

Tareas de Diseño de Interfaces

Daniel.

Mi tarea se desarrolló en el sprint1 y fue la de crear el mockup de pantalla de inicio, la primera pantalla visible del juego, para su desarrollo, usé:

- Uso de animaciones, explicadas y documentadas en el propio código
- Una media query para móviles, inhabilitando los botones para no acceder al juego y redistribuyendo los distintos elementos y animaciones para que tuvieran sentido en dicha resolución.
- Búsqueda de imágenes para el desarrollo de las distintas vistas.
- Diseño de icono vectorial para la categoría de tierra y diseño de la carta ruster usada en el juego.
- Mockups con accesibilidad.(Achecker)

Laura.

Realice esta parte en el segundo sprint. Realicé los mockups del CRUD de Categorías, CRUD de Subcategorías y el CRUD de reflexiones. También me encargué de hacer el nav del administrador, usando flex-box. Está distribuido de tal manera que los botones principales tienen un flex-grow mayor y un flex-shrink mayor. Tienen valores 2 los botones principales, valor 1 el cerrar sesión para que aparezca más pequeño y valor 0 el logo para darle aún menor tamaño ya que no lo necesita; está comentado en el css. He usado los colores y tipografías corporativos. Además están hechas varias mediaquery para adaptar los mockup para tablet y móvil y corregir elementos que según se hace más pequeña la pantalla se descolgaba. En el nav está hecha una media query que cuando llega a una resolución menor el nav desaparece y aparece un bocado en la esquina, al pulsar aparece desplazándose desde la izquierda con las opciones.

David.

- Inserción de música que se auto reproduce y está en loop dentro de la pantalla de inicio/juego, mediante el uso de la etiqueta <audio>.
- Creación de mockups:
 - Listado de rankings.

- Alta de puntuaciones.
- Administración preguntas/respuestas. (Alta, listado, modificación)
- Administración de reflexiones (Alta, listado, modificación).

Todos ellos tienen:

- Dos medias queries: para el ajuste de las páginas a dispositivos móviles, en dichas medias queries el contenido de la web se modifica para que todo pueda caber en pantalla, ya sea achicando la letra, o pasando algún elemento a la siguiente línea, modificando el ancho de algún elemento, o en el caso de tablas, ocultando alguna columna.
 - Certificado AChecker: todos ellos han pasado las pruebas de accesibilidad de AChecker (nivel WCAG 2.0 AA).
 - Cajas flexibles: Los mockups de preguntas/respuestas y reflexiones tienen el flexbox creado por Laura en la cabecera de ambos.
- Icono vectorial de aire junto con su memoria gráfica.

Juanma.

SPRINT1 INTERFAZ

He realizado los mockups de la pantalla de juego y de inicio de sesión del administrador. También en el sprint3 creé por DOM de js y le apliqué la query de móvil a las instrucciones para que no salgan en ese dispositivo ya que no se va a poder jugar. He usado porcentajes y em en las letras, también he editado varias imágenes(nubes, sol, luna, botones de comenzar y ranking, icono de la categoría agua...). No he usado flexbox ya que no podía justificar su uso en el sprint 1.

Tareas de Servidor

Daniel.

Para el sprint 3, mis tareas principales fueron las implementaciones del crud completo de reflexiones asociadas a las categorías principales del juego y además el alta de preguntas y respuestas que se utilizarán en el juego.

Para llevar a cabo estos requerimientos implementé el patrón MVC con el elemento de fachada (`php/index.php`), el cual es el encargado de recibir las distintas peticiones http y a partir de él se deriva a los distintos controladores y posteriormente a los distintos modelos.

Por otra parte destacar la funcionalidad de recibir peticiones AJAX con JSON en los bodys de las peticiones que también derivan siguiendo el anterior patrón y responde con un JSON creado a partir de consultas a la base de datos del juego.

La validación que se lleva a cabo tanto en el alta de preguntas como en el de reflexiones es el de los tamaños de la pregunta y reflexión.

El `modelopreguntas.php` y debido a la repartición de tareas, es usado tanto por mi compañera Laura como por mí mismo, usando cada uno nuestros propios métodos creados.

Laura.

Modificación de preguntas y respuestas, borrado de preguntas y respuestas. Validación de servidor si las dos respuestas son iguales, la modificación no se registra. Usado el MVC, creada la fachada (`php/hafacha/fachada.php`) donde se llama a los controladores. Creado el controlador de modificación y además se usa el modelo de preguntas creado por Daniel (`modelopreguntas.php`). A parte se corrigen errores de sprint anteriores como añadir la ventana de confirmación de borrado tanto en preguntas como en subcategorías y visualizar las categorías de cada subcategoría. Se han creado vistas para la modificación y también para el borrado.

David.

Creación de MVC para el inicio de sesión del administrador hacía la parte de CRUDs de la página. De este MVC:

- **Modelo:** Es dónde se realizan todas las consultas necesarias, y contiene todos los métodos necesarios:
 - login: Toma de parámetros el nombre y la contraseña. Y lo que hace es conectar con la BBDD y con una consulta preparada obtiene el nombre del administrador y el hash de la contraseña guardada en la BBDD. Después de comprobar que la consulta devuelva alguna fila, y que el hash de la contraseña de la BBDD y la contraseña pasada como parámetro coincidan, se crea una cookie con 600 segundos de vida, con el nombre del administrador, y con el perfil del administrador (GI)
 - altaAdmin: Para dar de alta al administrador en caso de que no haya uno existente. Toma de parámetros el nombre y la contraseña que tendrá el administrador. Conecta con la BBDD y mediante una consulta preparada inserta en la tabla 'Administrador' un nuevo registro con el nombre, y el hash de la contraseña pasada al método.
 - conectar: Inicia una conexión la BBDD, los datos de conexión se sacan de constantes definidas en un archivo de configuración.
 - checkAdmin: Comprueba mediante una consulta preparada si existe o no un administrador en la base de datos. Devuelve un booleano.
- **Vista:** Las vistas (indexalta.php, indexlogin.php) se comunican con el controlador.
- **Controlador:** El controlador tiene los mismos métodos que el modelo. Cada uno de estos métodos llaman al método respectivo del modelo.

Juanma.

SPRINT2 SERVIDOR

He realizado los CRUDS de Categorías(creación y modificación del icono) y de Subcategorías(creación, modificación, borrado y listado). He aplicado la subida de archivos así como la modificación de los mismos. También en el sprint3 realicé el listado del ranking por php, así como la inserción de los

jugadores en el ranking. No he usado MVC y las validaciones no me dió tiempo a realizarlas.