

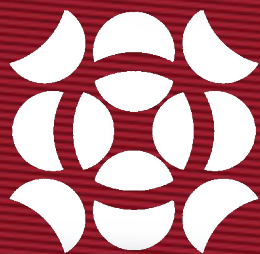


Universidad Nacional Autónoma de México

Instituto de Ciencias Aplicadas a la Tecnología

Desarrollo de herramientas computacionales
para el laboratorio de termodinámica

Plan de trabajo



ICAT

Instituto de Ciencias
Aplicadas y Tecnología

Alumno:

Josue Demian Avila Romero

Profesor responsable:

M. en I. David Palomino Merino

Introducción

La irrupción de la pandemia presentó retos académicos que la comunidad universitaria ha ido sobrellevando de acuerdo a sus posibilidades y dentro de los aprendizajes que deja la situación mundial queda el ampliar el papel que juegan las herramientas digitales en el proceso de participación y adquisición de conocimientos.

Los sitios por los que ha optado la universidad facilitan el acercamiento e interacción de los alumnos con el docente y el acceso a los materiales y fuentes de información, sin embargo, existen áreas de oportunidad en la formación técnica del área de ingeniería ya que es una carrera que requiere un contacto y la manipulación de herramientas y materiales.

Una de esas áreas son los laboratorios de las diferentes disciplinas dado que estos espacios son elaborados con el fin de producir fenómenos físicos que comprueben lo visto en clase teórica.

El propósito de este proyecto es abarcar de forma integral la experiencia y necesidades de alumnos y docentes, ello creando un ambiente que centralice las simulaciones requeridas por las prácticas, así como las actividades, el desarrollo de fórmulas y ecuaciones que den soporte a los conocimientos teóricos.

El objeto a entregar consiste en una aplicación disponible para ejecutar en dispositivos móviles y computadoras de escritorio, que presente una versión digital e interactiva de las prácticas, tanto de los fundamentos teóricos previos, las simulaciones interactivas de los instrumentos y practicas en tercera dimensión, y un espacio de resolución de ejercicios

Objetivos

General

Construir una plataforma disponible para escritorio y dispositivos móviles que funcione como un ambiente integral para alumnos y docentes que permita realizar las actividades concernientes a las prácticas del laboratorio de termodinámica, utilizando practicas y herramientas modernas, escalables y preferentemente de código abierto que puedan ser desplegadas de forma automática y en cualquier infraestructura que así se determine.

Específicos

- Construir un esquema de base de datos relacional que almacene aspectos importantes de la plataforma, tales como usuarios, recursos externos, metadatos como las prácticas y las calificaciones por alumno, el semestre en curso, etc.
- Establecer usuarios en el servidor de base de datos específicos para lectura, escritura, modificación y borrados de los datos, así como un administrador.
- Poner a disposición de la plataforma los datos concernientes mediante la definición de rutas en una API que controle los aspectos de lectura, escritura y modificación de datos (el borrado se reserva para el usuario administrador).
- Respetar los estándares de la UNAM (https://www.ingenieria.unam.mx/protocoloTI/herramientas_colaboracion.html) en términos de integración con plataformas como Google Classroom así como tratamiento y protección de datos.
- Construir las vistas respectivas para la interacción de los usuarios con la página (una vista de “login, una de registro, una “landing page” dependiendo del tipo de usuario que acceda, una para acceder a la visualización y realización de las prácticas, un panel de administración, entre otras)
- Programar los modelados en tercera dimensión de todas las practicas, procurando que se visualicen de forma adecuada los instrumentos, los materiales, las mediciones, las interacciones y el énfasis en el fenómeno físico al cual está enfocado la practica.

