



**Certified Tech
Developer**
The Ultimate Degree

DigitalHouse >

DIGITAL RENTAL CAR

Informe de Desarrollo

Antonella Manentti
Daniela Bilbao
Liliana Vasquez

Yesid Barragan
Gabriel Salamanca
David Muñoz



INDICE

Resumen.....	2
Objetivos.....	3
Sprint 1.....	5
Planning.....	6
Daily.....	13
Daily.....	15
Weekly.....	17
Daily.....	19
Review.....	21
Daily.....	23
Retro.....	25

Objetivos

El Proyecto Integrador es una aplicación práctica de los conceptos teóricos estudiados en los bimestres anteriores. Tiene como objetivo el desarrollo de un sitio web. Funciona como una estrategia didáctica que consiste en la realización de un conjunto de actividades interrelacionadas con un inicio, un desarrollo y un final. La metodología que utilizamos en DH de “aprender haciendo” favorece el aprendizaje a través de la articulación de conocimientos teóricos y prácticos, además de acercarte a la realidad del mercado profesional.

- 1- Permite aplicar las herramientas aprendidas en las materias realizadas.
- 2- Nos desafía a llevar las ideas a la práctica.
- 3- Invita a poner en juego nuestras habilidades sociales

El trabajo integrador se realiza en pequeños grupos para estimular la comunicación, la innovación y la creatividad a partir del conocimiento individual y colectivo. Esto permite un trabajo colaborativo. También nos permite practicar nuestras habilidades sociales, favoreciendo el desarrollo de soft skills.

El objetivo final del PI es crear un repositorio de GitLab con el contenido programado, el link y el esbozo de la infraestructura utilizada en AWS. Además, es necesario construir un documento de proyecto.

Al final de cada sprint, Digital Booking espera recibir una actualización del proyecto. Se realizará a través de entregas con la presentación y estatus de cada tarea. Durante el sprint, cada miembro del equipo debe ser el owner de algún requisito que tiene que presentar en la revisión.

SPRINTS

Nº	Tema	Entrega
1	Página home con buscador y login	Realizar una demo mostrando la realización de la pantalla de home y los bloques de login y registro. Presentar una API de categorías funcional.
2	Página de detalles del producto	Realizar una demo mostrando la realización de la pantalla de producto. Presentar API de ciudades y producto y conectarlas con las pantallas de home y producto.
3	Página de reserva y confirmación de reserva	Realizar una demo mostrando la realización de la pantalla de reserva y confirmación de reserva Presentar API de reserva y login.
4	Presentación final	Demo final basada en la presentación de la aplicación completa satisfaciendo los requerimientos obligatorios del proyecto integrador.



Producto seleccionado



Funcionalidades

Página de admin.

Página de registro de usuario.

Página de inicio con buscador y login.

Página de detalles del producto.

Página de reserva.

Página de confirmación del pedido.

Herramientas

FrontEnd

HTML + CSS + JS
React
Node + NPM

Bases de datos

MySQL
MySQL DBeaver

Testing

Postman
JUnit + Mockito

Gestión

Git + GiLlab
Discord
Zoom

BackEnd

Java
Spring
Hibernate

APIs

Rest
JSON

Infraestructura

AWS
Terraform

Responsables de cada área

Desarrollador front-end: Yesid Barragan, Liliana Vasquez

Desarrollador back-end: Gabriel Salamanca, David Muñoz

Analista de base de datos: Gabriel Salamanca

QA / Analista de testing: Daniela Bilbao

Analista de infraestructura: Antonella Manentti

SPRINT 1

Objetivo

Página de inicio con buscador, listado de productos y login.

Crear una página de inicio que incluya el formulario de login y registro. La home debe tener una barra de búsqueda, donde se pueda seleccionar la fecha, lugar, además de un filtro por categorías. También debe presentar una lista de productos que será presentada a partir del filtro seleccionado o de la búsqueda. Y un pie de página con el logo, año, copyright y redes sociales. Además, se debe presentar una API de categorías funcional.

Tareas

- 1.Crear el repositorio del proyecto en Gitlab y añadir los colaboradores del grupo.
- 2.Implementación del template general responsive.
- 3.Implementación de los templates login y registro.
- 4.Implementación del template bloque categorías.
- 5.Implementación del template bloque listado.
- 6.Implementación del template bloque buscador.
- 7.Evento de header.
- 9.API de Categorías.
- 10.CRUD de Categorías.
- 11.Tabla Categorías.
- 12.Un diagrama de la red.
- 13.Planificación y ejecución de las pruebas.
- 14.Prueba de la API.

PLANNING

En este evento desarrollamos el backlog del sprint utilizando las historias de usuario, asignando las responsabilidades* y prioridades de cada historia. Para iniciar, conocimos nuestro grupo asignado por DH, creamos cuentas personales en Gitlab y elegimos un logo, una tipografía, una paleta de colores y una biblioteca de iconos, dentro de las opciones brindadas utilizando Google Fonts Icons y Font Awesome. Debatimos de forma grupal las tareas, su complejidad, estimamos tiempos y las puntuamos utilizando PlanIt Poker. Para finalizar, asignamos las tareas a cada integrante.

14. Prueba de la API.	40	00:00:33
13. Planificación y ejecución de pruebas.	40	00:00:48
12. Un diagrama de la red.	40	00:01:24
11. Tabla Categorías.	40	00:00:23
10. CRUD Categorías.	100	00:00:26
9. API de Categorías.	100	00:00:35
8. Evento de login y usuario	20	00:00:39
7. Evento de header.	20	00:00:52
6. Implementación del template bloque buscador.	20	00:01:28
5. Implementación del template bloque listado.	20	00:01:17
4. Implementación del template bloque categorías.	13	00:01:08
3. Implementación de los templates login y registro.	20	00:01:34
2. Implementación del template general responsive.	20	00:01:01
1. Crear el repositorio del proyecto en Gitlab y añadir los colaboradores del grupo	3	00:01:50

***Las asignaciones serán indicadas al costado de cada tarea, con nombre y apellido del responsable.**

1. Crear el repositorio del proyecto en Gitlab y añadir los colaboradores del grupo. [Liliana Vasquez]

El nombre del repositorio será “grupo_#_nombre”. Donde # es tu número de grupo y nombre es el nombre del proyecto. Debes añadir al resto de los miembros del grupo como colaboradores.

2. Implementación del template general responsive. [Liliana Vasquez]

Header I

Creación de un header al 100 % del ancho de la pantalla y de posición fija al tope de la página que se verá en todas las pantallas de la aplicación, siguiendo el modelo de referencia.

Header II

Dentro del Header I armar un bloque alineado a la izquierda que incluya el logotipo y el lema de la empresa. Ambos deben ser cliqueables por las personas usuarias y dirigir a la home del sitio.

Header III

A su vez, armar un bloque alineado a la derecha que contenga (según el caso) un botón de inicio de sesión y un botón de registro si el usuario no está logueado. En caso contrario, mostrar el nombre de usuario, un avatar de letras con las iniciales del nombre de usuario y un link de cierre de sesión.

Body

Representar un body general con el color de background elegido que ocupe el 100 % del alto de la pantalla.

Footer I

Creación de un footer al 100 % del ancho de la pantalla y al pie de la misma que, se verá en todas las pantallas de la aplicación, siguiendo el modelo de referencia.

Footer II

Dentro del footer armar un bloque alineado a la izquierda que incluya el copyright y el año.

Footer III

Dentro del footer armar un bloque alineado a la derecha que incluya íconos de redes sociales. Estos no necesitan estar vinculados a ninguna página.

Registro: Creación de un formulario de registro que contenga los siguientes inputs:

- Nombre (de type="text")
- Apellido (de type="text")

- Email (de type="email")
- Password (de type="password")
- Repetir password (de type="password")

Además, debe tener un botón que diga Crear cuenta / Registro, y debajo del botón tiene que haber un texto que redirija al login.

3.Implementación de los templates login y registro. [Yesid Barragan]

Los siguientes pasos son necesarios para su implementación:

Bloque general: Creación de un bloque que se muestre a pantalla completa sin scroll, dejando visible en la parte superior de la pantalla solo el header y el bloque de búsqueda. Que pueda contener en su centro un formulario.

Login: Creación de un formulario de inicio de sesión que contenga los siguientes inputs y labels:

- Email (de type="email")
- Password (de type="Password")

Además, debe tener un botón que diga "Iniciar sesión" y debajo del botón tiene que haber un texto que redirija al registro.

4.Implementación del template bloque categorías. [David Muñoz]

Creación de un bloque que incluya un heading <h2> como título, un párrafo <p> y debajo 4 imágenes con sus correspondientes captions. Tienen que respetarse los efectos visuales tales como bordes, sombras, etcétera.

5.Implementación del template bloque listado. [David Muñoz]

Los siguientes pasos son necesarios para su implementación:

Card I: Creación de una card en la que se representará cada producto.

Card II: Dentro de la card incluir un bloque alineado a la izquierda que incluya una imagen centrada.

Card III: Dentro de la card incluir un bloque alineado a la derecha que incluya la categoría del producto, el nombre, ubicación, la descripción. Y un botón que diga ver detalle.

Lista 1: Crear un JSON con un array que contenga 8 elementos. Estos serán objetos con los siguientes atributos:

img: (string) que contenga una dirección a una imagen de banco de fotos.

category: (string) con una de las 4 categorías brindadas por back end tienen que ser incluidas al menos 2 veces cada categoría en el JSON.

title: (string) Título del producto a elegir.

location: (string) Ciudad.

description: (string) Ver catipsum.com.

Lista 2: Crear una función map que permita iterar el JSON y representar 8 cards. Una por cada objeto del JSON.

Lista 3: Retornar la función en un bloque debajo del bloque de categorías.

