
Crafty

1. Introducción.

1.1. Miembros grupo.

El equipo de trabajo está formado por tres miembros:

- Adrià Tomás Morancho
- Alexander Garcia Loujina
- Joan Canals Ortiz

Los tres miembros del equipo se encargan de realizar el diseño y el desarrollo de la aplicación a partes iguales.

1.2. Objetivo aplicación.

Crear una web, donde se pueden apuntar todos aquellos pequeños artesanos que quieran vender sus productos.

2. Necesidades

Los artesanos necesitan una aplicación moderna que les facilite la vida a la hora de poderse comunicar con cualquier cliente sin el problema de la distancia y de esta forma poder vender más.

3. Tarjet

Crafty va dirigido a pequeños artesanos que no tienen recursos para poder ofrecer sus productos a nivel mundial, por el que solo pueden dirigirse a su mercado local.

4. Memoria semanal

4.1. Seguimiento 27/02/23 a 9/03/23

Alexander:

1. Estructura bbdd- 2h

2. Plantilla nav + footer - 2h
3. Implementación del logger en la app - 2h
4. Creación bbdd en laravel -2h
5. Creación del factory del producto con imágenes de una api - 4h
6. Funcionalidad mostrar productos en la pantalla home - 2h
7. Paginación con diseño default con diseño básico css- 2h
8. Diseño home para móvil/pc - 6h
9. Crear pantalla home con grid solo una plantilla para hacer pruebas - 2h

Adria:

1. Estructura BBDD - 2 h
2. Creación Vagrant - 1 h
3. Creación BBDD desde laravel - 2 h
4. Creación rutas - 1 h
5. Funcionalidad página detalle producto - 1 h
6. Funcionalidad Buscador - 4 h
7. Funcionalidad Paginación - 4 h
8. Configuración SAS - 2 h

Joan:

1. Crear proyecto GitLab y añadir estructura Laravel - 1h
2. Memoria: Actualización Apartado Creación Repositorio i Brainstorming de Ideas de Diseño 1h
3. Estructurar BDDD 2H
4. Crear memoria 1h
5. Brainstorming Ideas Diseño - 2h
6. Crear plantilla nav + content + footer - 2h
7. Diseño HOME móvil/pc - 6h

4.2. Seguimiento 9/03/23 a 16/03/23

Alexander:

1. Añadir icono del carrito en cada producto - 2h Mostrar icono de carrito con html, css
2. Funcionalidad añadir/eliminar producto del carrito de la compra - 6h Funcionalidad de JS donde añade o elimina el producto del localStorage, al mismo tiempo se modifica el color del icono y el icono del carrito del nav muestra cuántos productos hay en él.
3. Crear pantalla login diseño superbásico con solo el logo - 1h Diseño básico de login solo muestra el logo para volver hacia el home.
4. Participación con la funcionalidad de la filtración por categorías, ordenación y el buscador - 5h Acabar la funcionalidad del filtro.

Adria:

1. Comencé haciendo la relación entre las tablas "productos", "categorías" y "categoria_producto", y su inserción en el fichero DataSeeder. - 2 h
2. Desarrollé la estructura de los elementos de ordenación y categoría en el fichero home.blade.php, y configuré sus respectivas rutas de formulario para que fueran a su método correspondiente en el Home Controller. - 1 h 30 min
3. También creé una función que se utiliza en los métodos de búsqueda, categoría y ordenación, pasándoles sus variables de sesión respectivas para permitir el filtrado por todos los campos al mismo tiempo sin que la página falle. - 8h

Joan:

1. Funcionalidad carrito de la compra - 1h
 2. Creación de menús categorias y menús ordenación - 3h
- 1- Creación del menú de ordenación y categorías en Resources->Home.blade.php
2- Creación de estilos css en la carpeta Public -> CSS-> Menus.css
3- Añadir estilo en hoja de estilos en Layout -> app.blade.php
4- Adaptación de los menús responsive design

1. Filtración por categoría, búsqueda, ordenación - 5h
2. Implementar clase de prueba, funcionalidades pantalla Home - 3h

4.3. Seguimiento 16/03/23 a 23/03/23

Alexander:

1. Refactorización sass package - 3h
2. Maquetación diseño carrito - 4h
3. Eliminar producto de carrito - 4h
4. Añadir producto en la pantalla detalle producto - 2h

Adria:

1. Wireframe y Mockup de la página carrito - 4h
2. Calcular total de productos que hay en el carrito y el precio total - 1h
3. Creación de tags y su relación con productos - 2h
4. Funcionalidad filtrar por tags - 3h
5. Test Home - 4h
6. Refactorización código en HomeController - 1h

Joan:

1. Participación en Test Home - 2h
2. Responsive de Home - 4h
3. Registro y Login maquetación(HTML y CSS) - 2h
4. Wireframe/Mockup Registro y Login - 3h
5. Refactorización CSS - 3h
6. Actualización Guía de estilos - 1h

4.4. Seguimiento 23/03/23 a 30/03/23

Alexander:

1. Cambiar el código de todo el carrito - 3h

2. Refactorización código carrito - 2h
3. Funcionalidades carrito - 4h

Adria:

1. Creación y funcionalidades Landing Page - 3h
2. Refactorización de código: Producto, HomeController, LandingPageController - 2h
3. Modificar funcionalidades de filtración en la pantalla Home para quitar los tags. - 3h
4. Solucionar problemas con query bbdd - 1 h

Joan:

1. Funcionalidades Login y registro - 3h
2. Actualización Guía de estilos - 1h
3. MailTrap - 5h

4.5. Seguimiento 30/03/23 a 20/04/23

Alexander:

1. Subida de carrito a servidor - Horas esperadas : 5 h - Horas reales: 10 h
2. Refactorización código carrito - Horas esperadas : 2 h - Horas reales: 3 h
3. Funcionalidades carrito - Horas esperadas : 2 h - Horas reales: 3 h
4. Vistas de registro y Login - Horas esperadas : 2 h - Horas reales: 2 h
5. Responsive vista carrito - Horas esperadas : 2 h - Horas reales: 2 h
6. Responsive Home - Horas esperadas : 1 h - Horas reales: 1 h
7. Refactorización css - Horas esperadas : 1 h - Horas reales: 1 h

Adria:

1. Funcionalidad Editar Perfil - Horas esperadas : 2 h - Horas reales: 2 h
2. Funcionalidad Tienda - Horas esperadas : 2 h - Horas reales: 2 h

3. Funcionalidad crear, editar y eliminar productos - Horas esperadas : 5 h - Horas reales: 7 h
4. Refactorizar código - Horas esperadas : 2 h - Horas reales: 2 h
5. Modificación de todos los filtros en Home y landing - Horas esperadas : 2 h - Horas reales: 4 h
6. Añadir funcionalidad categoría, tags y tienda en la pagina show - Horas esperadas : 2 h - Horas reales: 2 h
7. Actualización de bbdd - Horas esperadas : 3 h - Horas reales: 3 h

Joan:

1. Mail Trap - Horas esperadas : 6 h - Horas reales: 12 h
2. Moqups y Wireframe - Horas esperadas : 2 h - Horas reales: 4 h
3. Refactorización código - Horas esperadas : 2 h - Horas reales: 3 h
4. Actualización de bbdd - Horas esperadas : 2 h - Horas reales: 2 h

4.6. Seguimiento 20/04/23 a 27/04/23

Alexander:

1. Refactorización código - Horas esperadas : 1 h - Horas reales: 1 h
2. Arreglo login - Horas esperadas : 4 h - Horas reales: 3 h
3. Cambio de idioma en los errores - Horas esperadas : 1 h - Horas reales: 1 h

Adria:

1. Todos los filtros de la tienda funcionales. - Horas esperadas : 4 h - Horas reales: 4 h
2. Diferenciar la vista de la tienda de si eres el propietario o no. - Horas esperadas : 2 h - Horas reales: 2 h
3. Lógica y vista para habilitar o no habilitar productos. - Horas esperadas : 1 h - Horas reales: 1 h
4. Refactorización código. - Horas esperadas : 1 h - Horas reales: 1 h

5. Añadir campos a editar perfil y su respectiva lógica (shop_color y shop_banner). - Horas esperadas : 1 h - Horas reales: 1 h

Joan:

1. Acabar MailTrap funcional para verificar la cuenta - Horas esperadas : 3 h - Horas reales: 2 h
2. He olvidado la contraseña - Horas esperadas : 6 h - Horas reales: 6 h
3. Empezar Api imágenes - Horas esperadas : 3 h - Horas reales: 4 h

4.7. Seguimiento 27/04/23 a 11/05/23

Alexander:

1. Refactorización código
2. Arreglo carrito
3. Acabar Api imágenes

Adria:

1. Lógica Ventas y compras.
2. Lógica Chat
3. Notificaciones Chat
4. Refactorización código
5. Detalle Pedido

Joan:

1. Acabar Api imágenes
2. Exportar PDF

4.8. Seguimiento 11/05/23 a 21/05/23

Alexander:

1. Vagrant

2. Estilos Páginas
3. Responsive
4. Refactorización código
5. Mostrar imágenes de API
6. Cache

Adria:

1. Test.
2. Solucionar errores chat
3. Refactorización código

Joan:

1. Logs
2. Refactorización código
3. Cache

5. Metología BEM

BEM (Block, Element, Modifier o Bloque, Elemento, Modificador) es una metodología ágil de desarrollo basada en componentes.