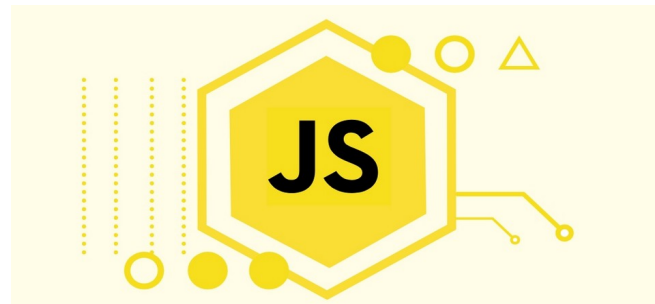
- Desarrollar un espacio para el desarrollo de formulas matemáticas que den el soporte teórico a la visualización.
- Atender de forma dinámica los requerimientos y sugerencias que se lleguen a presentar sobre la marcha del proyecto.

Infraestructura y software a utilizar

Lenguaje de programación principal para el frontend: Javascript

Javascript es el lenguaje estandarizado para su ejecución en navegadores web, su principal ventaja frente a otros lenguajes de programación es que es independiente de la plataforma en que es ejecutado siempre que esta tenga

acceso a internet, posee librerías y utilidades soportadas por una amplia comunidad de desarrolladores que pueden ser integradas en el proyecto, tales como la renderización de gráficos en tercera dimensión, gratificación, realización de planos cartesianos, desarrollo de formulas y matemáticas computacionales.



Lenguaje de programación principal para el backend: Python

Python es un lenguaje de programación interpretado de multipropósito, posee múltiples funcionalidades y librerías para manejo de servidores y datos que acortan el tiempo de desarrollo de forma drástica



Gestor de base de datos: MongoDB

MongoDB es un gestor de base de datos no relacional, cuya ventaja es su alta velocidad y su forma de coleccionar datos que se adapta de forma adecuada a la estructura de datos diccionario en python y objeto en javascript, lo que permite una comunicación sin mayores complicaciones en ambas tecnologías.



Manejo de backend y peticiones al servidor: FastAPI

FastAPI es una librería para python, que pone a disposición información en la base de datos de acuerdo a criterios de búsqueda y endpoints, devuelve y recibe la información en forma de colecciones, hace procesos de validación y gestiona la comunicación mediante el estándar OpenAPI, dicho paradigma lo separa por completo el frontend siendo capaz de cambiar el diseño y trabajarlo de forma paralela sin afectar el flujo de información



Frontend escritorio: VueJS

VueJS es una librería de javascript enfocada en la creación de interfaces de usuario reactivas, es decir que cambian a lo largo del tiempo dependiendo de las acciones del usuario donde dichas acciones desencadenan las peticiones de datos a servidor, gestiona las rutas y el almacenamiento de lado del usuario.



Diseño y modelado en 3D: Blender

Blender es un software enfocado al modelado y animación en 3ra dimensión, proporcionando un ambiente integrado con múltiples herramientas de edición de

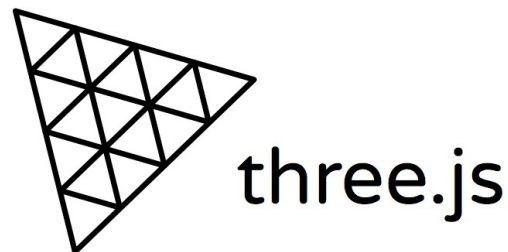


figuras, texturas, movimientos, etc. En este software se desarrollan las simulaciones de las practicas a nivel de animación bruta, es decir, acciones que no estan relacionadas con su análogo de fenómeno físico.

Importación y manejo de modelos 3D:

ThreeJS

Para integrar las simulaciones y programar su funcionamiento en el sitio web hacemos uso de la librería threejs, recibe data de figuras 3D en formato .gltf y las renderiza en el navegador; ademas ayuda a detectar las acciones del usuario que permiten la interacción con la simulación.



Desarrollo de formulas matemáticas en la plataforma: Katex

Renderiza formulas matemáticas mediante la sintaxis del lenguaje latex



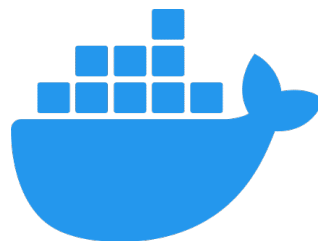
Desarrollo de planos cartesianos y gráficos de otros tipos: D3.js

Renderiza multiples tipos de gráficos en 3 y 2 dimensiones, entre ellos los planos cartesianos necesarios para realizar regresiones lineales y raíces medias cuadráticas



Contenerización de la aplicación: Docker

Docker es un software desarrollado con el fin de homologar un ambiente de desarrollo y de



docker®

despliegue eliminando cualquier inconveniente en términos de dependencias y compatibilidad. Integra todos los componentes de software en un “contenedor” capaz de ser eliminado, modificado y creado sin afectar el ambiente local.

