ÁLVARO BARREIRO ABAD

IES ABASTOS

Ciclo Superior en Desarrollo de Aplicaciones Multiplataforma

CURSO 2022/2023

GRUPO 7U

26 de junio de 2023

TUTOR: RAFAEL PUIG POZO

BOOKING CLASSROOM

Aplicación para de reserva de aulas para profesores

Contenido

[1 PRESENTACIÓN 2](#_Toc138106746)

[2 IDENTIFICACIÓN, JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO. 3](#_Toc138106747)

[2.1 IDENTIFICACIÓN Y JUSTIFICACIÓN. 3](#_Toc138106748)

[2.2 OBJETIVOS. 3](#_Toc138106749)

[3 DISEÑO DEL PROYECTO 5](#_Toc138106750)

[3.1 ANÁLISIS DE LAS TECNOLOGÍAS DISPONIBLES 5](#_Toc138106751)

[3.2 ANÁLISIS DE LAS FASES DEL PROYECTO 7](#_Toc138106752)

[4 DESARROLLO DEL PROYECTO 7](#_Toc138106753)

[5 EVALUACIÓN Y CONCLUSIONES FINALES 7](#_Toc138106754)

[6 REFERENCIAS 7](#_Toc138106755)

# PRESENTACIÓN

La gestión de la reserva de aulas en entornos educativos puede ser un desafío complejo. En muchos colegios e institutos, la organización interna se ve afectada por problemas relacionados con la asignación de espacios disponibles. Los profesores a menudo se encuentran con situaciones frustrantes, como la falta de aulas disponibles cuando necesitan realizar exámenes o actividades académicas. Para abordar esta problemática, se ha desarrollado una aplicación web llamada Booking Classroom.

Booking Classroom es una solución diseñada para facilitar la reserva de aulas por parte de los profesores. Esta aplicación ofrece un sistema centralizado que simplifica el proceso de reserva, permitiendo a los profesores acceder a la disponibilidad de aulas de manera eficiente y realizar sus reservas de forma sencilla.

La aplicación proporciona una interfaz intuitiva que permite a los profesores realizar reservas individuales y en un futuro permitirá programar reservas recurrentes. Además, ofrece una visión general en tiempo real de la disponibilidad de las aulas, evitando conflictos de horarios y optimizando la utilización de los espacios.

Al centralizar la información y agilizar el proceso de reserva de aulas, Booking Classroom ayuda a mejorar la organización interna de los colegios e institutos.

Por tanto, Booking Classroom tiene como objetivo ofrecer una solución práctica y eficiente para la gestión de la reserva de aulas en entornos educativos.

# IDENTIFICACIÓN, JUSTIFICACIÓN Y OBJETIVOS DEL PROYECTO.

## IDENTIFICACIÓN Y JUSTIFICACIÓN.

En entornos educativos como colegios e institutos, la gestión de la reserva de aulas presenta desafíos significativos. Uno de los problemas recurrentes es la falta de disponibilidad de aulas en momentos clave, como durante la planificación de exámenes o la organización de actividades académicas. Esta situación dificulta la programación eficiente de las actividades docentes y afecta negativamente la organización interna de las instituciones.

Para abordar esta problemática, se ha desarrollado un proyecto enfocado en la creación de una aplicación web llamada Booking Classroom. El objetivo principal de este proyecto es proporcionar una solución práctica y eficiente para la reserva de aulas, permitiendo a los profesores realizar reservas de manera sencilla y asegurando una programación eficiente de las actividades académicas.

## OBJETIVOS.

Objetivos Funcionales:

1. Facilitar la reserva de aulas:
   * Proporcionar a los profesores una interfaz intuitiva y fácil de usar para realizar reservas de aulas.
   * Simplificar el proceso de reserva, eliminando la necesidad de comunicarse directamente con el personal administrativo o entre profesores.
   * Agilizar el acceso a la disponibilidad de las aulas, permitiendo a los profesores encontrar rápidamente espacios disponibles para sus actividades académicas.
2. Evitar conflictos de horarios:
   * Ofrecer una visión general en tiempo real de la disponibilidad de las aulas, evitando la superposición de reservas y conflictos de horarios.
   * Evitar que los profesores puedan reservar un aula si ya está ocupada para el horario deseado, garantizando una programación eficiente.
3. Administrar las aulas:
   * Los administradores de la aplicación podrán crear y gestionar las aulas disponibles en el sistema.
   * La información de las aulas y las reservas se almacenará en una base de datos, facilitando el acceso y la gestión para el personal administrativo de manera centralizada.
4. Informar sobre los espacios:
   * Proporcionar información sobre la capacidad y características de cada aula, permitiendo una asignación más precisa y adecuada según las necesidades específicas de cada actividad. Por ejemplo, especificando si el aula dispone de proyector u ordenadores.