6.Implementación del template bloque buscador. [Liliana Vasquez]

Buscador I

Creación de un bloque que incluya un heading <h1>.

Buscador II

Creación de un formulario que incluya:

- Select: Ciudad
- Fecha de inicio y fin: Doble calendar
- Botón

7.Evento de header. [Yesid Barragan]

En el caso de mobile estos botones serán accesibles cuando el usuario haga clic en el burger.

Clic botón de ir a login: Deberá mostrarse el header, el buscador y el bloque de login en la pantalla.

Clic botón de ir a registro: Deberá mostrarse el header, el buscador y el bloque de registro en la pantalla.

8.Evento de login y usuario.

Validaciones en inputs: El formulario debe validar que el campo de email sea un email válido y que el password tenga más de 6 caracteres.

Botón: Al hacer clic que funcione la validación de ser correcta que compare los datos con un email y un password de prueba creados en un objeto.

Credenciales inválidas: El formulario indique "Por favor, vuelva a intentarlo sus credenciales son inválidas".

Credenciales válidas: Se simulará que el usuario está logueado, desaparecerá el formulario de login y volveremos a la home inicial, pero en el header a la derecha, en vez de ver los botones de inicio de sesión y registro veremos: "Hola [nombre_de_usuario]", un link de cerrar sesión y un avatar circular con las iniciales del usuario.

9.API de Categorías. [Gabriel Salamanca]

Es necesario crear el proyecto en Java para implementar la API de categorías. Estructurando el proyecto con los paquetes model, repository, service y controller (mínimo).

Desarrollar API de categorías para ser consumida por el frontend. El controller deberá tener los métodos necesarios para:

Agregar una categoría nueva.

Listar todas las categorías.

Editar una categoría en particular.

Eliminar una categoría.

10.CRUD de Categorías. [Gabriel Salamanca]

Conectarnos a la base de datos para realizar las operaciones CRUD. Mapear los atributos de la tabla “categorías” con una clase de nuestro modelo, los atributos serán los mismos que tiene la tabla. Para poder realizar las operaciones CRUD con la base de datos, debemos utilizar Spring Data JPA.

11.Tabla Categorías. [Antonella Manentti]

Crear la base de datos y una tabla llamada “categorías”. Esta tabla deberá contener los siguientes atributos (mínimo): ID, Título, Descripción, Imagen.

12.Un diagrama de la red. [Antonella Manentti]

Se requiere un diagrama de la red AWS con sus componentes para mostrarlo al CEO. Por ejemplo, servidores, almacenamiento, red interconectada y base de datos.

13.Planificación y ejecución de las pruebas. [Daniela Bilbao]

Planificar los casos de prueba basados en las historias de usuario que llegaron para el sprint y, después del desarrollo, ejecutarlos. Realizar también una prueba exploratoria sobre lo desarrollado durante el sprint.

14.Prueba de la API. [Daniela Bilbao]

Validar el funcionamiento de la API REST creada por el back end durante el sprint. Utilizar Postman para las pruebas y crear los scripts de prueba necesarios.

EXTRA [Antonella Manentti]

Realización de un informe. Confeccionamiento de un template para una documentación detallada que funciona como organizador del grupo.

DESARROLLO

Liliana Vasquez

El día en que comenzamos oficialmente. Creé el proyecto con React e intenté subirlo a la rama *main* pero esta pedía permisos adicionales para hacerlo, por lo que creé una rama secundaria. Además, la rama secundaria se planteó como buena práctica para que todos los integrantes del grupo pudiéramos trabajar sin afectar al proyecto principal. Luego, comencé a trabajar en la implementación del template general responsive, creando los archivos que contienen los componentes Header y Footer. Adicional a estos, creé una hoja de estilos *-general.css-* que contendrá las características que compartirán las páginas de nuestra web y los breakpoints a utilizar. Definimos hacerlo mobile first.

Concluí que es importante comenzar con buenas prácticas desde el inicio del proyecto porque estas facilitan nuestro trabajo grupal.

Aprendí a trabajar con Gitlab y valoro sus ventajas como el *board*. Refresqué mis conocimientos de cómo crear un proyecto con React y comandos importantes de Git.

Lo que planeé para mañana es seguir trabajando en los componentes del Header y Footer.

Antonella Manenti

Como primer día de trabajo, creé mi cuenta en Gitlab e inspeccioné la herramienta para un mejor entendimiento de su uso. Me uní a la invitación del proyecto, me asigné las tareas de diagramado de redes y creación de la base de datos y su tabla respectiva. Decidí realizar primero la tarea que más tiempo debía invertir, infraestructura. Revise tanto la información de Gitlab como la de playground para tener en claro los requisitos. Destiné el día en revisión de conceptos y teoría. Además cree un google drive compartido para que compartamos archivos con facilidad.

Gabriel Salamanca

AYER:

1. Revise la información del Sprin y las tareas asignadas a mi nombre para poder tener claro el mapa que debía seguir.
2. Teniendo en cuenta los requerimientos, levanté el proyecto de Maven en spring boot agregando las dependencias necesarias y algunas otras que me parecieron importantes pero que aun no íbamos a utilizar.

HOY:

Planeó empezar a estructurar el programa de Java, con la distribución de carpetas y las clases necesarias para esta parte del proyecto.

IMPEDIMENTOS:

La claridad de que es lo que debe estar en esta parte del proyecto, cuando hablan de las categorías a que se refieren?

Yesid Barragan

- Ayer: Revise el contenido de playground sobre las tareas para el primer sprint.
- Hoy: Este día empecé a revisar información sobre la tarea que me fue asignada, la implementación del template de login y registro, refresqué mi memoria sobre el uso de formularios en react. Revisé los diseños que nos facilitaron en Figma, con el grupo se decidió trabajar con el enfoque mobile first.
- Mañana: Planeé comenzar un proyecto de react con el comando CRA
- Conclusiones: Creo que lo más importante es tener claro el alcance de cada una de las tareas para el primer sprint.

Daniela Bilbao

Creé mi cuenta en Gitlab y me dediqué a inspeccionar el mismo. Una vez ya repartidas las tareas proseguí a hacer una revisión total de la materia "Testing" en el playground, allí marque como primordial lo que tenía que leer y revisar para poder desarrollar mis actividades satisfactoriamente, una vez hecho esto me puse a ver videos y conceptos.

David Muñoz

- Ayer: Realicé lectura del material de playground enfocado en el Sprint I
- Hoy: Este día empecé a repasar conceptos de React casi desde el principio. Realicé creación de la cuenta en gitalab. Accedí a la misma y chequeé su información relacionada con el proyecto.
- Conclusiones: Creo que lo más importante es tener claro el alcance de cada una de las tareas para el primer sprint.

DIA 4 Jueves - DAILY

Liliana Vasquez

Ayer trabajé en la creación del proyecto con React y lo subí al repositorio en una rama secundaria que añadí. También creé los componentes Header y Footer, así como una hoja de estilos general.

Hoy trabajé en los componentes de Header y Footer, añadiendo estructura, y en la hoja de estilos, añadiendo características que compartirán todas las páginas de nuestra aplicación web. Comencé con diseño para mobile, seguí con tablet y, finalmente, con desktop.

Concluyo que comentar la hoja de estilos es vital para un mejor entendimiento del código por parte de las otras personas del equipo.

Una de las mejores lecciones de este día es la importancia de tener diferenciados los estilos generales de la página y la de cada componente que hará parte del proyecto. Esto facilita la visibilidad del código y una futura mejora o corrección.

Planeé para mañana definir la tipografía, paleta de colores y logo de la página. También cambiaré los íconos que había definido en el componente Footer por imágenes svg.

Antonella Manenti

Hoy comencé a realizar un boceto de la red de infraestructura. Se me presenta la dificultad de que no hemos realizado un esquema tan completo, solo uno que abarcaba la única instancia de front-end de EC2. La información del playground no alcanza e inicio búsqueda de mayor información en internet. Me baso en información que comparte AWS y en teoría que publica a través de su programa de educación. Continuamente me mantengo al tanto de Discord y las dudas de otros compañeros que puedan ayudarme.

Repasando información del playground mientras buscamos la forma de organizarnos comunicacionalmente, reparo en el requisito de elaborar un informe. Como tarea, comienzo a realizar un esquema del mismo.

Gabriel Salamanca

AYER:

Creé la estructura de paquetes y carpetas para la API. Se crearon los paquetes model, modelDTO, Repository, Controller y Service. Además de crear la estructura de la clase para atender las necesidades de este spring.

HOY:

Terminar de definir la clase. Separandola en clase DTO y clase que va a impactar en la base de datos para tener una mejor distribución.

IMPEDIMENTOS:

Falta de conocimiento para saber diferenciar entre clase DTO y clase para base de datos y la función del model mapper.

Yesid Barragan

- Ayer: Revisé información sobre la tarea que se me asignó.
- Hoy: Creé un proyecto de react, organicé las carpetas correspondientes, como components y assets. Empiezo a desarrollar los dos componentes para cada uno de los templates, el de login y el de registro. Encuentro un problema sobre cómo enlazar cada uno de los templates, es decir cómo me va a llevar el link de registro a la página de registro e igual con la de login.
- Mañana: Estudié sobre react router.

Daniela Bilbao

Ayer después de haberme informado y refrescado información un poco más di por finalizado mi día.

Hoy empecé a completar mi tarea asignada, luego de revisar el discord que teníamos en dicha materia y de poder revisar actividades que habíamos hecho, decidí crear en excel la tabla exploratoria del primer Sprint.

Finalizada la daily con Selene, procedimos grupalmente a exponer como veníamos trabajando, cerrar dudas, pedir opiniones y -en caso que fuera necesario- realizar una mini votación sobre cómo sería abordado algún tema en especial.