Infraestructura para el despliegue: Capa gratuita de Google Cloud Platform para las pruebas.

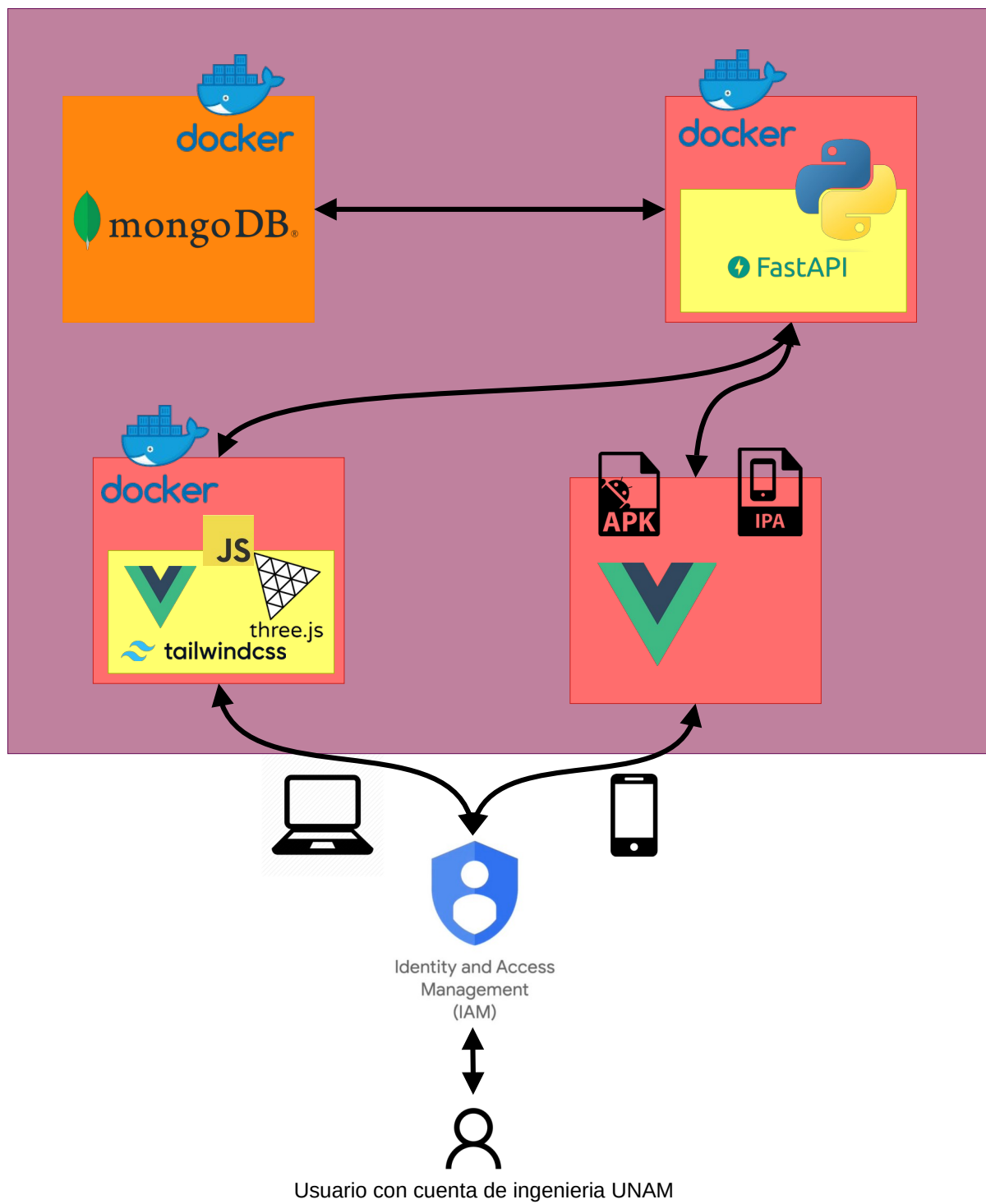
La UNAM tiene servicios de tecnologías de la información integrados en la infraestructura de nube de google, lo que nos permitirá realizar despliegues de forma escalable, así como poder usar su sistema de autenticación el