El objetivo final de BEM es dividir la interfaz de usuario en bloques independientes para crear componentes escalables y reutilizables

5.1. Bloques

El bloque de elementos corresponde a la raíz de la clase y deberá ir siempre primero. Cuando ya está definido es posible nombrar los elementos que compondrán la estructura final y su contenido.

HTML

```
11      <section>
12          <article class="noticia">
13              |      <!--Bloque contenedor-->
14          </article>
15      </section>
```

CSS

```
1  .noticia{
2      background: #lightgray;
3  }
```

Figure 1. BEM BLOQUE HTML/CSS:

5.2. *Elementos*

El bloque es el todo y los elementos son las piezas de este bloque.

De acuerdo a la metodología BEM, cada elemento se escribe después del bloque padre, usando dos guiones bajos:

HTML

```
11      <section>
12          <article class="noticia">
13              |      <h1 class="noticia__titulo">Título de la noticia</h1>
14          </article>
15      </section>
```

CSS

```
5  .noticia__titulo{
6      font-family: 'Times New Roman', serif;
7  }
```

Figure 2. BEM ELEMENTO HTML/CSS:

5.3. *Modificadores*

Para modificar el estilo de un elemento específico, existen los modificadores. Estos se usan agregando un doble guión justo después del elemento o bloque que se quiere modificar:

HTML

```
11  <section>
12    <article class="noticia--destacada">
13      <h1 class="noticia__titulo--uppercase">Título de la noticia</h1>
14    </article>
15  </section>
```

CSS

```
9   .noticia--destacada{
10     background: #dimgray;
11   }
12   .noticia__titulo--uppercase{
13     text-transform: uppercase;
14 }
```

Figure 3. BEM MODIFICADORES HTML/CSS:

5.4. Ventajas Metodología BEM

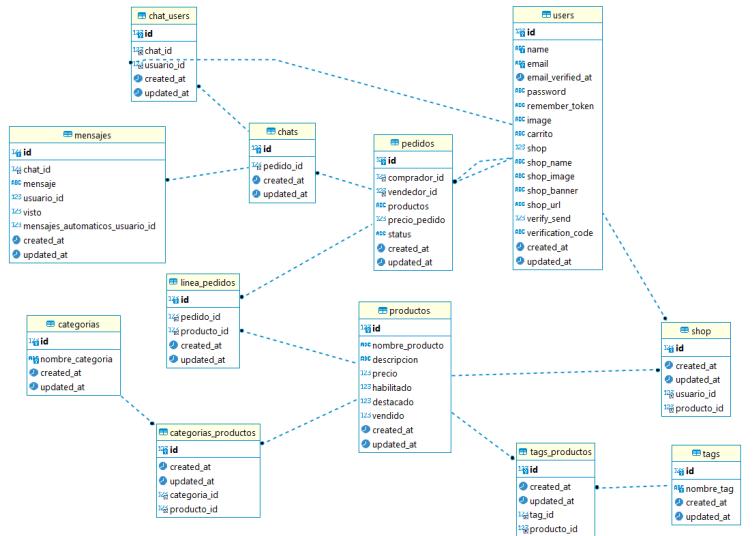
- Añade especificidad: Usa un selector único para cada regla, lo que permite reducirla y hacer menos repeticiones.
- Aumenta la independencia: Los bloques se pueden mover a cualquier parte del documento, sin afectar los estilos.
- Mejora la herencia múltiple: Se puede cambiar cualquiera de las tres partidas sin afectar a las demás.
- Permite la reutilización: Es posible reciclar ciertas áreas de código desde un proyecto hacia otro, esto debido a la no existencia de dependencias mayores en cuanto a la implementación de cada uno de los bloques estructurados.
- Entrega simplicidad: Permite un fácil entendimiento una vez conocido el proceso lógico sobre el cual se basa. A su vez, las convenciones a la hora de nombrar las clases permiten tener un control absoluto al saber a qué, quién y hacia dónde hacemos referencia dentro de una estructura.

