

Del concepto al Diseño

14 de Septiembre 2024

Fecha de entrega en U-Cursos: **26 de Septiembre 2024**

Fecha de presentación: **28 de Septiembre 2024**

VISIÓN GENERAL

La transición de un diseño conceptual a una realidad funcional es un paso crucial en el proceso de desarrollo. Para garantizar que los equipos trabajen en sintonía, es fundamental que todos los involucrados tengan claridad y acceso a los diseños y las especificaciones. En esta etapa, centraremos nuestros esfuerzos en asegurarnos de que las directrices de diseño y la arquitectura de información sean coherentes con la implementación y la funcionalidad. **La colaboración entre UX, App mobile, Frontend y Backend es esencial para lograr un producto cohesivo.**

HITO

Cuando ya tenemos nuestra AI, Sitemap, y mayor conocimiento del código, podemos ir definiendo nuestro diseño, ir imaginando como se verá gráficamente toda nuestra organización de conceptos y contenido. A nivel de mobile, se espera completar la maqueta, ajustando las diferencias entre entregas, planificando actividades en el tablero Kanban y cumpliendo con los criterios de aceptación para finalizarla al 100%. Por parte de frontend, se desarrollarán formularios de creación de usuarios y productos con validaciones necesarias, documentando los campos de API y endpoints, e implementando una autenticación simulada con roles para vistas protegidas. Como backend, se gestionarán variables de entorno según el ambiente, se creará la imagen Docker y documentación asociada, e integrar Swagger con información dinámica, asegurando la ejecución local y productiva con Docker Compose.

OBJETIVOS

1. Para Diseño UX/UI

- a. Tomando en cuenta las actividades realizadas en las sesiones anteriores, evaluar si es necesario realizar ajustes sobre las historias de usuario existentes.
- b. Según las HdU priorizadas en relación a funcionalidades básicas del e-commerce, sumado a que ya existe un mapa de nuestro sitio y/o App, es necesario empezar a tirar líneas de cómo queremos que esto se vea de cara al usuario. Para esto es necesario pasar por un proceso de bocetaje creado con Wireframes en baja resolución. En el contexto de los sitios web de comercio electrónico, podría ser una página de inicio, una página de carrito de compras, una página de descripción de productos, una página de catálogo de productos, etc. Es fundamental agregar notas de texto al describir estas tareas que demuestren de manera breve y clara el propósito de cada uno de los wireframes.
- c. Diseñar con la metodología MOBILE FIRST, ya que tenemos nuestro contenido y Al casi listas, daremos prioridad al contenido que si o si nuestro usuario debe visualizar.

2. En el repositorio Mobile (si aplica):

- a. Es importante completar la maqueta que se está construyendo. Para ésto primero se debe determinar el gap entre lo solicitado en las entregas anteriores y la entrega actual.
- b. Luego de determinado ese gap, se debe realizar una lista de actividades, la cual deberían registrar en el tablero Kanban como una

historia de usuario con los ajustes necesarios a su prototipo que lo complementen al 100%.

- c. Considerar que es importante cumplir con los criterios de aceptación de lo faltante, es decir, deberán evidenciar que se cumple con lo necesario como para dar terminada la maqueta.

3. En el repositorio Frontend:

- a. Dado que nuestro sitio maneja usuarios y productos por ahora. Debemos crear formularios de creación de usuarios y productos.
- b. El formulario de creación de usuarios debe considerar la incorporación de todos los cambios necesarios para que el backend pueda crear correctamente un usuario. No es necesario integrarnos con el backend vía API para enviar el formulario, sin embargo debemos considerar todas las validaciones para cada campo de formulario y documentar en el servicio de API los campos que esperamos enviar y recibir para la creación del producto
- c. El formulario de creación de productos debe considerar la incorporación de todos los cambios necesarios para que el backend pueda crear correctamente un producto. No es necesario integrarnos con el backend vía API para enviar el formulario, sin embargo debemos considerar todas las validaciones para cada campo de formulario y documentar en el servicio de API los campos que esperamos enviar y recibir para la creación del producto
- d. En nuestro fichero documenta SERVICIOS.md, debemos actualizar y agregar los endpoints que esperamos consultar para cada servicio, así como los verbos http que supone vamos a utilizar para cada servicio que consumamos, sea de consulta o de cambio de estado.