cual se encuentra ligado con el correo institucional de la Facultad de Ingeniería que para en quien en ultima instancia se desarrolla el proyecto



Google Cloud Platform

Stack completo de desarrollo



Interpretación del stack de desarrollo

El mapa previamente plasmado representa el flujo de datos que existirá cuando la aplicación se encuentre en fase productiva.

En primera instancia se encuentra un usuario, dicho usuario posee un dispositivo con conexión a internet, ya sea un móvil o una computadora, además posee un correo con dominio de la Facultad de Ingeniería de la UNAM con el cual puede identificarse como miembro de la comunidad, alumno o docente.

Habiendo superado el paso de autenticación, dependiendo el dispositivo del usuario se accederá a la página web o a la interfaz de la aplicación, donde interactuará con toda la aplicación web o móvil.

Dicha interacción es gestionada por los elementos del frontend, desarrollados con el lenguaje javascript teniendo integrada la librería de interfaz reactiva Vue.js, a su vez, se acoplan dependencias que cumplen con las funcionalidades requeridas para la página como renderización de gráficos en tercera dimensión, renderización de fórmulas matemáticas, realización de planos cartesianos, responder ejercicios, cuestionarios y realizar conclusiones.

Todas las interacciones que impliquen ingreso o salida de información deben ser recibidos y provenir del backend; el backend con esta desarrollado con el lenguaje de programación python, con funcionalidades ampliadas con el framework FastAPI, una API gestiona el flujo de datos mediante peticiones y mantiene la interfaz independiente de los datos, a su vez realiza trabajos de procesamiento y autenticación.

Todos los datos son guardados para su posterior lectura, escritura, actualización y eliminación desde el gestor de base de datos MongoDB.

Procedimiento de modelado de practicas

El modelado de las prácticas para la simulación consta de las siguientes etapas:

Observación de la prácticas

Establecimiento de la consecutividad de las acciones a razón de causa y efecto

Establecimiento de las acciones independientes

Modelado en 3D de los instrumentos independientes

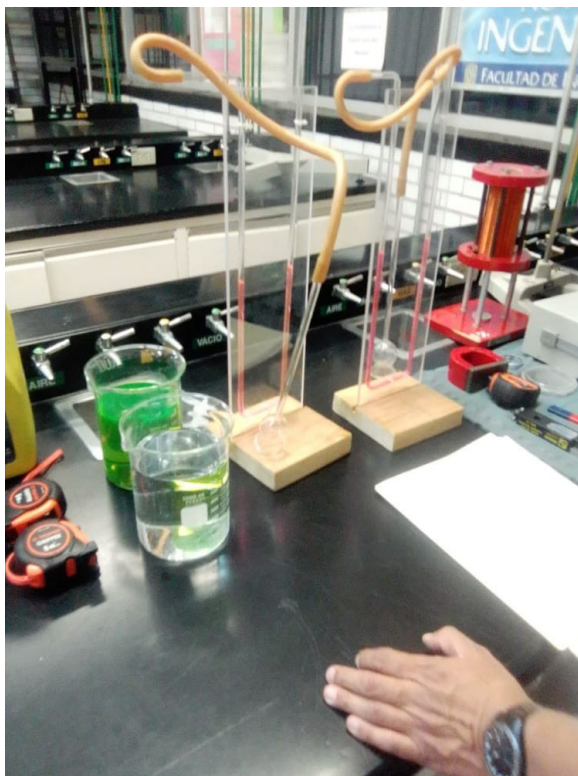
Animación de las acciones independientes

Definición de las formulas pertinentes para la simulación

Programación de la consecutividad

Visualización de la práctica

Ejemplificando con la práctica , el proceso constó primero en visualizar la practica de forma presencial para la toma de evidencia, así como definir el proceso físico que se quiso demostrar, el cual en este caso fue el de la presión.