Objetivos Técnicos:

1. Implementar un sistema de autenticación utilizando Angular y Firebase:
   * Crear un sistema de registro y autenticación seguro para los profesores y el personal administrativo.
   * Integrar Firebase Authentication para la autenticación y autorización de usuarios.
   * Permitir la recuperación de la contraseña, así como, el cambio de la misma.
2. Diseñar y desarrollar una interfaz intuitiva en Angular:
   * Utilizar Angular para diseñar una interfaz de usuario atractiva y fácil de usar.
   * Implementar componentes y módulos reutilizables para agilizar el desarrollo y mejorar la mantenibilidad del código.
3. Utilizar Firebase Firestore como base de datos:
   * Diseñar y configurar una estructura de base de datos eficiente para almacenar información sobre las aulas, reservas y horarios.
   * Implementar operaciones CRUD (crear, leer, actualizar y eliminar) para gestionar la información de las aulas, así como, de las reservas de aulas.
   * Aprovechar las capacidades de consultas en tiempo real de Firestore para mostrar la disponibilidad de las aulas de manera dinámica.
   * Almacenar los datos de los usuarios que tengan rol de administrador.
4. Implementar funcionalidades de disponibilidad en tiempo real:
   * Utilizar Firestore para mostrar la disponibilidad de las aulas en tiempo real.
   * Actualizar automáticamente la información de disponibilidad cuando se creen nuevas aulas o se realicen nuevas reservas o se cancelen reservas existentes.
5. Configurar Firebase Hosting para el alojamiento de la aplicación:
   * Configurar Firebase Hosting para alojar la aplicación web y garantizar su disponibilidad en línea.

# DISEÑO DEL PROYECTO

## ANÁLISIS DE LAS TECNOLOGÍAS DISPONIBLES

En la fase de diseño del proyecto, es importante considerar las diferentes soluciones y herramientas disponibles en el mercado que nos permitirán desarrollar la aplicación Booking Classroom de manera efectiva. Además, es fundamental identificar las tecnologías que se pueden utilizar para ofrecer los servicios requeridos.

En cuanto a las soluciones y herramientas disponibles, existen varias opciones que pueden ser consideradas. Por ejemplo, hay plataformas y aplicaciones especializadas en la gestión de reservas y programación de espacios. Es cierto que algunas de las cuales pueden ser adaptadas y personalizadas para satisfacer las necesidades específicas de reserva de aulas en entornos educativos. Estas soluciones comerciales pueden ofrecer características avanzadas y estar respaldadas por un soporte técnico adecuado, pero por contrapartida tienen que suelen ser servicios de pago. Como ejemplo podemos citar a *clearooms[[1]](#footnote-1)* o *kalena[[2]](#footnote-2).*

Por otro lado, también se podría explorar el uso de herramientas como Google Calendar[[3]](#footnote-3), aunque su adaptación para la gestión de reservas de aulas en entornos educativos podría resultar laboriosa y limitada en funcionalidad. Además, existe el riesgo de que la configuración manual genere errores y dificulte la eficiencia del proceso.

En este contexto, hemos optado por desarrollar una aplicación personalizada. Para ello en el mercado existen diferentes herramientas de desarrollo de software disponibles, como frameworks y bibliotecas, que pueden facilitar el proceso de desarrollo de la aplicación.

En el lado del front-end, algunas opciones populares son Angular, React y Vue.js, que son frameworks de desarrollo web ampliamente utilizados y que ofrecen un conjunto de herramientas y componentes para la creación de interfaces de usuario interactivas y modernas.

En el ámbito del back-end, se presentan dos opciones distintas: las bases de datos relacionales y las bases de datos NoSQL. En el caso de las bases de datos relacionales, se pueden utilizar tecnologías consolidadas como MySQL o PostgreSQL, en combinación con lenguajes de programación ampliamente adoptados como Java, Python o PHP. Estas tecnologías ofrecen un enfoque tradicional y robusto para la gestión y manipulación de datos estructurados.

Por otro lado, las bases de datos NoSQL ofrecen un enfoque alternativo y moderno. Entre las opciones más populares se encuentran Firebase Firestore, MongoDB y Cassandra. Estas bases de datos NoSQL proporcionan flexibilidad en el esquema de datos y permiten el almacenamiento y procesamiento eficiente de grandes volúmenes de información distribuida.