El día de mañana me tocaría completar la primera tabla creada en excel.

David Muñoz

- Ayer: empecé a repasar conceptos de React casi desde el principio. Realicé creación de la cuenta en gitalab. Accedí a la misma y chequeé su información relacionada con el proyecto.
- Hoy: Este día debo continuar repasando los conceptos de React enfocado en creación de componentes a través de carpetas con archivos .jsx y .css. También debo continuar repasando el uso de comandos con npm y el concepto de nodos en los archivos App.js e Index.js
- Conclusiones: Creo que lo más importante es tener claro el concepto de componentes de react y el uso de props.

DIA 5 Viernes - DAILY

Liliana Vasquez

Ayer trabajé en los componentes Header y Footer y la hoja de estilos que reúne las características generales del proyecto.

Hoy definí la tipografía, paleta de colores y logo de nuestro sitio web:

- Tipografía Poppins, estilos Light (300), Regular (400), Medium (500) y Bold (700) - fuente Google Fonts
- Paleta de colores opción C con colores #fbc02d, #607d8b y #263238 - fuente Figma recibido
- Segundo logo amarillo - fuente Figma recibido

Finalmente, cambié los íconos de redes sociales que había implementado en el componente Footer por imágenes svg. Esto con el fin de que al cambiar sus tamaños no se pixelen.

Como conclusión y lección comento que no sabía sobre las ventajas de utilizar imágenes svg.

El lunes de la otra semana trabajaré en algunos aspectos que debo corregir del componente Header y comenzaré con la implementación del componente bloque buscador.

Antonella Manenti

Ayer logré encontrar información que me indique como no solo realizar un esquema, sino además de como poder luego crear todas esas instancias y conexiones en AWS, cumpliendo con la alta disponibilidad y el escalamiento. Mi duda es si debíamos realizar solo un diagrama o ya comenzar a trabajar con AWS, pero al ver que en discord se aclara que las cuentas de Amazon serán dadas más adelante, concluyo que es solo un diagrama. Para realizarlo utilice la herramienta "Visual Paradigm". En el diagramado surge otra duda respecto a las redes privadas para el back-end, queda pendiente para el lunes.

En paralelo continúo desarrollando el informe.

Gabriel Salamanca

AYER:

- Se definió la clase categorías, separando la clase modelo (CategoryDTO) de la clase que tendrá persistencia en la base de datos (Category).
- Se definieron los campos de la base de datos y además se conectó el modelo con una base de datos H2 para hacer las pruebas iniciales antes de conectar el modelo con la base de datos MySQL.

HOY:

Se piensa crear los metodos en el service, implementando los métodos de la JPA. Esta dependencia ha sido agregada con anterioridad.

IMPEDIMENTOS:

Para esta etapa se presento el desconocimiento de la teoria de DTO y clase normal. Pero se ha ido solucionando a medida que se introduce nueva teoria

Yesid Barragan

- Ayer: Empecé con el desarrollo de los componentes de login y registro.
- Hoy: Investigó sobre cómo funcionan los enlaces a otras páginas con react router, encuentro información sobre las Single Page Application y cómo implementarlas con react router. Instaló react router en mi proyecto y empiezo a hacer pruebas con Link to de la librería router. Clono la rama test del repositorio que había creado Lili y empiezo a subir los componentes al repositorio en Gitlab.
- Mañana: Creé la carpeta de pages y desarrollé los componentes de páginas.

Daniela Bilbao

Ayer trabajé en la creación de la tabla de excel y en completar la misma, junto a mis compañeros aclaramos dudas y nos ayudamos mutuamente a la hora de ver cómo emprender una actividad.

Hoy, después de la daily y la reunión con mis compañeros (de la que saque gran provecho) ya que pude exponer lo que venía haciendo y escuchar opiniones pude concluir con la tabla de "prueba exploratoria".

Me compartieron varios links sobre postman que mañana empezaré a leer/ver.

David Muñoz

- Ayer: debí continuar repasando los conceptos de React enfocado en creación de componentes a través de carpetas con archivos .jsx y .css. También debí continuar repasando el uso de comandos con npm y el concepto de nodos en los archivos App.js e Index.js.
- Hoy: inicio la tarea que me corresponde de crear el template de bloque de categorías a través de componentes de react.
- Dificultades: el uso y la incorporación de array de objetos literales tipo .json dentro de los componentes de react. Debo acudir a la ayuda de mis compañeros para poder entender y poder usar estos elementos.
- Conclusiones: es importante repasar los métodos map, filter los bucles foreach, entre otros para manipulación de objetos literales de los cuales se extrae información para componentes en React.

DIA 6 Lunes - WEEKLY

Liliana Vasquez

El viernes trabajé en la selección de la tipografía, paleta de colores y logo de la página. También cambié los íconos del Footer a imágenes svg.

Hoy arreglé algunos detalles pendientes del Header y corregí los botones (tags, estilos y links). Debí reformular los breakpoints para responsive y finalmente quedaron de la siguiente manera: 0 - 479 mobile / 480 - 590 tablet pequeña / 591 - 1199 tablet grande / 1200 + desktop. Quedaron dos breakpoints para tablet ya que había un problema de colapso con los botones del header. Se espera -de haber tiempo disponible en el 4to sprint- implementar un breakpoint para laptop. También debí corregir los componentes Header y Footer para convertirlos de clase a funcionales. Comencé con la implementación del componente de bloque buscador, al que definí como SearchingBlock.

Llegué a la conclusión de que es fundamental definir los breakpoints antes de iniciar a codear, teniendo en cuenta todos los posibles retos o dificultades, para evitar luego correcciones y ajustes.

La lección del día de hoy fue la mejora a futuro para nuestro proyecto de implementar componentes funcionales por sobre componentes de clase.

Mañana planeo seguir trabajando en el componente SearchingBlock.

Antonella Manenti

Comencé el día realizando la consulta pendiente al Teach Lead, en la conversación surgieron otras dudas y llegué a la conclusión de que debía modificar el diagrama realizado. Se utiliza una red privada para back-end junto con base de datos, y el NAT GW debo eliminarlo ya que es pago y modificarlo por una conexión con menos seguridad para comunicar al back-end con el mundo, pero realizar ajustes al momento de configurar las ips y los puertos para reemplazar de algún modo esta seguridad. Me cercioro de dar por finalizada esta tarea hoy. Continuo redactando el informe y completando según la información que me comparten mis compañeros. Mañana crearé la base de datos.

Gabriel Salamanca

SEMANA PASADA:

- Levante el proyecto de Maven en Spring Boot con las dependencias necesarias.
- Cree la estructura de paquetes acorde a lo pedido.
- Cree las clases Category (La que va a tener persistencia y la del modelo como tal).
- Cree los metodos del service implementando los metodos dados por la JPA.

HOY:

- Implementar los metodos del CONTROLLER (GET, POST, DELETE y UPDATE).

-Testear (pruebas unitarias) para tener certeza de que los métodos están funcionando.

Yesid Barragan

- Ayer: Instalé react router en el proyecto y subí los cambios al repositorio en Gitlab.
- Hoy: Creé la carpeta pages para alojar las páginas de nuestro proyecto, home, login y registro. Completé los componentes de las páginas, LoginPage, RegisterPage y Home, con los componentes que desarrolló Lili y los que yo desarrollé.
- Mañana: Empecé con la tarea de evento de login y usuario.

Daniela Bilbao

El viernes trabajé en la tabla de la prueba exploratoria y pude darle un cierre a la actividad.

Hoy después de la Daily comencé a crear la tabla de defectos (template informe de defectos) solicitado para en un futuro poder seguir completando y continuar de cerca el seguimiento para darles seguimiento y solución en caso se presente alguno

Mañana veré vídeos y documentación sobre postman para poder adentrarme en el tema.

David Muñoz

- Ayer: inicié la tarea que me corresponde de crear el template de bloque de categorías a través de componentes de react.
- Hoy: continúo con la tarea del template de bloque de categorías practicando en codesandbox para hacer ensayos más rápidos.
- Dificultades: uso de métodos de array de objetos literales. Debo seguir recurriendo a mis compañeros, a repasar material del PG y búsqueda de información en la red.
- Conclusiones: se debe practicar más el uso de los métodos de array de objetos literales en componentes React.

DIA 7 Martes - DAILY

Liliana Vasquez

Ayer hice algunas correcciones en el componente Header, redefiní los breakpoints de responsive, creé el componente SearchingBlock y pasé los componentes que había hecho de clase a funcionales.

Hoy seguí trabajando en el componente de SearchingBlock, más que todo diseño y estilos.

No fue un día muy productivo porque tuve que consultar algunas cosas y tomó bastante tiempo.

Mañana planeo seguir trabajando en el componente SearchingBlock.

Antonella Manentti

Al momento de planificar este sprint, en grupo coincidimos en que íbamos a trabajar en base de datos con la herramienta Dbeaver, algo nuevo para mí pero altamente recomendado por compañeros que tienen experiencia y trabajan con ella. Hoy dediqué mi día en descargar Dbeaver, aprender a utilizarla, realizar la conexión con MySQL y crear la base de datos con la tabla categorías, sus columnas y la definición de sus atributos (incluyendo especificar el tipo de dato). Una vez hecho esto, envié a través del grupo de whatsapp el nombre de la bdd, usuario y contraseña para que se pueda realizar la conexión con el back-end. Además, me reuní con Daniela (testing) para completar unos casos respecto a cómo realicé la infraestructura y la base de datos recién narrada.

Gabriel Salamanca

AYER:

-Cree los métodos del CONTROLLER para el CRUD de operaciones.

-Se testeó con postman de que los métodos están recibiendo, entregando, borrando y actualizando la información pedida.

-Se revisó en la base de datos H2 que estos métodos impactaran en esta.