Establecimiento de la consecutividad de las acciones a razón de causa y efecto

En esta etapa se debe definir por que pasa el fenómeno que se está observando, tanto su explicación a nivel físico como la causa directa que lo provoca.

Para el caso de la práctica 1, el fenómeno físico se resume en el cambio de altura del agua debido al empuje de la columna de aire a medida que se hunde la campana de vidrio en un fluido; dicha altura variará dependiendo la presión absoluta que ejerza el fluido.

Establecimiento de las acciones independientes

Sube y baja la columna de agua

Sube y baja la columna de aire

Sube y baja la campana de vidrio

Cambia el fluido del vaso

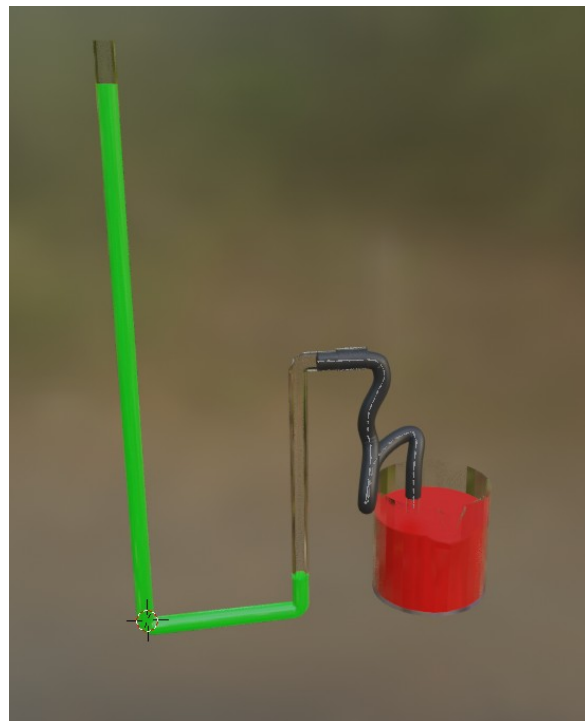
Modelado de los instrumentos

Vaso del fluido

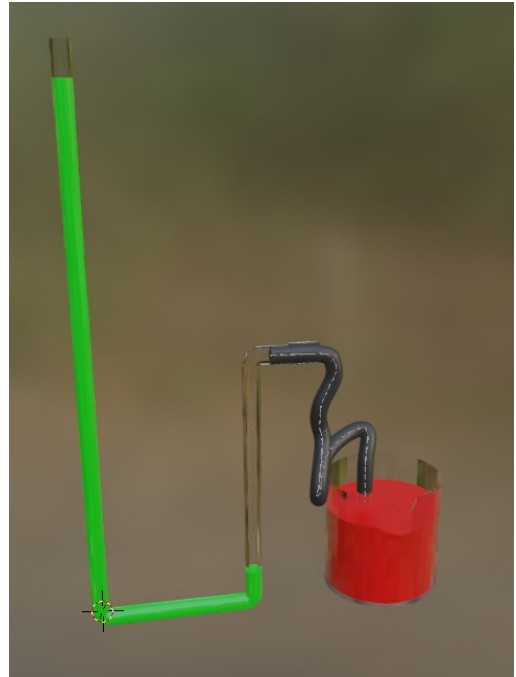
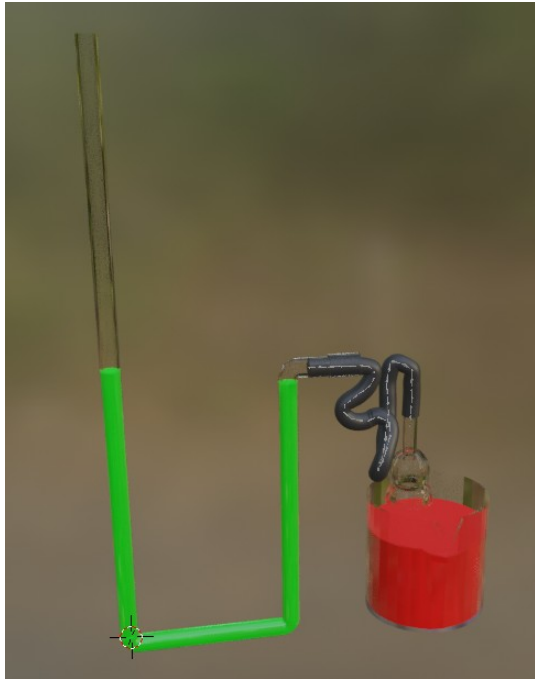
Tubo del fluido