- e. Para la creación del producto y usuarios, deberemos acceder a una página protegida mediante autenticación y rol. Para esto se deberá simular mediante un nombre de usuario y contraseña genéricos la autenticación (sin ir al backend) , y a partir de una correcta autenticación, se deben asignar roles los cuales deben validarse en la vista de creación de producto.

4. En el repositorio Backend:

- a. Modificar el proyecto para cargar las configuraciones desde variables de entorno (al menos el puerto de ejecución del NestJS). **Utilizar @nest/config.**
- b. Validar que estén cargadas correctamente todas las variables de entorno necesarias para ejecutar el proyecto,
- c. Crear 2 archivos para mantener estas variables: **.env.desarrollo**, y **.env.produccion**. Determinar cuál de los archivos se cargará a partir de una variable de entorno del sistema operativo que indique el ambiente de trabajo (desarrollo o produccion).
- d. Crear una imagen docker con el proyecto preparado para instalar en un ambiente productivo. El tag de la imagen debe coincidir con el número de versión del package.json.
- e. Subir la imagen al registro de imágenes de docker hub y completar la documentación de la imagen en docker hub.
- f. Crear archivo docker-compose donde se pueda crear un contenedor a partir de la imagen anterior. Cargar la variable de entorno correspondiente al archivo **.env.produccion**

- g. Crear archivo docker-compose que permita ejecutar el proyecto en el ambiente local de desarrollo (compartiendo los fuentes a través de un volumen). Cargar la variable de entorno correspondiente al archivo **.env.desarrollo**
- h. Configurar el Swagger para mostrar los datos de título, descripción, versión, contacto y licencia obtenidos desde el **package.json**. Como el proyecto tiene configurado un Swagger por módulo, en el título se debe mostrar: **NOMBRE_PROYECTO - MODULO (AMBIENTE)**.
- i. Modificar el archivo **README.md** y ajustarlo de acuerdo al manual adjunto (**Backend-README.pdf**). Tomar en cuenta que los ambientes se iniciarán utilizando docker compose.

CRITERIOS BÁSICOS

Los wireframes ayudan a explorar en las primeras etapas del proceso de diseño.

Los equipos de diseño de UX pueden utilizar wireframes para:

- **Alinear a las partes interesadas.** Los wireframes fomentan la colaboración con estrategias, gerentes de productos y el equipo de desarrollo en cuanto a la estructura de alto nivel, la arquitectura de la información, el flujo de usuarios y el diseño de la interfaz de usuario.
- **Acelerar el diseño de productos.** Los wireframes son fáciles de crear, no requieren retoques. Los equipos de diseño pueden crearlos rápidamente para definir las características básicas y las interacciones de los usuarios.
- **Crear diseños centrados en el usuario.** Los wireframes de prueba de usuarios ayudan a los diseñadores a encontrar una dirección que satisfaga mejor las necesidades de la audiencia.
- **Detectar posibles problemas.** Los wireframes ayudan a los equipos a identificar problemas de forma temprana, lo que evita errores costosos en los diseños finales de un producto digital.

En esta etapa no queremos que visualmente se vea “bonito” sino más bien funcional y ordenado según la AI, nos interesa detectar posibles problemas en etapas tempranas del proyecto. Es por esto que solo trabajaremos con wireframes en baja fidelidad, dibujados a mano o con alguna herramienta online.

- **Equipo Backend:**
 - Los archivos docker-compose.yml se deben colocar en rutas diferentes, para que puedan tener el mismo nombre.
 - Es posible asignar valores a las variables de entorno dentro del Dockerfile con el comando **ENV NOMBRE_VARIABLE=valor_variable**.

- El README.md se revisará directamente en el repositorio de Github y se ejecutará el paso a paso escrito en él para determinar que funcione correctamente.
- La documentación de la imagen se revisará en el repositorio de docker hub. Tomar como ejemplo la documentación de otras imágenes de docker hub para completar documentación de su imagen.
- La ejecución del proyecto se debe realizar utilizando los archivos docker-compose para ambos ambientes (desarrollo y producción).
- **Equipo Frontend**
 - No será necesario integrarse con el backend, pero los envíos de formulario de forma correcta deben ser notificados vía consola de javascript, así como también los errores de formulario deben indicarse en la interfaz de usuario para guiarlo a completarlos correctamente.
 - Consideren un nombre de usuario **administrador**, y contraseña **administrador** para un administrador, así como un nombre de usuario **usuario** y contraseña **usuario** para un usuario normal, lo cual les debe permitir ingresar al sitio de forma autenticada. Consideren los roles “user” para el **usuario** y “admin” para el administrador.
 - Se debe poder autenticar y desautenticar (login/logout)
- **Equipo de Mobile:**
 - La aplicación debe funcionar directamente en el emulador de Android Studio, considerando los elementos solicitados en esta tarea.
 - Deberán evidenciar, a través del uso de la aplicación móvil, del cumplimiento de los criterios de aceptación establecidos para la historia, por lo cual deberán realizar una comparación entre lo definido al principio de la historia y lo obtenido al término del desarrollo.
 - Recuerde que todo debe quedar en el repositorio mobile del GitHub, con su commit respectivo.