HOY:

-Tratar de hacer la conexión a MySQL.

-Ayudar algunos de mis compañeros a sacar adelante sus actividades

Yesid Barragan

- Ayer: Desarrollé los componentes de las páginas del proyecto.
- Hoy: Empecé con la validación de los campos de los formularios de login y registro, también con la validación del login con datos de prueba creados en un objeto. Agregué el menú de hamburguesa para el responsive de móvil.
- Mañana: Continué con el desarrollo de la tarea de evento de login y usuario.

Daniela Bilbao

Ayer vi y leí sobre postamón, baje la aplicación y me registré.

Hoy me puse a investigar dentro de la aplicación y ver para qué servía un poco todo.

A la noche me junté con Gabriel quien sabe mucho de esto y testeamos la API. Me explico como se hacían las pruebas y repasamos un poco de teoría de los diferentes tipos de error y su significado.

Esto lo grabamos y se lo pasamos al grupo ya que ellos no podían conectarse al mismo horario pero todos queríamos saber cómo utilizarlo, fue de gran ayuda.

David Muñoz

- Ayer: sigo con la tarea que me corresponde de crear el template de bloque de categorías a través de componentes de react.
- Hoy: me enfoco en la parte del uso de key y de props.
- Dificultades: uso de key y de props..
- Conclusiones: se debe practicar más el uso de key y de props en componentes React.

DIA 8 Miércoles - REVIEW

Liliana Vasquez

Ayer trabajé un poco en el componente SearchingBlock, pero no pude adelantar lo suficiente.

Hoy planeaba seguir trabajando en el componente SearchingBlock pero estuve ayudando a mi compañero David con la implementación del componente de bloque de categorías. Esto nos tomó toda la mañana y ya luego vino la Review.

Mi conclusión es que tenemos muchos vacíos y al tratar de llenarlos perdemos mucho tiempo en documentarnos para poder implementarlo en el código. En la review fue evidente que nos faltaron muchas cosas. Destaco que hay un muy buen trabajo en equipo, que todos nos hemos comprometido y hemos tratado, en lo posible de nuestras capacidades, sacar el proyecto adelante en estos seis días.

Mañana planeo diligenciar el formato que creamos para el informe que debemos entregar sobre lo que hemos realizado en este sprint.

Antonella Manentti

Hoy terminamos de subir todo a Gitlab y a la rama main. Nos organizamos en equipo como dar la presentación, el orden de oradores, quienes iban a compartir y que decir como una corta introducción a nuestro grupo. En la práctica esto no salió de la forma en que esperábamos, sumado a que no logramos terminar todas las tareas por errores en estimaciones de tiempos y comunicación respecto a las dificultades que atravesamos y que quisimos resolver solos, haciéndonos responsables de la tarea que nos tocó. A pesar del momento, nos reunimos para hablar juntos, darnos palabras de aliento y pensar en mejoras. Respecto a estos problemas que surgieron, el informe también comencé a adaptarlo para que nos sea útil también en el seguimiento de nuestro trabajo.

Gabriel Salamanca

AYER:

- Cree los métodos del CONTROLLER para el CRUD de operaciones.

- Se testeó con postman de que los métodos están recibiendo, entregando, borrando y actualizando la información pedida.

- Se reviso en la base de datos H2 que estos métodos impactaran en esta.

HOY:

- Tratar de hacer la conexión a MySQL.

- Ayudar algunos de mis compañeros a sacar adelante sus actividades

Yesid Barragan

- Ayer: Empecé con la validación de los campos de los formularios.
- Hoy: Antes de la Review traté de implementar el login simulado del usuario, pero no alcancé, debido a un problema con el método que estaba desarrollando.
- Mañana: Finalizar las dos tareas de los eventos.
- Conclusiones: Creo que la falta de tiempo, el estar refrescando conocimientos e investigando no nos permitió llegar con algunas tareas, sin embargo, tenemos un gran equipo, con una gran comunicación y con ganas de aplicar lo que se aprendió en bimestres pasados y de seguir aprendiendo.

Daniela Bilbao

Ayer fue testada la API y se me dio una clase muy útil de postman

Hoy termine de pulir todo y de subirlo al gitlab, practicamos con mis compañeros como llevar a cabo la review tanto en que decir, como que mostrar y en qué orden expondremos, finalmente nuestros planes fueron cambiados estando en la review y igualmente dimos lo mejor de nosotros.

Lección aprendida: ser más relajada si los planes cambian a último momento y no angustiarme.

Terminada la review fueron muchas las palabras de aliento de mi grupo que lograron quitarme el sabor amargo con el que me había ido.

David Muñoz

- Ayer: sigo con la tarea que me corresponde de crear el template de bloque de categorías a través de componentes de react.
- Hoy: me enfoco en la parte de los estilos en el archivo css.
- Dificultades: acomodar adecuadamente los componentes REact para que no se traslapen con otros componentes y con elementos html creados previamente por mis compañeros..
- Conclusiones: el trabajo en equipo es fundamental en mi caso para poder avanzar en mi tarea que ahce parte del front end. He recibido una gran ayuda de parte de todos mis compañeros.

DIA 9 Jueves - DAILY

Liliana Vasquez

Ayer trabajé con un compañero en el componente de bloque de categorías y tuvimos la primer review.

Hoy diligencié el formato que hicimos en el equipo, el cual debemos entregar mañana al finalizar el sprint.

Antonella Manentti

Hoy fue el primer encuentro luego de la exposición. Modifique un error de tipeo del diagrama de red y lo subí nuevamente a Gitlab. Fue un día en equipo para conversar lo ocurrido el día anterior. Hicimos un recuento de los faltantes y los dividimos para poder terminarlos antes del próximo sprint o adelantar todo lo que podamos. El cansancio se nota pero las ganas de querer hacer un excelente trabajo son más. Muy contenta con mi grupo y con muchas ideas y ganas de seguir mejorando y el próximo sprint hacerlo mejor.

Gabriel Salamanca

AYER:

Estuve presente en el testeo de la API. Me lleve muchos comentarios positivos y otros para mejorar. La API esta lejos de estar completa. Tengo que trabajar en algunas validaciones extras y cambiar algunas cosas que no estan bien

HOY: Tuvimos el Review. Tenemos cosas que mejorar, cambiar y arreglar. Pero sabiamos que iba a ser asi. Estoy orgulloso de mi grupo porque lograron en una semana hacer cosas bunas y funcionales empezando desde 0 en conocimientos en todos los temas.

Yesid Barragan

- Ayer: Trate de implementar el login antes de la Review.
- Hoy: Logre solucionar el inconveniente con el localStorage que se me presentó el día anterior, termine las tareas pendientes del evento del header y el evento de login y usuario. También termine de diligenciar el formato para el informe que se entrega el dia de la retro.
- Mañana: Asistir a la retro.

Daniela Bilbao

Ayer mi día terminó después de la review ya que esta terminó alrededor de las 17.

Hoy tuvimos la daily y pudimos todos decir que íbamos a hacer en el día. La sentí muy productiva y me gusto mucho el dialogo y la comprensión que se logró con mis compañeros

Mejore los excel que había hecho para que quedaran mejor presentados.



David Muñoz

- Ayer: me enfoqué en la parte de los estilos en el archivo css.
- Hoy: inicio en la parte de flexbox y media query.
- Dificultades: acomodar adecuadamente los componentes React para que se distribuyan de una forma adecuada en la interfaz en distintos break points.
- Conclusiones: el trabajo en equipo es fundamental en mi caso para poder avanzar en mi tarea que ahce parte del front end. He recibido una gran ayuda de parte de todos mis compañeros.

DIA 10 Viernes - RETRO

En este evento en una primera parte presentamos los avances realizados sobre ciertas tareas de FrontEnd que quedaron incompletas al momento de la exposición.

En una segunda mitad, nos reunimos en una sala con el equipo para realizar la retro de la forma que plantea la modalidad scrum. Utilizamos la aplicación MetroRetro para responder a las preguntas ¿Qué deberíamos hacer? ¿Qué deberíamos dejar de hacer? y nuevas cursos de acción a tomar para mejorar nuestra forma de trabajar.

Start

What should the team start doing?

DARLE MÁS PROVECHO AL TIEMPO POST DAILY 1

Reuniones extra por discord 2

Hacer tareas difíciles en grupo para no trabarnos 3

consultar más a las personas de soporte de DH en discord para dudas técnicas 2

ESTABLECER DESDE EL PRINCIPIO UN ESTANDAR PARA LAS PROPIEDADES DE CSS, CON EL USO DE VARIABLES 2

Mejor estimación de tiempos 1

Preguntar con tiempo cuando no se entienda algo 1

escribir con más frecuencia lo que hacemos y las dificultades diarias en nuestro archivo en word y en el GitLab 2

Preguntar más a los tech para que no nos lo echen en cara en las review 3

Mayor comunicación en nuestras dificultades y progreso diario 3

Stop

What should the team stop doing?

COMPARARNOS CON OTROS GRUPOS 3

No comunicar cuando tenemos problemas y querer resolverlo solos 2

QUEDARNOS CALLADOS SI TERMINAMOS CON MAL SENTIMIENTO 2

reducir intentos de trabajo individual si toma mucho tiempo, por ej dos dias (pues al final se hace necesario pedir ayuda) 2

Continue

What should the team continue doing?

mantener la comunicación x whatsapp 2

SEGUIR CONTIANDO EN NOSOTROS 2

CONTINUAR AYUDANDONOS EN LAS TAREAS NO IMPORTA LA HORA QUE SEA 2

pedir y dar apoyo entre nosotros para dudas y dificultades 2

Actions

Add follow-up actions for the team.