Campana de aire

Manguera



Animación de las acciones independientes



Programación de la consecutividad

Integra la practica al espacio de visualización en la aplicación web

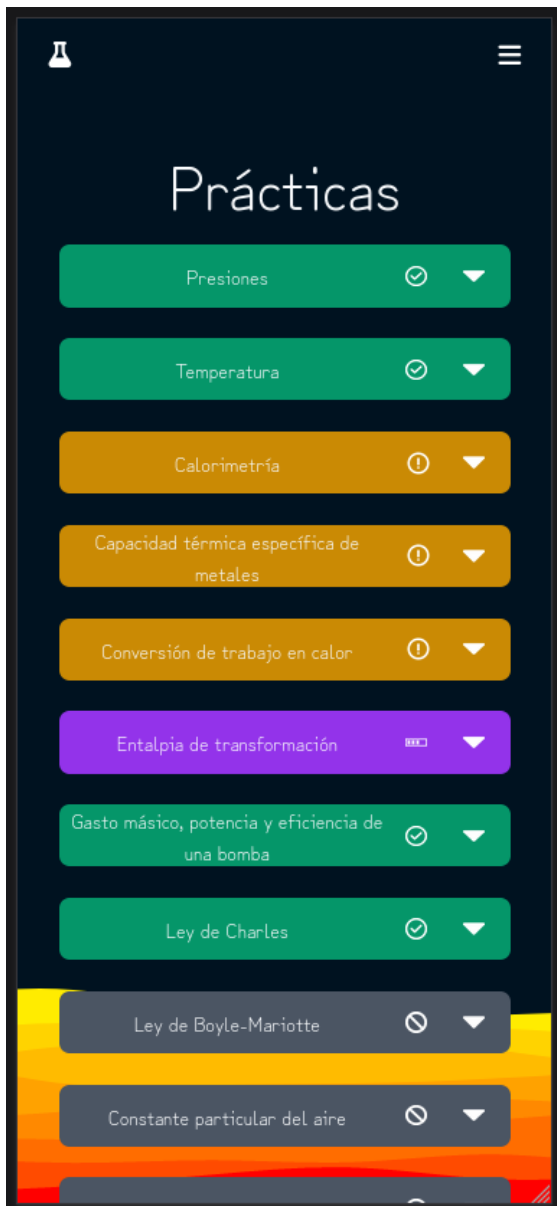


Programación de la interfaz

La interfaz web se encuentra construida en base a la librería Vue.js por medio de una funcionalidad reactiva, lo que implica que las vistas se modifican de acuerdo a sucesos, instrucciones y acciones del usuario o definidas en el servidor.

Con dicho paradigma todos los componentes quedan definidos en plantillas que son llenadas con información traída contextualmente desde el servidor.

Por ejemplo, la pantalla con su listado de practicas está definida de forma dinámica desde el servidor, lo que permite agregar o remover practicas desde un formulario sin afectar la arquitectura de la pagina.



