



Instituto Industrial Luis A. Huergo
Desarrollo de Sistemas 2025
Cuatrimestre N°2
Trabajo Práctico N°2



Objetivos

El presente trabajo tiene por objetivo poner en práctica los temas vistos acerca del diseño de sistemas y las tecnologías de backend y frontend. Se espera que los alumnos desarrollen y tengan la autonomía para buscar librerías, documentación y cualquier recurso que consideren necesario para construir su sistema.

Se espera que el sistema funcione en su integralidad y cumpla con los requerimientos pedidos.

Se deberá poder interactuar con el sistema mediante un cliente web en conexión con un servidor, modificando su estado u obteniendo respuestas de él. Además el servidor deberá contar con una serie de tests que validen su funcionamiento.

Sobre el sistema a diseñar y desarrollar

Los alumnos pueden optar por la propuesta planteada debajo sobre el sistema de eventos o pueden proponer otro sistema a diseñar y desarrollar. Para ello, el sistema propuesto deberá cumplir con los **Objetivos** planteados para este trabajo, y en términos generales, tener la misma complejidad que el sistema de eventos. Se deberá acercar la propuesta y ser validada primeramente antes de su puesta en marcha. No se podrán tomar trabajos de otras materias.



Página web de eventos: “Migueventos”



Ante el éxito de Miguel con su negocio gastronómico ahora ha decidido extenderse a un rubro distinto. Se le ocurrió, ante la falta de aplicaciones disponibles, que se podría crear una donde cada uno pueda crear y sumarse a eventos. Esto es: festivales, reuniones temáticas, encuentros barriales, recitales, cumpleaños, casamientos, etc.

Para ello quiere tener una página web donde se puedan crear aquellos eventos.

Se debería contar con un módulo de autenticación: una pantalla de inicio de sesión y de registro de usuario donde se hagan todas las validaciones de datos correspondientes.

El acceso a los eventos podría ser sin ánimo de lucro (gratuito), como un cumpleaños; o con cargo, como un recital. Para ello el usuario debería disponer de una página desde donde pueda cargar dinero al sistema y consultar su saldo disponible, el cuál se debitaría al comprar entradas a un evento pago.

Si el evento es sin cargo el usuario tiene la opción de confirmar asistencia y luego cancelarla. Si el acceso al evento es marcado como pago, puede únicamente comprar las entradas.

El usuario creador de un evento se marcará automáticamente como asistente y a su vez, él solo tendrá la capacidad de cancelarlos.

Las pantallas que se deberían tener indispensablemente son: la pantalla de inicio de sesión y la pantalla de registro de usuario; la pantalla principal donde se listen todos los eventos disponibles de forma minimizada, la cual tendría un buscador y filtros por categoría y por condición de gratuidad; la pantalla de detalle de evento; la pantalla de pago, en caso de evento con cargo o de confirmación de asistencia en el caso contrario; la pantalla de datos de usuario, donde se muestren todos los datos del registro y se pueda ver y modificar su saldo; la pantalla donde se listen los eventos a los que asistirá.

Los datos asociados al usuario son: nombre de usuario, su nombre de pila, apellido, DNI, email, contraseña y saldo disponible. Así como también la cantidad de entradas compradas para cada evento y sus confirmaciones de asistencia.



Los datos asociados a un evento son: título, fecha, descripción breve, descripción ampliada, ubicación, imágenes, si se le suma la condición de pago o es sin cargo, si fue cancelado y el costo de acceso al evento.

Consigna

1. Se debe diseñar y desarrollar una página web y un servidor que cubran todos los requerimientos del cliente. El servidor debe estar hecho con Typescript (Javascript) y debe poder ser ejecutado con [Node.JS](#). Se deberá persistir toda información mediante una integración con una base de datos SQLite. El proyecto debe ser versionado con GIT. Se podrá utilizar el estilo de arquitectura que se crea conveniente: se puede diferenciar en microservicios, se puede pensar en varias capas, etc. El cliente web debe contar con una interfaz visual consistente en todas sus páginas y debe hacer uso de colores, tipografías, ordenadores de contenido, etc. Se podrán utilizar cuantas librerías crean convenientes.
2. Seleccionar 4 casos de uso del sistema y escribir los tests pertinentes para verificar su funcionamiento.

Recomendaciones

Plantear primero todos los requerimientos de forma explícita y derivar de ello los casos de uso; bosquejar pantallas e interfaces visuales; definir cuáles serían los endpoints necesarios para atender a la interacción de esas pantallas; modelar la base de datos en función de los casos de uso y las entidades que se manejan; diseñar tentativamente qué forma tendrá la estructura del backend y del frontend. Y si no queda ninguna decisión de diseño preliminar, comenzar a escribir el código.



Ejes de evaluación

El trabajo será evaluado en cinco medidas distintas: en qué tanto se trabajó durante las clases, en qué grado los requerimientos fueron cubiertos por el sistema, en cuán completo está el trabajo en relación a las consignas, en cuántos de los contenidos vistos hasta la presentación del trabajo fueron aplicados al sistema: patrones de diseño y criterios de diseño de sistemas. Y por último, en la consistencia de la interfaz gráfica a través de las diferentes páginas.

Sobre los equipos

Los grupos deben estar conformados, como máximo, por 3 integrantes; pueden estar formados por menos.

Condiciones de entrega

Sobre el código fuente del punto 1: debe poder ser descargado de Github. Debe haber un archivo [README.md](#) en la carpeta raíz del repositorio que indique quiénes integran al grupo de trabajo, de qué forma se debe compilar el código del servidor y del cliente web, las decisiones tomadas respecto al diseño del sistema y cualquier otra cuestión que crean conveniente observar.

El archivo de la base de datos, que debe ser interpretada por *sqlite*, debe hallarse en la carpeta raíz bajo el nombre “*bd*”.

Todo código que haya sido pusheado al repositorio remoto posterior a la fecha de entrega, no será tenido en cuenta para la corrección.

Momento límite de entrega: 28/10 23:59hs