No olvidarse de actualizar Gitlab constantemente (es lo que evolucionan y aparecen las fechas) 3

SI HAY DUDAS CON UNA CONSIGNA PREGUNTAR SI TODOS LA RESOLVERIAN COMO CREO QUE ES 2

Establecer fechas límite más específicas para no retrasarnos 2

si en el segundo día de intentar hacer una tarea no se cerrado, pedir ayuda a quien sea 3

SER MAS ACTIVOS EN EL GRUPO DE WHATSAPP SI SE TIENE ALGUNA DUDA PARA RESOLVER 2

Completar el informe diariamente 1

SPRINT 2

Objetivo

Página de detalles del producto

Realizar una demo mostrando la realización de la pantalla de producto. Presentar API de ciudades y producto y conectarlas con las pantallas de home y producto.

Tareas

15. Crear tabla "características" en la base de datos.
16. Crear tabla "imagenes" en la base de datos
17. Crear tabla "productos" en la base de datos
18. Crear tabla "ciudades" en la base de datos
19. Mapear las tablas "ciudades", "productos" e "imágenes" con clases de nuestro modelo.
20. Agregar relación ManyToMany entre producto y característica
21. Agregar relación OneToMany entre producto e imágenes.
22. Agregar relación ManyToOne entre categoría y producto
23. Agregar relación ManyToOne entre ciudad y producto
24. Agregar las acciones de listar y crear al controlador de ciudades
25. Agregar las acciones de crear, listar y buscar por id al controlador de productos
26. Agregar las acciones de buscar por ciudad o por categoría al controlador de productos
27. Crear las rutas para la página principal y el detalle de producto
28. Crear la página de detalle de producto
29. Crear bloque de fechas disponibles y botón de reserva
30. Implementar el consumo de APIs en el Front End
31. Implementar filtros de búsqueda en el listado de productos
32. Implementar tests automatizados

33. Crear la infraestructura necesaria para alojar el sitio

34. Testing Manual

PLANNING

En este evento desarrollamos el backlog del sprint utilizando las historias de usuario, asignando las responsabilidades y prioridades de cada historia. Para iniciar, nos reunimos como equipo luego de la lectura del Sprint 2. Creamos una cuenta de AWS, usuario root, iam y los permisos, establecimos alerta de presupuesto. Empleando los cambios que propusimos en la Retro del Sprint anterior, realizamos una planning más exhaustiva, debatiendo la nueva forma de organizarnos y de cada issue la forma que sabemos o no sabemos realizarlo con mejor estimación de tiempos basándonos en la experiencia. Utilizando PlanIt Poker puntuamos con la metodología “etiqueta de camiseta” y asignamos las tareas en gitlab.

TITLE:	EST.:	TIME.:
33. Crear la infraestructura necesaria para alojar el sitio	L	00:01:26
32. Implementar tests automatizados	XXL	00:04:05
31. Implementar filtros de búsqueda en el listado de productos	XXL	00:01:35
30. Implementar el consumo de APIs en el Front End	XXL	00:02:46
29. Crear bloque de fechas disponibles y botón de reserva	XXL	00:01:12
28. Crear la página de detalle de producto	XXL	00:02:00
27. Crear las rutas para la página principal y el detalle de producto	L	00:03:31
26. Agregar las acciones de buscar por ciudad o por categoría al controlador de productos	XXL	00:01:54
25. Agregar las acciones de crear, listar y buscar por id al controlador de productos	XL	00:00:52

24. Agregar las acciones de listar y crear al controlador de ciudades	XL	00:01:29
23. Agregar relación ManyToOne entre ciudad y producto	XL	00:00:53
22. Agregar relación ManyToOne entre categoría y producto	XL	00:01:16
21. Agregar relación OneToMany entre producto e imágenes	XL	00:00:40
20. Agregar relación ManyToMany entre producto y característica	XL	00:02:37
19. Mapear las tablas "ciudades", "productos" e "imágenes" con clases de nuestro modelo	L	00:01:51
18. Crear tabla "ciudades" en la base de datos	M	00:00:10
17. Crear tabla "productos" en la base de datos	M	00:00:26
16. Crear tabla "imágenes" en la base de datos	M	00:03:38
15. Crear tabla "características" en la base de datos.	M	00:03:47

#15 Crear tabla "características" en la base de datos. [\[Antonella Manentti\]](#)

Deberá tener como atributos:

- id
- nombre
- icono

Los atributos id, nombre, icono se encuentran en la Tabla "características" tanto en el backend como en la BD.

Se ejecuta el programa en IntelliJ y corre sin inconvenientes

#16 Crear tabla "imagenes" en la base de datos [\[Daniela Bilbao\]](#)

Deberá tener como atributos:

- id
- titulo
- url

Los atributos id, titulo, url se encuentran en la Tabla "imagenes" tanto en el backend como en la BD.

Se ejecuta el programa en IntelliJ y corre sin inconvenientes

#17 Crear tabla "productos" en la base de datos [\[Liliana Vasquez\]](#)

Deberá tener como atributos:

- id
- nombre
- descripcion

Los atributos id, nombre, descripción se encuentran en la Tabla "productos" tanto en el backend como en la BD.

Se ejecuta el programa en IntelliJ y corre sin inconvenientes

#18 Crear tabla "ciudades" en la base de datos [\[David Muñoz\]](#)

Deberá tener como atributos:

- id
- nombre
- pais

Los atributos id, nombre, país se encuentran en la Tabla "ciudades" tanto en el backend como en la BD.

Se ejecuta el programa en IntelliJ y corre sin inconvenientes

#19 Mapear las tablas "ciudades", "productos" e "imágenes" con clases de nuestro modelo. [\[Gabriel Salamanca\]](#)

Crear las clases necesarias dentro de nuestro paquete "Model" para mapear con las tablas de la base de datos.

#20 Agregar relación ManyToMany entre producto y característica [\[Liliana Vasquez\]](#)

Utilizar Hibernate o realizarlo desde la base de datos y luego mapearlo en la clase producto.

La clase producto deberá contener un listado de características dentro de sus atributos.

Comentario

En la clase característica se genera un atributo del tipo Productos que también es un array y al cual se le hace la anotación @ManyToMany y también se agrega el mappedBy.

En la clase Product se genera un atributo del tipo Característica que también es un array y al cual se le hace la anotación @ManyToMany y la anotación @JoinColumn

#21 Agregar relación OneToMany entre producto e imágenes. [Daniela Bilbao]

Utilizar Hibernate o realizarlo desde la base de datos y luego mapearlo en la clase producto.

La clase producto deberá contener un listado de imágenes dentro de sus atributos

Comentario tarea #21:

En la clase imagen se genera un atributo del tipo Productos y al cual se le hace la anotación @ManyToOne y la anotación @JoinColumn.

En la clase Product se genera un atributo del tipo imágenes que también es un array y al cual se le hace la anotación @OneToMany y también se agrega el mappedBy.

#22 Agregar relación ManyToOne entre categoría y producto. [Antonella Manentti]

Utilizar Hibernate o realizarlo desde la base de datos y luego mapearlo en la clase producto.

La clase producto deberá tener un atributo que indique la categoría a la que pertenece, para implementarlo vamos a usar una relación muchos(producto) a uno(categoría).

Comentario tarea #22:

En la clase imagen se genera un atributo del tipo Productos y al cual se le hace la anotación @ManyToOne y la anotación @JoinColumn.

En la clase Product se genera un atributo del tipo imágenes que también es un array y al cual se le hace la anotación @OneToMany y también se agrega el mappedBy.

#23 Agregar relación ManyToOne entre ciudad y producto [David Muñoz]

Utilizar Hibernate o realizarlo desde la base de datos y luego mapearlo en la clase producto.

La clase producto deberá tener un atributo que indique la ciudad a la que pertenece, para implementarlo vamos a usar una relación muchos(producto) a uno (ciudad).

Comentario tarea #23:

En la clase Product se genera un atributo del tipo City y al cual se le hace la anotación @ManyToOne y la anotación @JoinColumn.

En la clase City se genera un atributo del tipo Product que también es un array y al cual se le hace la anotación @OneToMany y también se agrega el mappedBy.

#24 Agregar las acciones de listar y crear al controlador de ciudades **[Yesid Barragan]**

El controlador de ciudades deberá permitirnos:

- Listar todas las ciudades
- Crear ciudades (OPCIONAL)

Comentario tarea #24:

Después de haber creado las clases de las capas del Model, DTO, Service y las interfaces de la capa Repository, la clase CityController es creada. En ésta son creados los métodos save y findAll con ciertas anotaciones y ciertos parámetros. Se muestra a continuación parte del código de estos métodos:

```
@PostMapping("/save")
```

```
public ResponseEntity save(@RequestBody CityDTO cityDTO)
```

```
@GetMapping()
```

```
public List<CityDTO> findAll()
```

#25 Agregar las acciones de crear, listar y buscar por id al controlador de productos **[Gabriel Salamanca]**

El controlador de productos deberá permitirnos:

- Crear productos
- Listar productos
- Buscar y retornar producto por id

Después de haber creado las clases de las capas del Model, DTO, Service y las interfaces de la capa Repository, la clase ProductController es creada.

#26 Agregar las acciones de buscar por ciudad o por categoría al controlador de productos **[David Muñoz]**

El controlador de productos deberá permitirnos:

-Listar productos según categoría, es decir, nos deberá devolver los productos que pertenezcan a cierta categoría.

-Listar productos según ciudad, es decir, nos deberá devolver los productos que pertenezcan a cierta ciudad

Después de haber creado las clases de las capas del Model, DTO, Service y las interfaces de la capa Repository, la clase ProductController es creada.

#27 Crear las rutas para la página principal y el detalle de producto [David Muñoz]

Crear un enrutador con las siguientes rutas:

- Página de inicio

- Detalle de producto. Ésta última estará asociada al id de cada producto (producto/[identificador_de_producto]).