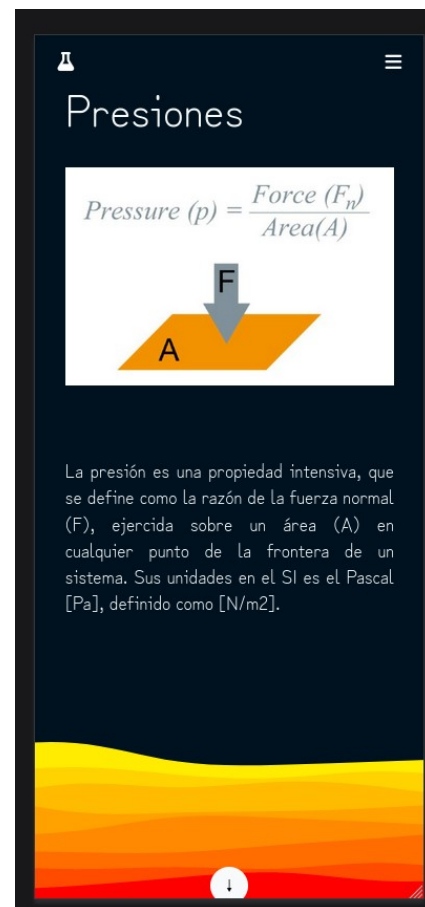
```
practicas : [
  {
    id: 1,
    nombre: "practica_1",
    titulo: "Presiones",
    disponible: true,
    fecha_entrega: new Date().setDate(new Date().getDate()-140),
    subtareas: [
      {
        id_subtarea: 1,
        titulo: "Previo"
      },
      {
        id_subtarea: 2,
        titulo: "Desarrollo"
      },
      {
        id_subtarea: 3,
        titulo: "Conslusiones"
      }
    ],
    mostrar_subtareas: false
  },
  {
    id: 2,
    nombre: "practica_2",
    titulo: "Temperatura",
    disponible: true,
    fecha_entrega: new Date().setDate(new Date().getDate()-90),
    subtareas: [
      {
        id_subtarea: 1,
        titulo: "Previo"
      },
      {
        id_subtarea: 2,
        titulo: "Desarrollo"
      },
      {
        id_subtarea: 3,
        titulo: "Conslusiones"
      }
    ],
    mostrar_subtareas: false
  }
]
```

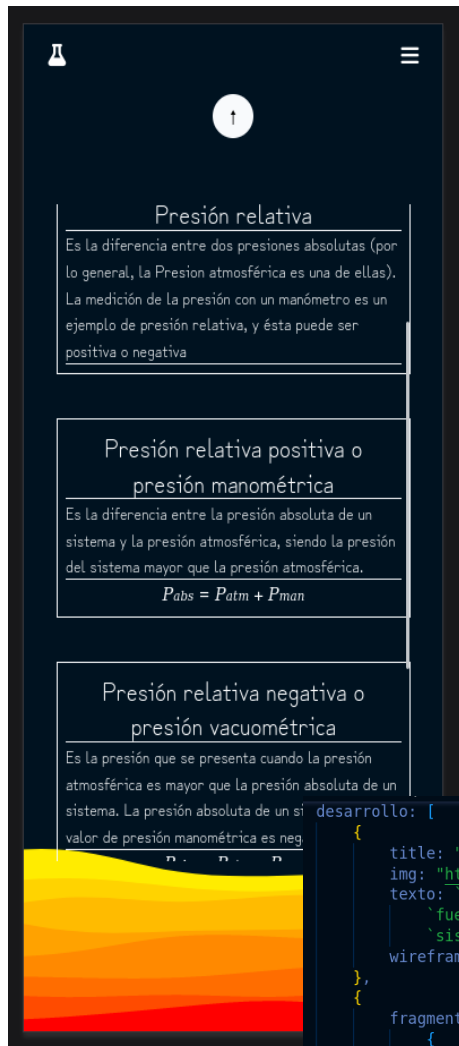
Se debe notar que igualmente, el color de los botones viene definido de acuerdo a la fecha y al avance de la práctica, donde prácticas adentro de tiempo se renderizan de morado, practicas en progreso fuera de tiempo están en amarillo, practicas terminadas están en verde y practicas en gris no están disponibles.

```
avance_practicas: [  
  {  
    id_prac: 1,  
    avance: 100  
  },  
  {  
    id_prac: 2,  
    avance: 100  
  },  
  {  
    id_prac: 3,  
    avance: 70  
  },  
  {  
    id_prac: 4,  
    avance: 28  
  },  
  {  
    id_prac: 5,  
    avance: 70  
  },  
  {  
    id_prac: 6,  
    avance: 63  
  },  
  {  
    id_prac: 7,  
    avance: 100  
  },  
]
```

Esta es una de las ventajas de utilizar una librería reactiva, es decir, se puede definir la lógica en forma de plantillas y después rellenar dichas plantillas con texto normal.