Después de analizar las diferentes opciones disponibles, hemos tomado la decisión de desarrollar la aplicación Booking Classroom utilizando Angular y Firebase. Estas opciones nos brindan flexibilidad, escalabilidad y facilidad de integración. A continuación, justificaremos nuestra elección:

Angular:

* Amplia comunidad y soporte: Angular cuenta con una gran comunidad de desarrolladores activos y una extensa documentación. Esto nos brinda acceso a recursos valiosos, tutoriales, ejemplos y soluciones a problemas comunes. Además, al ser mantenido por Google, se beneficia de actualizaciones y mejoras constantes.
* Otra razón importante que respalda nuestra elección de Angular es su facilidad para crear templates y darles funcionalidad. Angular ofrece un enfoque claro y estructurado para la construcción de la interfaz de usuario, lo que nos permite separar de manera efectiva la lógica del negocio de la presentación visual. Utiliza el patrón de arquitectura MVC (Modelo-Vista-Controlador) y promueve las buenas prácticas de desarrollo, lo que nos ayuda a mantener un código limpio, estructurado y fácilmente mantenible.
* Otra de las razones para elegir este framework es que es el que estoy utilizando en las prácticas de empresa. Este hecho genera la posibilidad de adquirir experiencia y conocimiento en su uso.
* Angular proporciona diversas herramientas y componentes predefinidos, como Angular Material, que facilitan la creación de una interfaz de usuario interactiva y amigable. Además, Angular nos permite desarrollar componentes personalizados, los cuales pueden ser reutilizados, siendo muy fácil organizar la aplicación. Mediante el uso de componentes, podemos crear templates reutilizables y modulares que encapsulan la lógica y la apariencia de elementos específicos de la interfaz. Además, Angular proporciona enlaces de datos bidireccionales que facilitan la sincronización automática entre la vista y el modelo de datos subyacente.

Firebase:

* Autenticación de usuarios: Firebase proporciona un sólido sistema de autenticación de usuarios, lo que nos permite gestionar de manera segura el acceso y la identificación de los profesores que utilizan la aplicación. Con Firebase Authentication, podemos implementar opciones de inicio de sesión con correo electrónico, autenticación social y más.
* Base de datos en tiempo real: Firestore, la base de datos en tiempo real de Firebase, nos brinda una solución escalable y flexible para almacenar y sincronizar la información de las reservas de aulas. Esto garantiza que los datos estén siempre actualizados y disponibles para los profesores en tiempo real.
* Integración sencilla: Firebase se integra fácilmente con Angular, lo que facilita la configuración y el desarrollo de la aplicación. Además. Esta integración viene favorecida por el hecho de que ambas herramientas son proporcionadas por Google.
* Hosting: Además de lo mencionado, Firebase ofrece un servicio de hosting que nos permite desplegar fácilmente la aplicación y garantizar su disponibilidad en la web.

## ANÁLISIS DE LAS FASES DEL PROYECTO

En esta fase, se identificarán los pasos clave necesarios para llevar a cabo el proyecto.

Fase 1: Análisis del problema.

Se realizará un análisis exhaustivo del problema a resolver y de los requisitos del proyecto. Se identificarán las necesidades de los usuarios y se definirán los objetivos a alcanzar. También se decidirá la tecnología a utilizar.

Fase 2: Definición de los datos necesarios.

Se determinarán los datos necesarios para el funcionamiento de la aplicación. Esto implicará identificar los tipos de información requeridos y cómo se almacenarán y gestionarán.

Fase 3: Estructura del proyecto.

Se establecerá una estructura organizada de carpetas y archivos para el desarrollo del proyecto. Esto facilitará la gestión y el mantenimiento del código.

Fase 4: Diseño gráfico de las pantallas.

Se realizará el diseño visual de las interfaces de usuario de la aplicación. Se crearán los mockups o prototipos de las pantallas para definir la apariencia y la disposición de los elementos.

Fase 5: Codificación.

Se procederá a la implementación de la lógica y funcionalidades de la aplicación.

Fase 6: Pruebas.

Se realizarán pruebas exhaustivas para verificar el correcto funcionamiento de la aplicación. Se identificarán y corregirán posibles errores y se asegurará que cumpla con los requisitos establecidos.

Fase 7: Despliegue.

Una vez que la aplicación haya pasado las pruebas, se procederá a su despliegue en un entorno de producción.

## METODOLOGÍA DE TRABAJO

En cuanto a la hora de cómo enfocar y gestionar el trabajo, se ha elegido utilizar la metodología Agile en conjunto con Kanban.

La metodología Agile es un enfoque de gestión de proyectos que se basa en la colaboración, la adaptabilidad y la entrega incremental. En lugar de planificar exhaustivamente todo el proyecto desde el principio, Agile se centra en ciclos de desarrollo más cortos, conocidos como "iteraciones" o "sprints".

La elección de utilizar esta metodología se fundamenta en varias razones. En primer lugar, nos permite realizar entregas incrementales y frecuentes de funcionalidades, lo que nos brinda la oportunidad de obtener feedback temprano y realizar ajustes en el desarrollo del proyecto. Esta flexibilidad nos permite adaptarnos a medida que avanzamos y garantizar que la solución se ajuste a las necesidades que se han detectado.