Comentario #27:

Se utilizó react router para hacer las rutas de los componentes de página; LoginPage, RegisterPage, ProductPage y Home. Para la página de detalle de producto se obtiene el id por la URL y se realiza la petición a la API.

#28 Crear la página de detalle de producto [Liliana Vasquez]

Crear la página de detalle de producto

La página deberá seguir la estructura definida en los diseños y deberá incluir los datos relevantes del producto:

- Título y Categoría

- Ubicación, Imágenes, Descripción, Características

- Disponibilidad

- Política

Contenedor

- Esta página debe tener como contenedor el componente template general creado durante el primer sprint.

Bloque Heading

- Debe cubrir el 100% del ancho de la pantalla.

- Incluir alineado a la izquierda dentro del área de contenido:

- .Categoría del producto

- .Título del producto

- Incluir alineado a la derecha dentro del área de contenido:

- .Flecha volver atrás: Implementar un ícono debajo del Heading que permita volver a la Home.

Bloque de ubicación

- Crear un bloque que cubra el 100% del ancho de la pantalla que incluya dentro del área de contenido los datos de ubicación.

Bloque de Imágenes

Se presentará una galería de imágenes en los siguientes formatos dependiendo del dispositivo.

- La galería de imágenes debe estar incluida como librería en NPM o Yarn.

- La galería de imágenes debe incluir un botón que permita cerrar y retornar a la visualización del detalle del producto.

Desktop

- Debe presentar un bloque al 100% del ancho del contenedor que incluye 5 imágenes. Las mismas deben estar distribuidas del siguiente modo:

- .Mitad izquierda del bloque: Imagen principal.

- .Mitad derecha distribuida en una grilla de 2 filas y 2 columnas que incluye 4 imágenes.

- Las imágenes deben presentar bordes redondeados y no deben disparar ningún evento.

- El bloque debe incluir en su región inferior derecha el texto "Ver Más", el cual, permitirá acceder a un componente de galería de imágenes para ver todas las imágenes disponibles del producto. (la elección de la galería específica a utilizar es libre, lo mismo que su estética (no es necesario seguir al pie de la letra la galería presentada en Figma).

Tablet y phone

- Debe presentar una galería con auto deslizamiento de imágenes al 100% del ancho del contenedor que incluye 1 imagen que irá cambiando a medida que pasen 3 segundos.

- El bloque debe incluir en su región inferior derecha un texto que indique cuál es el número de la foto actual y el total de imágenes (la elección de la galería específica a utilizar es libre).

Bloque de descripción

- Crear un bloque que cubra el 100% del contenedor que incluya:

- .Título

- .Texto de descripción del producto.

Bloque de características del producto

- Crear un bloque que cubra el 100% del contenedor que incluya:

- .Título subrayado.

- .Grilla de (4 desktop, 2 tablet y phone) columnas con una enumeración de los atributos del producto junto a su ícono asociado.

Bloque de política del producto

-Crear un bloque que cubra el 100% del contenedor que incluya una grilla de (3 desktop, 2 tablet, 1 phone) columnas con:

.Normas: Título y descripción.

.Seguridad: Título y descripción.

.Cancelación: Título y descripción.

#29 Crear bloque de fechas disponibles y botón de reserva [Antonella Manentti]

Crear bloque de fechas disponibles y botón de reserva

Desktop

-Debe incluir un calendario alineado a la izquierda y que ocupe ($\frac{2}{3}$ partes de la grilla de contenido) que muestre 2 meses simultáneamente que indique tanto fechas disponibles como no disponibles.

-El calendario sólo debe permitir navegar entre distintos meses. (su finalidad es sólo visualización de disponibilidad).

-Bloque de reserva debe ocupar $\frac{1}{3}$ parte de la grilla de contenido:

.Alinearlo a la derecha del calendario.

.Debe incluir un texto.

.Debe incluir un botón de "Iniciar Reserva" (sin eventos).

Tablet

-Debe incluir un calendario al 100% del ancho del contenedor que muestre 2 meses simultáneamente que indique tanto fechas disponibles como no disponibles.

-El calendario sólo debe permitir navegar entre distintos meses. (su finalidad es sólo visualización de disponibilidad).

-Bloque de reserva con una grilla de 2 columnas:

.Debe incluir un texto.

.Debe incluir un botón de "Iniciar Reserva" (sin eventos).

Phone

-Debe incluir un calendario al 100% del ancho del contenedor que muestre 1 mes simultáneamente que indique tanto fechas disponibles como no disponibles.

-El calendario sólo debe permitir navegar entre distintos meses. (su finalidad es sólo visualización de disponibilidad).

-Bloque de reserva:

.Debe incluir un texto al 100% del ancho de la pantalla.

.Debe incluir un botón de "Iniciar Reserva" (sin eventos) al 100% del ancho de la pantalla.

#30 Implementar el consumo de APIs en el Front End [Yesid Barragan]

-Los elementos del listado de productos deben ser obtenidos de la API de productos.

-Los elementos del bloque de categorías deben ser obtenidos de la API de categorías.

-El selector de ciudades debe presentar en sus opciones datos consumidos de la API de ciudades.

-En la página de detalle de producto, utilizar el id de la URL para hacer un llamado a la API de producto por id, con el fin de completar la información de la pantalla con los datos devueltos.

#31 Implementar filtros de búsqueda en el listado de productos [Yesid Barragan]

-Los elementos del listado de productos deben poder ser filtrados al seleccionar una categoría. en el bloque de categorías.

-Al seleccionar una ciudad en el selector de ciudades y dar clic en el botón de búsqueda, la lista de productos debe filtrar los elementos por la ciudad elegida.

#32 Implementar tests automatizados [Daniela Bilbao]

-A través de Selenium webdriver, crear los scripts de tests automatizados necesarios para cubrir lo que fue desarrollado durante los sprints 1 y 2.

-Implementar los tests de las APIs utilizando Postman.

#33 Crear la infraestructura necesaria para alojar el sitio [Antonella Manentti]

1- Crear una instancia de EC2 en AWS para el servicio web

- La misma deberá estar comprendida dentro del free-tier (t2.micro)
- La misma deberá estar en la vpc "default" con IP pública
- La misma deberá tener asignada una key privada
- La misma deberá tener un security group permitiendo el acceso desde el mundo al puerto de su aplicación y acceso al puerto 22 desde sus respectivas IP
- Instalar un servidor web, por ejemplo: Apache, Nginx, etc.
- Instale los paquetes necesarios para que la aplicación puedan funcionar

2- Crear una instancia de RDS en AWS para alojar la base de datos

- Crear la base de datos para almacenar los datos del sitio de reserva. Ej. MySQL
- La misma deberá estar comprendida dentro del free-tier

- La misma deberá estar en la vpc "default" con IP pública
- La misma deberá tener un security group permitiendo el acceso al puerto de la db (3306 en mysql) desde el security group de la instancia EC2, y desde sus respectivas IP
- Crea tablas y relaciones en la base de datos.
- Conectar la base de datos a la instancia de EC2.

3-Crear un Bucket S3 para almacenar las imágenes de los productos

- Almacenar las imágenes de los productos
- Agregar las URLs a la base de datos

#34 Testing Manual [\[Daniela Bilbao\]](#)

- Realizar testing estático sobre las user stories del sprint.
- Realizar testing exploratorio sobre las funcionalidades agregadas en el sprint (documentar la ejecución)
- Agregar casos de prueba manuales sobre nuevas funcionalidades del sprint.
- Realizar mantenimiento de los casos de prueba ya creados, en caso de ser necesario.
- Clasificar los casos de prueba en Smoke y Regression Test Suite.

Importante: Agregar el link a la planilla de casos de prueba y defectos en el readme.md del proyecto.

Extras implementados

- **Épica usabilidad**

Hover en botones

- **Épica visualización del producto**

Bloque de mapa de Google

DESARROLLO

DIA 2 Martes - DAILY

Liliana Vasquez

Ayer, luego de la lectura del Sprint, nos reunimos para puntuar las 19 tareas correspondientes a este. Tal como lo sugirió el PO, utilizamos en PlanItPoker la técnica de talla de camisetas para evaluarlas y priorizarlas. Muchas de las tareas eran complejas de comprender, por lo que antes de evaluar cada una de ellas, dialogamos para contextualizarnos y comprenderlas bien. Luego, nos distribuimos las tareas.

Hoy nos reunimos para realizar el diagrama relacional de las tablas del Backend que, posteriormente, creamos en la base de datos. Para la parte de Infraestructura creamos la cuenta de AWS e hicimos algunas configuraciones.

Tuvimos inconvenientes para conectar la base de datos.

Antonella Manentti

Realizamos la daily. Recibimos feedback de lo realizado en el Sprint 1 de parte de los TL. En la reunión grupal organizamos y dividimos las tareas y estimamos tiempos. Habilité más permisos para el usuario IAM. Cree vpc, 4 subredes, internet gw, EC2 para front end, RDS para base de datos. Intenté conectar la bdd con AWS pero no pude, en grupo lo intentamos y no hubo caso. Por estos problemas decidimos crear en mi pc las 5 tablas de la base de datos de forma manual en Dbeaver para poder hacerlo de forma más rápida y luego haga la conexión con AWS, para esto en grupo primero realizamos un diagrama. Como pendiente para mañana voy a seguir buscando información y lograr la conexión.

Gabriel Salamanca

Yesid Barragan

Se reunió todo el grupo para crear la cuenta de AWS y configurar la cuenta IAM, también hicimos el diagrama de entidad relación para el backend y se crearon las tablas en la base de datos.