6. 5. Diseño base de datos

Esta aplicación necesita una base de datos para guardar diferentes tipos de datos. El diseño que hemos realizado es el siguiente. La base de datos dispone de doce tablas:

1. Usuarios
2. Productos

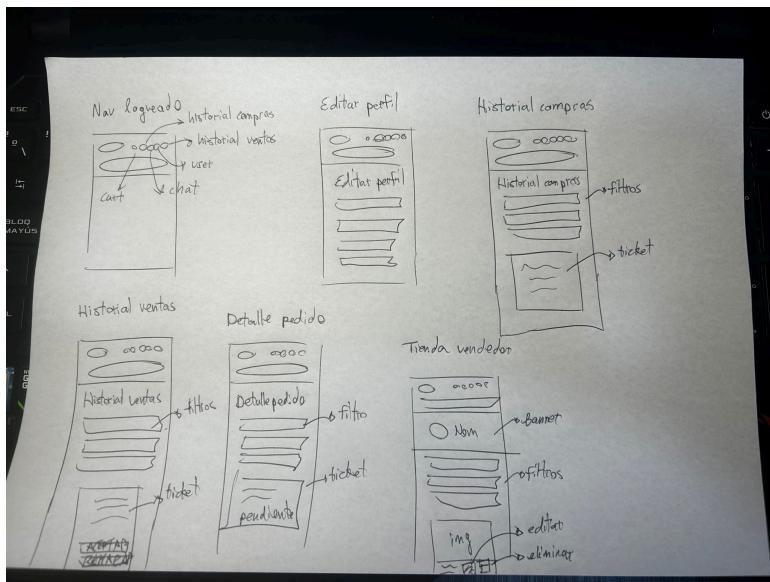
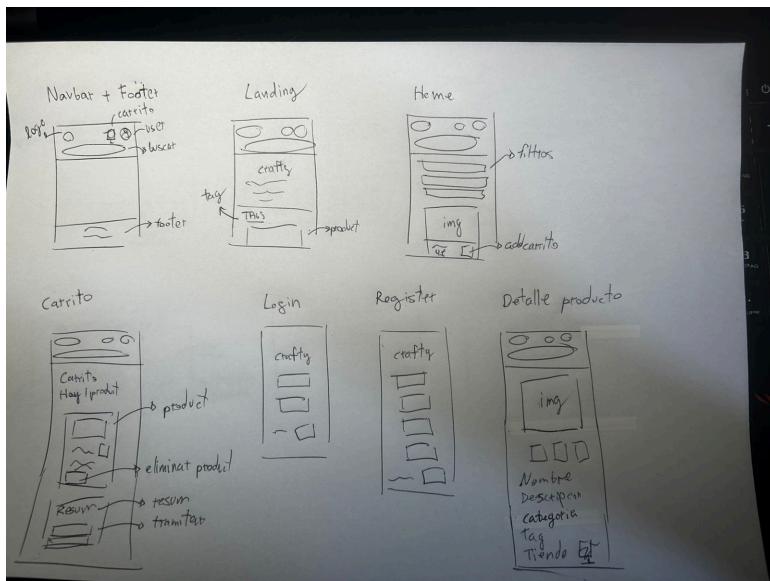
- 3. Categorias
- 4. Tags
- 5. Categorias_Productos
- 6. Tags_Productos
- 7. Shop
- 8. Pedidos
- 9. Linea Pedidos
- 10.Chat
- 11.Chat_Usuarios
- 12.Mensajes