Otro aspecto importante de Agile es su enfoque iterativo e incremental. En lugar de esperar a tener la aplicación completa antes de realizar entregas, Agile propone la entrega de funcionalidades en ciclos cortos y frecuentes. Esto nos permite obtener resultados tangibles en cada iteración.

Por otro lado, Kanban es una metodología visual de gestión de proyectos y procesos que se centra en la visualización del flujo de trabajo y la optimización del rendimiento. Se basa en la idea de utilizar tableros visuales con tarjetas o notas adhesivas para representar las tareas y el progreso de un proyecto.

El método Kanban se originó en el ámbito de la fabricación, específicamente en Toyota, y luego se adoptó en el desarrollo de software y otros sectores. La palabra "Kanban" significa "tarjeta" o "etiqueta" en japonés, y hace referencia a las tarjetas utilizadas para representar las tareas en el tablero.

El tablero Kanban consta de columnas que representan las diferentes etapas del flujo de trabajo, como "Por hacer", "En progreso" y "Finalizado". Cada tarea se representa mediante una tarjeta o nota adhesiva, y se mueve de una columna a otra a medida que avanza en el proceso. Esto permite tener una visión clara y visual del estado de cada tarea y del flujo general del trabajo.

Gráfico, Gráfico en cascada

Descripción generada automáticamente

Ilustración 1. Tablero Kanban

Entre las tareas más importantes a llevar a cabo podemos destacar las siguientes:

# DESARROLLO DEL PROYECTO

# EVALUACIÓN Y CONCLUSIONES FINALES

# REFERENCIAS

PLAZOS:

Dia 1: Establecer las bases y funcionalidades de la app.

Dia 2: Instalación del entorno. Subir el proyecto a GitHub. Definir modelo de datos.

Dia 3: Instalación de Angular material. Creación módulo login.

Dia 4: Se añaden validadores a login y mensajes de error. Componente de inicio y Barra de navegación. Routing y lazy-loading.

Dia 4: Se añaden botones a página de inicio.

Dia 5: Se añade formulario de registro de usuarios y un servicio para deshabilitar la barra de navegación cuando se registe un nuevo usuario.

Dia 6: Registro en Firebase.

REQUISITOS FUNCIONALES:

- Login y registro para usuarios y administradores.

- Los datos de registro serán sencillos. Se podrá subir una foto para el perfil.

- Los administradores podrán crear las clases y los horarios disponibles. Así como bloquear las clases de manera fija para el uso lectivo.

- Los usuarios podrán reservar las clases que necesiten en un rango de horas determinado y fecha. Los usuarios también podrán cancelar la reserva en el momento que quieran. No podrán tener dos reservas simultáneas en la misma franja horaria.

- Al iniciar la reserva se indicará el rango horario y la fecha para la que se solicita la reserva. Se mostrarán las clases que coincidan con la búsqueda.

- El administrador creará las clases que tenga el instituto. Se podrá buscar las clases libres que hay en una fecha y hora determinada. También, se podrá ver el detalle semanal de cada clase y ver cuando está libre. Cuando una clase esté ocupada se podrá ver quien la tiene ocupada y por cuanto tiempo.

\* Las clases indicarán el equipo del que disponen (pizarra, ordenadores(número), plazas).

\* Se informará por email a los usuarios de que la reserva se ha efectuado con éxito.

1. Los usuarios podrán indicar que están interesados en reservar una clase la cual está ocupada, de manera que si queda libre le llegue un correo.

DATOS:

Usuario:

Nombre

* + Apellidos
  + Email
  + Teléfono
  + Contraseña
  + Administrador: boolean
  + foto
* Clase:
  + ID
  + Dia
  + Fecha
  + Hora inicio
  + Hora fin
  + Nombre
  + Disponible: boolean
  + Capacidad
  + Equipamiento

REFERENCIAS:

Guía instalación angular: <https://docs.angular.lat/guide/setup-local>

Formularios reactivos: <https://angular.io/guide/reactive-forms>

Angular Flex-Layout: <https://tburleson-layouts-demos.firebaseapp.com/#/docs>

Lazy-loading: <https://www.tutorialesprogramacionya.com/angularya/detalleconcepto.php?punto=85&codigo=85&inicio=80>

Validator Contraseña: https://dev.to/jdgamble555/angular-confirm-password-validation-custom-validator-3pkl

Paginator https://www.tutorialesprogramacionya.com/angularya/detalleconcepto.php?punto=46&codigo=46&inicio=40

1. https://clearooms.com/capterra/?referrer=GetApp [↑](#footnote-ref-1)
2. https://kalena.es/ [↑](#footnote-ref-2)
3. https://www.youtube.com/watch?v=h\_mgc4JC95Q [↑](#footnote-ref-3)