Daniela Bilbao

Después de presenciar la lectura del segundo sprint nos conectamos vía Discord con el grupo para hacer las cuentas de AWS, el planItPoker y comentar sobre qué trata cada tarea nueva para así poder asignarlas en Gitlab. Desde las 20HS hasta aproximadamente las 00 hs Argentina nos juntamos grupalmente via zoom para crear juntos el diagrama y en base a eso se prosiguió a crear las tablas en la base de datos. Repartimos nuevas tareas y algunos temas del back end fueron explicados.



Mañana asistiremos a la Daily y nos pondremos a planificar más meticulosamente el sprint que vamos a afrontar.

David Muñoz

Realizamos el diagrama de las Tablas Relacionales en la herramienta diagrams teniendo en cuenta los atributos correspondientes para cada una.

También iniciamos la creación de la Base de Datos en AWS, en la cual surgieron algunos inconvenientes para poderla conectar con la Tabla creada en Dbeaver. La idea es que todos podamos acceder a la DB en AWS para poder alimentarla con información de automóviles.

DIA 3 Miércoles - CONFIRMACIÓN DE SPRINT

Liliana Vasquez

Junto a Yesid y David, creamos el componente ProductDetails que contiene los bloques Heading, Ubicación, Imágenes, Descripción y Características del Producto. Se hizo la maquetación y se agregaron algunos estilos teniendo en cuenta los diferentes breakpoints definidos en el primer sprint. Posteriormente, se implementó el componente en la página ProductPage para su respectiva renderización.

La única dificultad que se presentó es que el bloque Imágenes era que rompía el código y no comprendíamos la razón. Lo solucionamos quitando las llaves del import.

Antonella Manenti

Luego de realizar la confirmación de sprint, continúe configurando la infraestructura y tratando de solucionar el problema de conexión entre RDS y la base de datos. Tuvimos un inconveniente con AWS, lo hablamos y enviamos un email detallando lo sucedido a DH. Por la tarde el TL informa que no vamos a seguir los diagramas que realizamos en el sprint 1 y detalla cómo será la nueva configuración. Eliminé todo y volví a empezar. Creé RDS, EC2 y Bucket S3, configuré los security groups y pude solucionar el problema y realicé la conexión de la base de datos con RDS.

Gabriel Salamanca

Yesid Barragan

Junto con Lili y David comenzamos con la creación de la página de detalle de producto, decidimos dividir la página en 3 componentes; componente de características del producto, componente de reserva y el componente de políticas. Para el componente de características del producto se utilizó la librería react-image-gallery para la representación de la galería de imágenes. Yo me encargue del componente de reserva.

Daniela Bilbao

Hoy comencé a ver videos de Selenium, hice la instalación de JDK tools developing, cree la variable de ambiente "JAVA_HOME", después incluí este directorio en la variable path. Descargue las dependencias de chrome y por ultimo, descargue Maven y lo acople al IntelliJ.

Mañana comenzaré con la nueva tarea asignada de GitLab, "Testing manual". jejeje

David Muñoz

Hoy junto con Yesid y Liliana estuvimos trabajando en la página de detalle de producto. Iniciamos la implementación de la librería para las imágenes en mobile.

Estuve viendo los videos del profesor Javier Rausch de Backend sobre creación de clases en capas Model y DTO usando hibernate y el framework de springboot.

DIA 4 Jueves - DAILY

Liliana Vasquez

Junto a mis compañeros Yesid y David trabajamos en el bloque Imágenes del componente ProductDetails, dándole los estilos solicitados de los requerimientos y adaptándolo a los diferentes breakpoints.

Se nos presentó una dificultad en la versión Desktop al querer quitar la visualización previa. Implementamos la opción `window.innerWidth < 1199`, pero debemos refrescar la página para que el cambio se aplique, que no es la mejor solución. Vamos a preguntarle al TL para que nos ayude con este inconveniente. Para este bloque utilizamos `react-image-gallery`.

Antonella Manenti

Hoy continué trabajando en la Infraestructura e instalé el servidor web Apache en la EC2. Tengo dudas respecto a cómo colocar y configurar la base de datos dentro del RDS, trabajo en ello, si para mañana no puedo solucionarlo agendaré una reunión con el TL. Comienzo a repasar teoría sobre testing.

Gabriel Salamanca

Yesid Barragan

Se trabajó en la galería de imágenes con la librería para mobile y tablet, se presentó un problema para obtener el valor del ancho de la ventana para deshabilitar la librería de la galería en desktop. Este inconveniente lo solucione con la ayuda de un custom Hook, este se encarga de obtener el valor del ancho de la ventana.

Daniela Bilbao

Hoy se trabajó en las nuevas tablas de excel a entregar, tanto la prueba exploratoria como los template de casos de prueba, la tercera será cuando haya más para testear.

Mañana me pondré al día con la nueva tarea de GitLab que fue agregada hoy (Testing manual).

David Muñoz

Hoy continuamos con Yesid y Liliana estuvimos trabajando en la página de detalle de producto. Se terminó la implementación de la librería para las imágenes en mobile. Los elementos de número de páginas y símbolos de compartir y me gusta también fueron agregados. Agregamos el bloque de políticas que va en la parte inferior de la página de detalle de producto. Observó dificultades en cuanto al calendario. Hemos hecho bastante búsqueda en la web y ajustes en nuestro proyecto y así veo que por ejemplo al hacer click en una fecha también hace círculo sobre el mismo día de los otros meses. Para hacer ajustes a la forma y al CSS del Calendario hemos dedicado mucho tiempo. También he visto que otros grupos están reportando inconvenientes con el



tema del calendario. Considero que necesitamos apoyo de alguien de DH que nos explique y/o que nos ayude con esto.

Estuve viendo los videos del profesor Javier Rausch de Backend sobre creación de clases en capas Service. También estuve con Gabriel haciendo un par de clases en la capa Service. Luego yo hice las clases Service de Characteristic y de Image. Tuve que hacer constructor, getters y setters a las clases respectivas en Model. También hice las Interfaces Repository correspondientes.

DIA 5 Viernes - DAILY

Liliana Vasquez

Ya que mi compañero Yesid logró solucionar el problema que presentábamos con la versión Desktop, estuvimos trabajando todo el día en el formato de la galería en este breakpoint. Utilizamos grid y el resultado fue positivo. Para la opción de “Ver más” implementamos la librería Modal.

Agregué hover a todos los botones de la web (implementación extra) y solucioné el problema de que el footer se subiera en las páginas de Login y Registro. Además, arreglé el responsive del componente ProductPolicyBlock de la página de productos.

Antonella Manentti

Realicé la consulta sobre la base de datos al TL por discord, confirmado que no hay que subir nada de nuestro proyecto a AWS, el issue de infraestructura queda terminado, a excepción de que iré instalando los paquetes a medida que avancemos ya que no nos queda claro bien cuales son. Consultado con mi compañera Daniela, ambas trabajaremos en la implementación del testing automatizado. Comenzaré con la instalación de los programas y paquetes necesarios, en paralelo seguiré buscando información.

Gabriel Salamanca

Yesid Barragan

Junto con mi compañera Lili trabajamos en la implementación de la galería para desktop, utilizamos grid para las imágenes, para la galería que se despliega al hacer click en ver más se utilizó la librería que se trabajó en mobile y tablet. Para el modal de la galería se utilizó la librería react-responsive-modal.

Daniela Bilbao

Hoy comencé con la nueva tarea de Gitlab, “Testing manual” la parte de testing estático, cree una tabla de excel en la cual comencé con los casos de prueba, ya que también se pedía, los estoy haciendo tanto con el primer como con el segundo Sprint. Se pide también que se muestre la documentación y que tipo de prueba es, para esto cree dos columnas adicionales a los casos de prueba para que se pueda leer mas claramente y sea mas cómodo.

Una vez finalizado voy a proseguir a terminar detalles de la tabla de pruebas exploratorias para ya terminarlo.

David Muñoz

Estuve viendo los videos del profesor Javier Rausch de Backend sobre creación de clases en la capa Controller. También estuve con Gabriel haciendo un par de clases en la capa Controller. Luego yo hice las clases Controller de Characteristic y de Image.

DIA 6 Lunes - WEEKLY

Liliana Vasquez

Pasé el componente categorías de la página Home de flexbox a grid. Adicioné estilos como border y shadow. Quedaron adecuados para que fueran responsive. Un pequeño problema que me surgió fue la aparición de un espacio extra en el lateral derecho de la grilla padre, que no he logrado corregir.

Adicionalmente, surgió un problema -aún no sabemos el por qué- con la galería de la página de producto. Las imágenes no pierden el formato de grilla que habíamos adecuado para la versión Desktop, pero se agrandan muchísimo y se desbordan del ancho y alto de la pantalla.

Antonella Manentti

Hoy me reuní con Daniela, quien me explicó su trabajo realizado en testing para poder yo comenzar a buscar información sobre testing automatizado. Buscamos documentación sobre Selenium, Jest, Junit y Postman.

Gabriel Salamanca

Yesid Barragan

El fin de semana estuve investigando y haciendo pruebas con la librería Axios para las peticiones HTTP a la API.

Daniela Bilbao

Ayer habia comenzao la tabla de testing manual, hoy complete todos los test tanto del primero como del segundo sprint, faltaria agregarle la documentacion.

Hoy agregue toda la documentacion de las pruebas manuales y actualicé y agregue nuevos errores a la tabla de defectos.

Mañana comienza la etapa de pruebas automatizadas, que, junto a Antonella vamos a realizar y grabar para poder adjuntar a la documentación.

David Muñoz

Revisando con Gabriel, encontramos que hay que corregir un par de métodos en Clases de las capas de Service y Controller. Hacemos las correcciones.

Descargo Postman, Dbeaver y Jest teniendo como objetivo contribuir en lo relacionado a la API y sus endpoints.