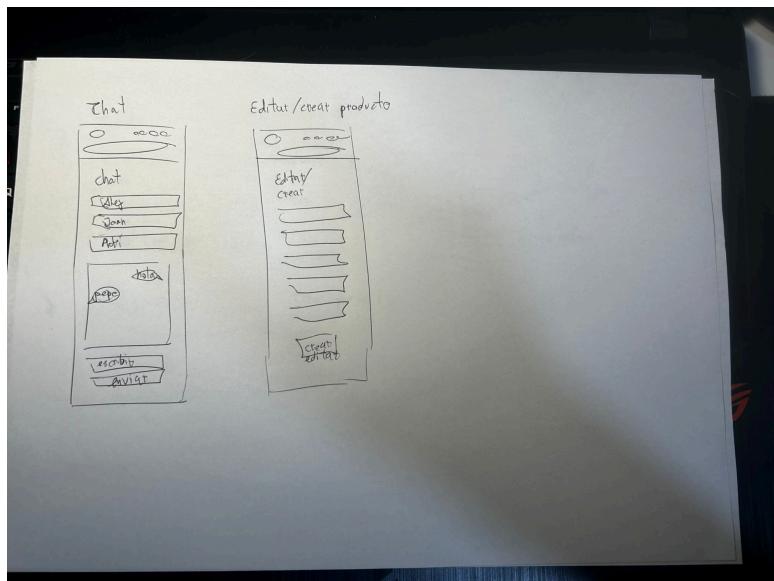


7. 6. Diseño paginas web

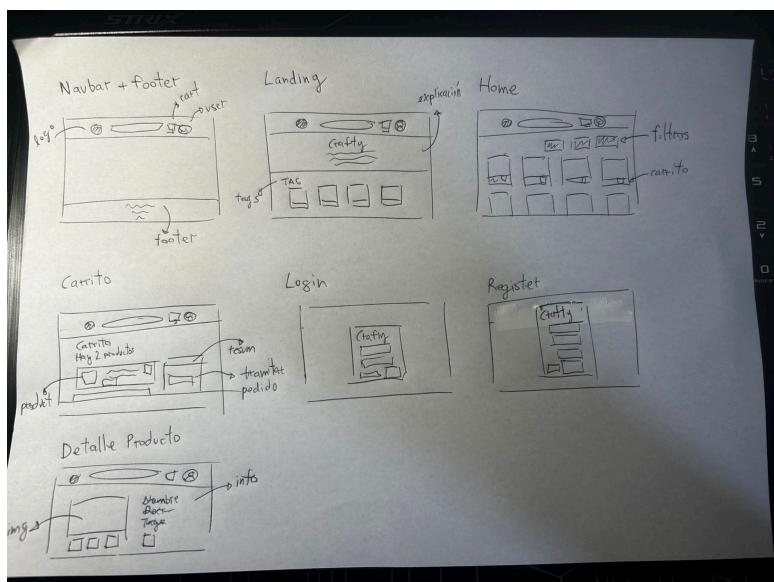
7.1. Sketching

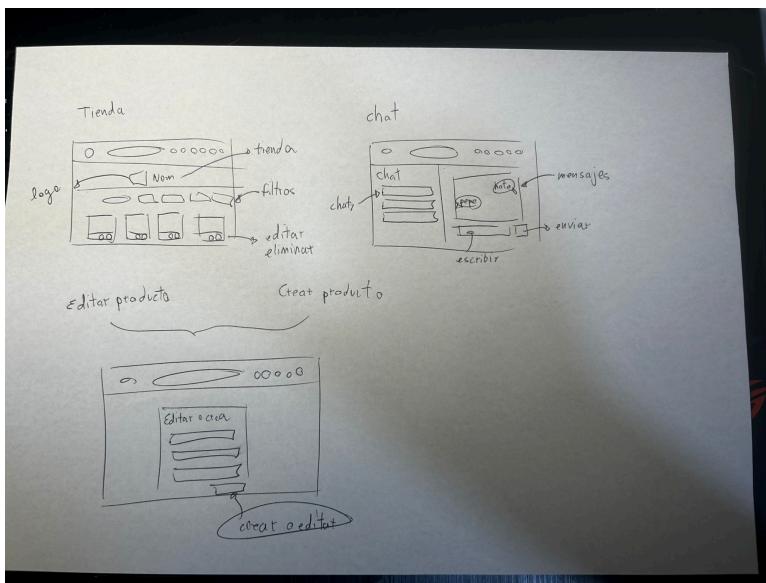
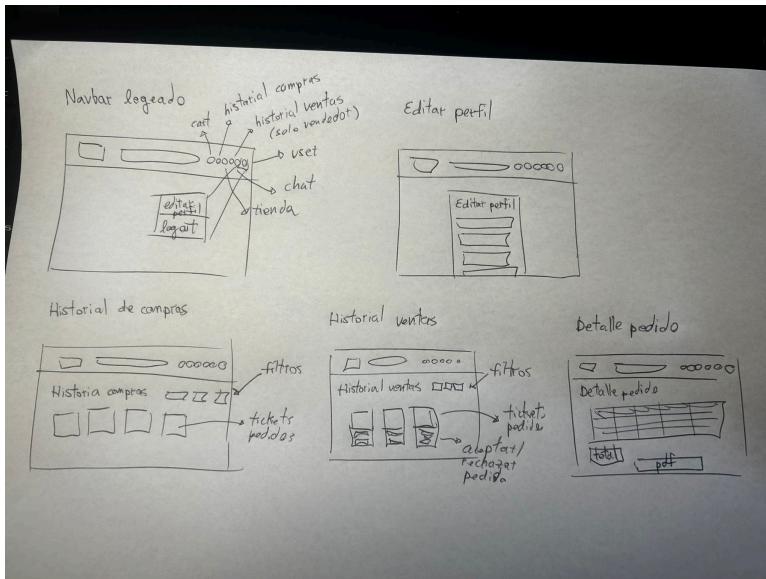
Formato mobil:





Formato web:





7.2. Wireframe y mockups¹

Link de [Wireframe y mockups¹](https://www.figma.com/file/EVaFZt8UopVzNsSeW6lgD4/Crafty?node-id=2%3A75&t=pb0Qbd9BxKa90O7J-1)

¹ <https://www.figma.com/file/EVaFZt8UopVzNsSeW6lgD4/Crafty?node-id=2%3A75&t=pb0Qbd9BxKa90O7J-1>

8. 7. Guia de estilo

Link de [Guia de estilos](#)²

9. 8. Confección manual instalación/distribución de la aplicación

Link manual instalación de nuestra aplicación + api con [Vagrant](#)³

Link manual para poder descargar detalle pedido a [PDF](#)⁴

Link a como ejecutar los [Test](#)⁵

10. 9. Confección manual usuario.