- **General**

- Recuerden validar que se hayan subido los cambios a los repositorios de Github.
- Recuerden validar que sus proyectos compilan y se ejecutan correctamente antes de hacer la entrega. (lint y build)
- Recuerden actualizar sus tableros indicando el estado de las tareas e historias de usuario.
- No olvidar adjuntar el hash del commit en github para poder ser evaluados.

ENTREGABLES

Realizar la entrega de una presentación en formato PDF en donde se incluya:

- **Link de Miro**, que contenga : Imágenes de los wireframes realizados según el sitemap construido, esto nos mostrará visualmente cómo se vería aquella página con sus contenidos. Usaremos la metodología Mobile First, por lo tanto los wireframes deben estar contruidos para su versión web mobile, desde ahí pasamos a su versión Desktop, y por otra parte los correspondientes a la App.
- El link debe estar acompañado de screenshots de estos wireframes dentro de la ppt. Para dejar registro en las presentaciones.
- Screenshots de las pantallas desarrolladas para la app (Mobile si aplica), incluyendo las pruebas realizadas que verifican el cumplimiento de los criterios de aceptación.
- Screenshot del fichero **SERVICIOS.md** actualizado y una tabla resumen de servicios a consumir e interfaces construidas (DTO's)
- Un screenshot de la pantalla de creación de productos
- Un screenshot de la pantalla de creación de usuarios.
- Resumen de componentes en react creados y su correspondencia con la maqueta del frontend.
- Screenshot del package.json del proyecto de NestJS.
- Screenshot del Swagger de los módulos.
- Screenshot de los archivos docker-compose.yml
- Screenshot del Dockerfile.
- Screenshot del README.md visto desde Github.

- Screenshot de la documentación de la imagen en Docker Hub.
- Screenshot del estado actual del tablero de github.
- El identificador del commit de cada uno de los repositorios (frontend, backend y mobile) con los desarrollos realizados por cada equipo.

Grabar la presentación de la entrega, en formato de video, de no más de 15 min, en donde presenten:

- Wireframes de Funcionalidades básicas y/o HdU priorizadas.
- Levantar el ambiente de desarrollo desde docker-compose.yml y mostrar el Swagger de cada módulo.
- Levantar el ambiente de producción desde docker-compose.yml y mostrar el Swagger de cada módulo.
- Mostrar la documentación de la imagen realizada en Docker Hub y el README.md desde Github.
- Mostrar el fichero SERVICIOS.md y las interfaces del frontend creadas para comunicarse con el backend (DTO's)
- Mostrar los componentes de react contruidos para el catálogo y detalle de productos y la navegación entre ellos.
- Evidenciar que lo mostrado en las páginas del catálogo y detalle de productos se obtiene desde el backend.
- Evidenciar que lo entregado en mobile esté funcionando de acuerdo al requerimiento y el cumplimiento de los criterios de aceptación.
- Mostrar el estado actual del tablero de github donde se evidencie en cual columna está cada una de las historias de usuario o tareas.

PRESENTACIÓN Y FEEDBACK

- Presenta el Grupo 3.
- Presentación de 15 minutos.
- Feedback de presentación, 10 minutos
- Feedback de entregas por u-cursos al resto de grupos, de forma individual.

MATERIAL DE APOYO

Qué es Mobile First

<https://www.ionos.com/es-us/digitalguide/paginas-web/disenio-web/mobile-first-la-nueva-tendencia-del-diseno-web/>

Impacto negativo que puede tener Mobile First

<https://www.nngroup.com/articles/content-dispersion/>

Como dibujar un wireframe

<https://www.nngroup.com/articles/draw-wireframe-even-if-you-cant-draw/>