DIA 7 Martes – DAILY

Liliana Vasquez

Nos reunimos con el equipo para subir información en las tablas que ya estaban creadas de la base de datos .

Agregué color en textos del componente RentCategory. Agregué color en textos y solucioné el problema responsive del componente ListBlock.

Antonella Manentti

Nos mantuvimos en reunión todo el día con el equipo para ir puliendo entre todos las tareas faltantes. Conecte a todos mis compañeros a la RDS y realizamos la conexión con DBeaver. Instalé terraform y AWS CLI, los configure y realicé la conexión con nuestra cuenta de AWS. Configure vpc, subred publica, tabla de ruteo pública, asociación de tabla de ruteo con subred, internet gw en terraform.

Gabriel Salamanca

Yesid Barragan

Se llenaron algunos campos de las tablas en la base de datos. Empecé con las peticiones a la API con Axios de los endpoints que ya estaban listos.

Daniela Bilbao

Ayer se cerraron 2 de 3 tablas de excel.

Hoy abrí la ultima, la de pruebas exploratorias y adicione todo del segundo sprint.

Nos reunimos en equipo a crear la informacion en las tablas que ya estaban creadas de la base de datos y poder cerrar ese issue.

David Muñoz

Adicional a lo compartido por la compañera Antonella, comienzo a ver videos relacionados con Postman, Dbeaver y Jest teniendo como objetivo contribuir en lo relacionado a la API y sus endpoints.

DIA 8 Miércoles - REVIEW

Liliana Vasquez

Me reuní con el TL de Front para que me ayudara a debuguear el problema que estábamos presentando en la versión desktop con el tamaño de las imágenes de la galería en la página de productos. Luego nos reunimos con el equipo para preparar la presentación de la review.

Por mi cuenta, estuve estudiando un poco de react con videos.

Antonella Manentti

Nos reunimos con el TL de Front para resolver dudas. Nos reunimos como equipo para practicar nuestra presentación y lograr respetar el horario de exposición y que todos hablemos. Luego de la review y con el feedback recibido, realicé el cambio del servidor web de Apache por Nginx ya que tiene una mejor performance. Continúe informándome sobre terraform.

Gabriel Salamanca

Yesid Barragan

Finalicé el componente del mapa con la librería leaflet, además de alimentar este componente con información de latitud y longitud obtenida de la API.

Daniela Bilbao

Ayer se completo toda la tarea del testing manual y se pudo cerrar ese issue.

Hoy se logro completar los test en Postman para poder testear la Api y sus endpoints en totalidad.

Mañana se hará los test de Jest para asi poder testear el front y tener los test automatizados.

David Muñoz

Junto con ayuda de Yesid y de Antonella, realizo algunos tests con datos en Postman y Dbeaver teniendo como objetivo contribuir en lo relacionado a la API y sus endpoints.

DIA 9 Jueves - DAILY

Liliana Vasquez

Junto con mi compañero Yesid estuvimos volviendo a levantar la base de datos, luego de un problema que surgió con la que teníamos. Subimos la información de categorías y detalles del producto. Agregamos todas las imágenes de los diferentes automóviles a la S3 de AWS. Posteriormente, alimentamos el front con dicha información y quedó desplegada correctamente en la página Home.

El problema que surgió fue referente a los estilos, ya que al tener diferentes medidas, las imágenes cambiaron el height del contenedor.

Antonella Manentti

Terminé de configurar rds, ec2, bucket s3 y grupos de seguridad en terraform. Complete el informe con los avances del sprint 2. Redacté el README de gitlab, tal como fue pedido en el feedback del sprint 1.

Gabriel Salamanca

Yesid Barragan

Debido a una modificación en la API se alimentó de nuevo la base de datos y se agregaron las url de las imágenes que fueron almacenadas en un Bucket de AWS. Se realizó la petición a la API para renderizar esta información en el componente de Home.

Daniela Bilbao

Hoy se comenzó a instalar jest y comenzar las pruebas para el front, hubo un error que nos impidió poder seguirlo.

Mañana se terminará el test y se podrá cerrar el issue de los test automatizados.

David Muñoz

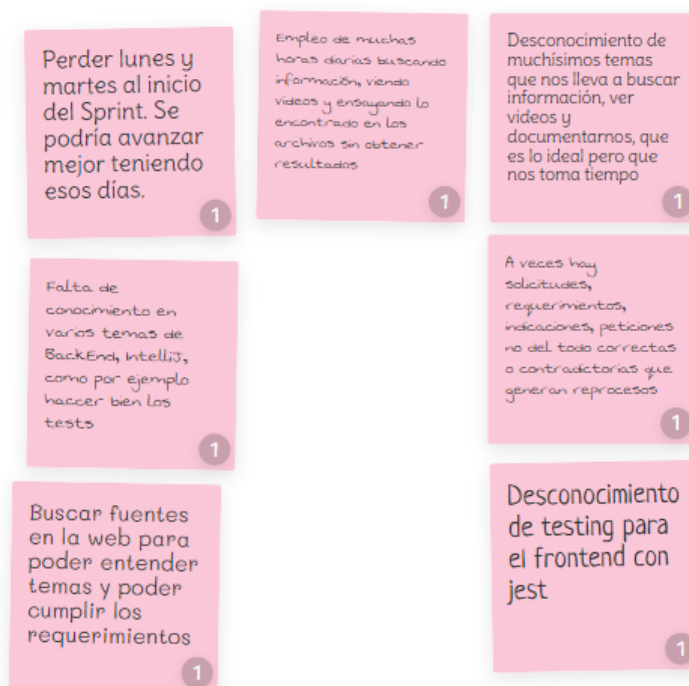
Teniendo como objetivo contribuir en lo relacionado a tests en el backend, comienzo a consultar información y ver videos relacionados con JUnit en IntelliJ.

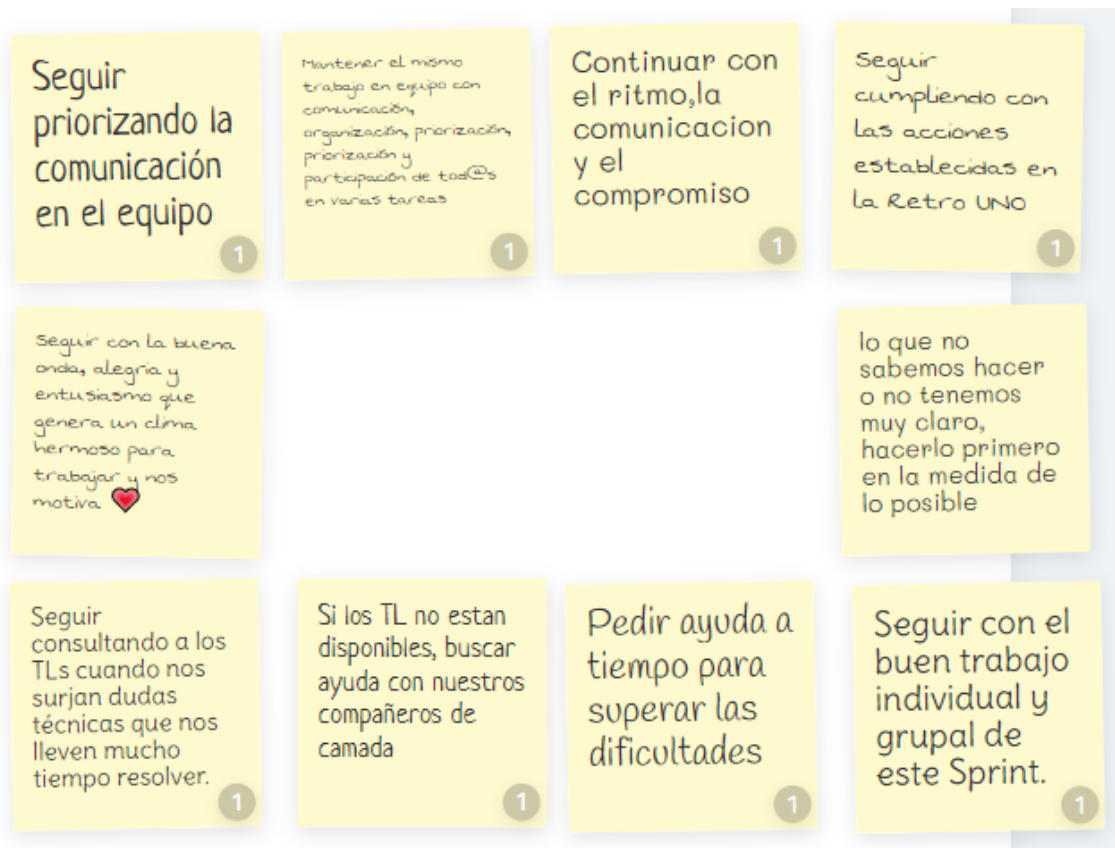
He estado teniendo problemas para poder hacer un test con el método findByName de la Clase CityService. Empiezo a solicitar ayuda.

DIA 10 Viernes - RETRO

En este evento en una primera parte presentamos los avances realizados sobre ciertas tareas de FrontEnd con la finalización de la API que quedaron incompletas al momento de la exposición. En paralelo seguimos trabajando para finalizar todos los issues.

En una segunda mitad, nos reunimos en una sala con el equipo para realizar la retro de la forma que plantea la modalidad scrum. Utilizamos la aplicación MetroRetro, en esta ocasión la forma de analizarlo fue respondiendo las preguntas ¿Qué nos ralentiza? (rosa) ¿Qué hace que avancemos? (verde) Y acciones a seguir (amarillo).





Yesid Barragan

Se completó la página de detalles del producto con la información proveniente de la API, además del ruteo para cada uno de los productos en la página Home. Se hicieron algunos ajustes en los estilos del componente Home.