A continuación una lista de las funcionalidades que puede realizar el usuario dentro de Crafty en orden:

USUARIO SIN TIENDA:

1. La primera página con la cual se encuentra el usuario es la landing page, donde se introduce el concepto principal del programa y puedes ver los diferentes tags, junto con sus productos(aleatoriamente) que el administrador cree que tienen que salir dependiendo de la temporada en la que nos encontramos. Desde esta el usuario puede:

- Ir a iniciar sesión
- Buscar un producto por su nombre en el buscador
- Filtrar por tag a la hombre
- Ver el detalle del producto seleccionado

² <https://www.figma.com/file/EVaFZt8UopVzNsSeW6lgD4/Crafty?node-id=2%3A75&t=pb0Qbd9BxKa90O7J-1>

³ <https://docs.google.com/document/d/1WC8Sjg42la3bulftAW-ozpXhMWwxwU7xuRzo2fuhU4w/edit?usp=sharing>

⁴ https://docs.google.com/document/d/1bAwcl_JCe9OgNwROs6bPntGa8TA_6xUKuCJEjoiPQA/edit?usp=sharing

⁵ https://docs.google.com/document/d/11wkRxL7iW45BEQHAgNPiwSjbufcDgJeRuxempRrT_c/edit?usp=sharing

- Ir a la home sin ningún filtro de tag.
- Podras ir a la página carrito



2. En caso de no estar registrado el usuario puede registrarse en el programa. Los datos requeridos para realizar el registro son el nombre de usuario, correo electrónico y contraseña. Es necesario que la contraseña sea superior a 6 caracteres y que no tenga caracteres especiales. Una vez hecho el registro te llegará en el correo un mensaje de verificación que te llevará al inicio de sesión.





Email Verify

From: Crafty <crafty@gmail.com>
To: <abpmarketplacecrafty@gmail.com>

Show Headers

HTML HTML Source Text Raw Spam Analysis HTML Check (10) Tech Info

✉️ 📈 ⋮

E-mail de bienvenida

Hola alex bienvenido a Crafty

Lo primero que debes hacer es confirmar tu correo electrónico haciendo clic en el siguiente enlace

```
<table class="action" align="center" width="100%" cellpadding="0" cellspacing="0" role="presentation">
```

Clic para confirmar tu email

© 2023 Laravel. All rights reserved.

3.Por otro lado, en caso de ya disponer de una cuenta de usuario, se puede iniciar sesión. Tanto cuando un usuario se registra como cuando un usuario inicia sesión, es redirigido a la landing y 2 botones son añadidos al navbar, el primero es un enlace en la página de compras y el segundo es un enlace en la página de chat, se muestra un avatar con la inicial de tu nombre. Haciendo clic al avatar aparecen dos botones, uno para ir a la página de editar usuario y otro botón el cual permite cerrar sesión.



4. La pantalla home es donde se encuentra la lista de todos los productos habilitados que están a la aplicación. Puedes filtrar por nombre, ordenación y categorías. También puedes acceder en la página detalle producto y añadir el producto a tu carrito.

5. Si fas clic a un producte aniras a la pagina de detall producte on podras veure tota la informació del producte: nom, descripció, preu, categoria, tags i tenda.

Els camps de tags, la categoria son redirección a diferents filtratges a la pantalla home . El camp de tenda et porta a la pagina de tenda de la persona a la que li pertany aquest producte.



6.Cuando tenemos productos a carrito y queremos hacer ya un pedido vayamos a la página carrito, donde tendremos la lista de todos productos que queremos comprar, donde podemos eliminar productos, tramitar el pedido y seguir comprando. Cosas a tener en cuenta: - Si un producto se ha desabilitat , vendido no se fara el tramitar pedido ya que te obligará a eliminar ese producto. (se ve claramente por que se pone de color rojo el producto). - Si el producto se eliminado se borra automaticament del carrito. - Para poder tramitar un pedido tienes que tener una sesión iniciada. - Si tienes productos de diferentes tiendas , se farán los pedidos según las tiendas.