Para la presentación de información teórica se realizó un procedimiento similar





El sistema de despliegue de contenido toma los fragmentos de información contenidos en texto y los presenta en pantalla de acuerdo a las reglas definidas en la programación.

Este sistema de plantillas a pesar de agregar una capa de complejidad al desarrollo, permite integrar contenido de forma dinámica sin que esto implique alterar el código de ninguna forma que sea alterar el desarrollo base de la aplicación

```
desarrollo: {
  {
    title: "Presiones",
    img: "https://static-int.testo.com/media/dc/fd/8313e81a8482/Pressure-Formula-1050x700px-EN_im4c.jpg",
    texto: "La presión es una propiedad intensiva, que se define como la razón de la '+
      'fuerza normal (F), ejercida sobre un área (A) en cualquier punto de la frontera de un '+
      'sistema. Sus unidades en el SI es el Pascal [Pa], definido como [N/m2].",
    wireframe: 0
  },
  {
    fragments: [
      {
        header: "Presión absoluta",
        text: "Es la presión en un punto determinado del sistema '+'
          'que se mide con respecto a la presión cero absoluto'
      },
      {
        header: "Presión relativa",
        text: "Es la diferencia entre dos presiones absolutas (por lo '+'
          'general, la Presion atmosférica es una de ellas). La medición de la presión con un manómetro es '+'
          'un ejemplo de presión relativa, y ésta puede ser positiva o negativa'
      },
      {
        header: "Presión relativa positiva o presión manométrica",
        text: "Es la diferencia entre la presión absoluta de un sistema '+'
          'y la presión atmosférica, siendo la presión del sistema '+'
          'mayor que la presión atmosférica.',
        attachment_type: "formula",
        attachment: "P_{abs} = P_{atm} + P_{man}"
      },
      {
        header: "Presión relativa negativa o presión vacuométrica",
        text: "Es la presión que se presenta cuando la presión atmosférica '+'
          'es mayor que la presión absoluta de un sistema. La presión '+'
          'absoluta de un sistema cuyo valor de presión manométrica es '+'
          'negativo',
        attachment_type: "formula",
        attachment: "P_{abs} = P_{atm} - P_{vac}"
      }
    ]
  }
}
```

Conclusiones

Si bien el proyecto esta pensado en una situación de emergencia sanitaria (o de cualquier otra índole) que implique la imposibilidad de interactuar con los instrumentos de forma presencial, es importante recalcar que las ventajas de desarrollar un software con estas características sobrepasan las circunstancias para las cuales fue creado, al ser concebido como un entorno integral permite la integración de múltiples tecnologías que facilitan la interactividad de todas las actividades lo que a final de cuentas resulta en un desarrollo didáctico de mejor calidad y una experiencia en el aula mas completa para alumno y profesor.

Este desarrollo en su conjunto se realiza con herramientas y procedimientos de uso generalizado en la industria de la creación de software por lo que su mantenimiento y eventual expansión a mediano y largo plazo por personal que se acerca por primera vez al proyecto sin complicaciones de índole técnica se encuentra garantizado.

Tomando en cuenta la metodología con la que se aborda el proyecto se logra abstraer a un entorno virtual todo el proceso que implica la realización de una práctica en el laboratorio de termodinámica, sintetizándolo en los pasos de adquisición de conocimientos previos (necesarios para realizar la práctica), la toma de datos producto de la interacción don los instrumentos, la comprobación por medio de ejercicios teóricos y la realización de conclusiones; todos los cuales son cubiertos en el desarrollo de este entorno.