, en caso de que tenga usuarios seguidos, obtener sus quizzes ordenados por fecha.

- Realizar Quizz: Permite al usuario registrado, realizar un quiz y obtener un resultado.
- Votar: Permite al usuario registrado, tener la opción de votar un quiz una vez realizado.



Subsistema social:

- Listar usuarios: Mediante una búsqueda filtrada, listará los usuarios con las coincidencias y podrá acceder a sus perfiles.
- Seguir/dejar usuario: Permite al usuario comenzar a seguir a otro usuario ajeno o dejar de seguirlo
- Perfil propio: Muestra al usuario su propio perfil e información
- Editar perfil: Permite al usuario modificar la información de su propio perfil.
- Listar Quizz: Muestra todos los quizzes del perfil en cuestión.



3.6. Análisis de clases