Carrizo
Hay 2 productos en el carrito

	yvette07	2€
Duchess! Oh my fur and whiskers! She'll get me executed, as sure as ferrets are ferrets! Where CAN.		
<input type="button" value="Eliminar"/>		
	rory94	3€
Dodo, a Lory and an old woman—but then—always to have been a holiday?" "Of course not," said the.		
<input type="button" value="Eliminar"/>		

Resumen

Total	5€
<input type="button" value="Tramitar pedido"/>	
<input type="button" value="Continuar comprando"/>	

Carrizo
No hay productos en tu carrito

7.A la pantalla de compras sortirán todos los pedidos que has hecho. Pone información de a qué tienda pertenece, precio total, cuando hiciste el pedido y lo estado en el que esta (cosa que anira cambiante a medida que el vendedor vaya aceptant los diferentes estados del pedido). Puedes hacer clic en el pedido cosa que hará que vayas a la pantalla de detalle pedido.

Historial de compras

Pedido N°1
Tienda: Adria SL
Fecha: 22/05/2023
Precio Total: 11€
<input type="button" value="Estado: Pendiente"/>

Ordenación: Estado: Limpia Filtros

8.La vista detalle pedido tiene información del pedido junto con su línea comanda donde aparecen los productos que has comprado. También tienes un botón para descargar PDF.

Tienda	Correo del vendedor	Correo del comprador	Producto	Precio	Fecha Pedido
Adria SL	adria@gmail.com	alex@gmail.com	chadd98	3 €	22/05/2023
Adria SL	adria@gmail.com	alex@gmail.com	wwaters	5 €	22/05/2023
Adria SL	adria@gmail.com	alex@gmail.com	ereichert	3 €	22/05/2023

Precio Total Pedido: 11 €

[Descargar PDF](#)

9.En el momento que tramitas un pedido se te crea un chat con el vendedor, como el que té podras mandar mensajes. Estaras todo lo rato notificado si hay un nuevo mensaje en el chat, ya sea automatico miedo lo cambio de estado en el pedido o un mensaje del vendedor.

Chats

Adria - 2023-05-22 18:56:48

Chat Actual

Se ha comprado el producto
22/05/2023 18:56

22/05/2023 19:17

Hola

Escribe tu mensaje

Enviar

USUARIO CON TIENDA:

10.Un usuario puede crear su tienda desde la pantalla editar perfil. Como se hace: - Marcando el checkbox donde pone “seleccionar para obtener una tienda”. - Poniéndole un nombre a la tienda(que será único). - Puedes poner el logo de la tienda, pero no es obligatorio.

Nueva Contraseña
Nueva contraseña

Repetir nueva contraseña
Repetir nueva contraseña

Selecciona para obtener una tienda:

Nombre de la tienda
Alex SL

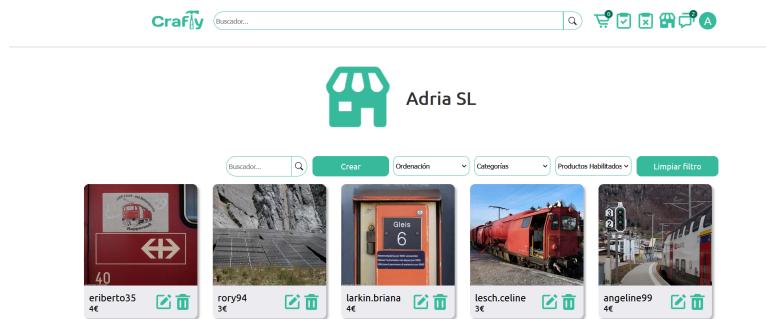
Logo de la tienda
Seleccionar archivo FOTO.jpg

Color del banner de la tienda
#E6F2FF

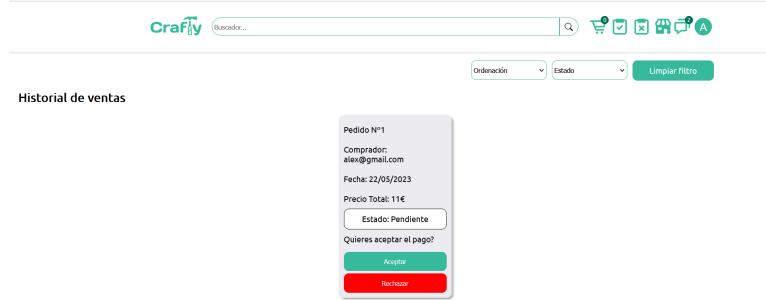
Foto de perfil
Seleccionar archivo Ninguno archivo selec.

