# EventPlanner

Gestión de Eventos

## Alcance

La idea de este proyecto se basa en el desarrollo de una aplicación web para **gestionar la asistencia a eventos sociales** de pequeñas proporciones. Entre los cuales se encuentran comidas, cenas, reuniones familiares, ocio, etc.

El funcionamiento es sencillo: cada usuario puede **crear** los eventos que considere necesarios; enviar una **invitación** del evento a los usuarios que considere oportuno; y éstos, pueden indicar si **asisten o no asisten** a cada evento; de forma que se almacenan los **asistentes en forma de listado**. De este modo, el creador de dicho evento podrá gestionar de forma muy sencilla el **número de asistentes** con los que contará y la **lista de sus nombres**.

Para ello necesitaremos tres **componentes**, principalmente: el **cliente**, el **servidor** y la **base de datos**. El cliente estará pensado para ejecutarse en el lado del usuario, y tanto el servidor como la base de datos, se ha decidido alojarlas en la máquina virtual de la que disponemos en la asignatura.

### Características técnicas

* El nombre de la aplicación web elegido es **EventPlanner**.
* Para el diseño del portal web se utilizará Bootstrap junto con HTML/CSS y Javascript. Se han decidido estas tecnologías por lo siguiente:
  + **Bootstrap:** es una herramienta de prototipado rápido para clientes web, por lo que nos ahorrará una cantidad considerable de tiempo a la hora de implementar nuestro cliente. Entre sus ventajas, nos permite crear proyectos receptivos y móviles de forma sencilla y completamente integrada con HTML, y es de código abierto. Su uso genera resultados rápidos y visualmente elegantes.
  + **Javascript:** Es necesario en el lado del cliente para añadir funcionalidades en el cliente web sin tener que recargar la página completa. Hay un conjunto grande de librerías en este lenguaje que nos facilitarán el trabajo de implementación.
* El portal web se comunicará con el servidor web mediante una **API** (backend) desarrollada en el lenguaje de programación **Node.js**. Esta API será tipo **REST**. Se ha decidido esta tecnología por lo siguiente:
  + **Node.js:** Este framework es conocido por los desarrolladores y es una alternativa muy sencilla ante la mayoría de lenguajes y frameworks disponibles. Se tuvo en consideración el uso de Python con la librería Flask, pero se descartó porque, a pesar de ser muy similar a la anterior, Node.js se escribe en Javascript, al igual que nuestro cliente, y ofrece una API estable para la librería LevelDB que utilizaremos de base de datos.
* Toda la información sobre los usuarios, los eventos, etc. se gestionará mediante una base de datos llamada **LevelDB** en forma local en la máquina virtual. Se ha decidido esta tecnología por lo siguiente:
  + **LevelDB:** Partiendo de que los desarrolladores conocen Javascript y Node.js y que estos dos son compatibles con la librería LevelDB, nos hemos decantado por utilizar esta librería por su sencillez de uso y buen rendimiento. Los datos que vamos a utilizar no necesitan de una estructura relacional ya que eso ralentizaría el sistema, sin embargo LevelDB tiene una estructura interna de pares clave-valor, lo cual se adapta correctamente a nuestras necesidades. Otro factor a su favor es que al menos la mitad de los desarrolladores ya conocen esta librería.

## Recursos

Para la implementación del sistema propuesto se hará uso de **una de las máquinas virtuales proporcionadas** para esta asignatura como principal medio de almacenamiento y despliegue. La máquina virtual citada también hará la función de servidor y contendrá la base de datos, como ya hemos comentado antes. Consideramos que son recursos suficientes y no disponemos de medios para obtener más (AWS u otras plataformas).

Además de lo anterior, necesitaremos diferentes **editores de texto** o **IDEs** para la programación del proyecto. Puesto que los lenguajes de programación que hemos propuesto son tan sencillos y populares, hay una variedad muy amplia de editores e IDEs disponibles y todos son compatibles entre sí. Con el fin de que los desarrolladores estén lo más cómodos posibles durante el desarrollo, se les deja a su elección tal herramienta.

## Tiempo

Estimamos una primera demostración de funcionalidad para el día 22 de abril, miércoles. Estimamos también la finalización del proyecto para la última semana de mayo. La fecha de finalización última queda pendiente de concretarse más, mediante una estimación de tiempo más detallada.

## Metodología

Seguimos una metodología basada en la **metodología ágil** junto con **kanban**.

Estamos siempre en contacto mediante un **servidor de Discord** exclusivamente dedicado a este proyecto. En ese servidor hemos habilitado dos chats de texto y uno de voz. De este modo la comunicación entre los integrantes del proyecto es cercana y continua. Hemos elegido Discord por ser una plataforma fiable, sencilla y con un amplio abanico de posibilidades para comunicarse.

El **chat de texto principal** (llamado simplemente chat) es el que se utiliza para comunicación general. El **chat de texto secundario** (llamado notificaciones) se utiliza con el único fin de que cada desarrollador notifique al resto del equipo los eventos que van ocurriendo, por ejemplo, el completar una tarea importante. El **chat de voz** se utiliza para realizar reuniones. Es posible, en caso de necesitarlo, activar webcam o compartir pantalla.

Las **reuniones** son solicitadas cuando al menos uno de los desarrolladores lo vé necesario, para resolver dudas, problemas o demás incidencias. Además de programar una reunión cada dos semanas (día y hora a concretar según disponibilidad) para hacer un seguimiento global del proyecto y poner al día a los integrantes del proyecto.

También incorporamos a nuestra metodología un **tablón de tareas**, extraído de la idea de kanban. La herramienta que implementa esto es **Trello**, con la que podemos crear tarjetas y organizarlas de una forma muy intuitiva. Además Trello es una herramienta ya conocida por los desarrolladores, por tanto no es necesario que inviertan excesivo tiempo en aprender a utilizar una diferente.

Con todo esto tenemos una metodología que refuerza la comunicación entre desarrolladores y ayuda a organizar las tareas de una forma sencilla.