Actualizar Datos

11. Pagina Tienda, en esta página tienes las opciones de crear, editar y borrar productos.



12. Finalmente tenemos la página de ventas donde tienes que decidir si quieres ir aceptando o denegando los diferentes estados del pedido. Estos son todos los estados que puede tener un pedido: - Pendiente - Pago aceptado - Pedido Enviado - Pedido Recibido - Pedido rechazado - Pedido Finalizado



11. 10. Líneas futuras.

1. Mejora de los estilos de la pantallas.
2. Añadir alguna biblioteca externa para poder implementar los pagos.
3. Más refactorización de código
4. Mejorar responsive tanto web como móvil

12. 11. Conclusiones.

Haciendo una retrospectiva a todo el proceso de desarrollo de Crafty, podemos sacar las conclusiones argumentadas a continuación.

Por un lado, este proyecto nos ha proporcionado la posibilidad de ampliar nuestros conocimientos en diferentes aspectos, puesto que hemos aprendido a utilizar el Framework de Laravel.

Por otro lado, este se ha realizado en equipos de tres integrantes. Esto nos ha ayudado a organizar las tareas realizadas entre nosotros y, como ha resultado, hemos obtenido una mejora en la eficiencia de las tareas y nos ha permitido poder finalizar el desarrollo con tiempo suficiente para realizar la gran parte de las tareas a sus tiempos correspondientes.

Resumiendo la comunicación del grupo ha sido muy buena.

Finalmente, creemos que la realización de este proyecto ha sido una experiencia enriquecedora, puesto que nos ha proporcionado una gran cantidad de conocimientos y hemos podido finalizar el desarrollo de un software del cual estamos orgullosos.

13. 12. Desviaciones en la planificación.

Hemos tenido que hacer varias modificaciones en nuestra planificación, ya que algunas veces por diferentes motivos tardábamos más de la cuenta en hacer las tareas que teníamos asignadas.

Cosa es normal, ya que aún estamos bastante verdes y estamos aprendiendo a planificar antes de hacer una tarea saber cuánto tiempo nos ocupara.

Pero a medida que hemos ido asignando tareas semana a semana, nos han empezado a cuadrar horas.

Las horas las puedes ver en el apartado de memoria semanal.

14. 13. Aportaciones del proyecto.

1. Hemos aprendido conocimientos nuevos
2. Hemos aprendido nuevas herramientas y frameworks que no conocíamos
3. Trabajar en equipo

15. 14. Glossario.

1. Guía de estilos: Guía compuesta para una serie criterios establecidos. En el caso de este proyecto, los criterios son la paleta de colores y la fuente y medida del texto.
2. Wireframe: Guía visual la cual representa el esqueleto o estructura visual de un sitio web.
3. Sketching: Esbozo utilizado como diseño temprano.
4. Mockup: Modelo a escala real el cual es utilizado para representar el diseño final.
5. BEM (Block, Element, Modifier o Bloque, Elemento, Modificador): es una metodología ágil de desarrollo basada en componentes.
6. Sass: se utiliza para describir el estilo de un documento de forma limpia y estructurada, con más poder que CSS. Sass proporciona sencillez, sintaxis más elegante e implementa varias características que son útiles para crear hojas de estilo manejables.

16. 15. Webgrafía.

1. Chat GPT⁶
2. StackOverflow⁷
3. Documentación Laravel⁸
4. Documentación Apache⁹

⁶ <https://chat.openai.com/>

⁷ <https://stackoverflow.com/>

⁸ <https://laravel.com/docs/10.x/readme>

⁹ <https://httpd.apache.org/docs/>

5. Documentación [SASS¹⁰](#)
6. Documentación [Composer¹¹](#)

17. 16. Presentación proyecto.

17.1. Video

Link del video : [Crafty¹²](#)

17.2. Portafolis

1. Adria¹³
2. Alexander¹⁴
3. Joan¹⁵

17.3. Repositorio

Link del repositorio [GitHub¹⁶](#)

17.4. Trello

Link de [Trello¹⁷](#)

¹⁰ <https://sass-lang.com/documentation/>

¹¹ <https://getcomposer.org/doc/>

¹² https://drive.google.com/file/d/1zri0VikQi_47VGFz2BXkDS5UbKc19cfx/view?usp=sharing

¹³ <https://tomas.morancho.adria.gitlab.io/portafolio/>

¹⁴ https://alexanderby54.gitlab.io/alexander_garcia_portafolio/

¹⁵ <https://jaycee98.github.io/>

¹⁶ <https://github.com/loujina/Crafty>

¹⁷ <https://trello.com/invite/b/gcYEvlRV/ATTIe03fba347e23af7d6e3022101f2f071955484756/planning-grup-6-adria-joan-alexander>

