



# Tecnólogo en Informática

## San José de Mayo

UTU - UTEC - UdelaR

**Curso:** Taller de sistemas de información .NET

**Año:** 2021

**Documento:** Laboratorio Final

**Docente:** Ing. Cristian Bauza

# Objetivos

El objetivo del laboratorio es que los estudiantes se familiaricen con las distintas tecnologías del stack .NET mediante el diseño e implementación de un sistema de información moderno con una arquitectura distribuida utilizando el abanico de tecnologías .NET. Los estudiantes deberán realizar un documento de arquitectura así como un paper basado en un template que les será entregado. Además deberán realizar una presentación final para la defensa. Al final del laboratorio el estudiante deberá ser capaz de:

- Comprender cómo generar una arquitectura distribuida con las tecnologías .NET.
- Familiarizarse con las tecnologías de acceso a datos (Entity Framework, ADO.Net).
- Comprender las principales tecnologías del stack para el desarrollo de servicios.
- Adquirir conocimientos con las tecnologías de desarrollo front-end.
- Ser capaces de realizar el deploy de un sistema distribuido en Azure.

# Problema

Existen diversas instituciones educativas con problemas de control de acceso a sus edificios. Actualmente muchas de ellas no cuentan con un sistema informático para el registro de ingresos a los diferentes locales, asignación de salones, portal de noticias por cada carrera, etc.

# Requerimientos

Se requiere realizar un sistema que pueda ser utilizado en distintas instituciones, y el mismo será ofrecido a los clientes en la modalidad SAAS, es decir que se cobrará una suscripción por el uso del sistema. Considerando lo anterior se listan las principales características con las que debe contar el sistema:

- ~~Manejar un rol SUPERADMIN con la capacidad de crear nuevos clientes (instituciones).~~
- ~~Manejar un rol ADMIN, correspondientes a los administradores de cada institución. Los usuarios con este rol son creados por los usuarios SUPERADMIN.~~
- ~~Manejar un rol PORTERO y un rol GESTOR, ambos pueden ser creados por el ADMIN.~~
- ~~Hay que considerar que una institución puede tener uno o varios edificios, los cuales deben estar georeferenciados. Cada edificio además puede tener uno o varios puntos de acceso (puertas de acceso).~~
- ~~Cada edificio puede tener N salones (o espacios) los cuales pueden ser reservados para un evento que puede ser único o recurrente en un lapso de tiempo.~~
- ~~En las porterías se debe registrar el acceso de cada persona al edificio. Los porteros deben estar asignados a un edificio y seleccionarán en qué puerta están trabajando.~~
- ~~De las personas que ingresan al edificio se debe registrar Tipo y Nro. de documento, apellidos, nombres, teléfono y mail de contacto.~~
- ~~Una vez registrada la persona, no será necesario volver a ingresar sus datos para el registro de acceso.~~
- ~~El sistema debe permitir crear una lista de personas mediante CSV o Excel para precargar datos.~~
- ~~De forma adicional, se requiere que el sistema cuente con una componente con reconocimiento facial, donde una vez registrada la persona bastará con posar frente a la cámara del dispositivo que esté en portería. (Ver:~~

<https://github.com/Azure-Samples/Cognitive-Face-CSharp-sample>,  
<https://azure.microsoft.com/en-us/services/cognitive-services/face/#overview>)

- ~~Es importante que el dispositivo que esté en portería alerte al portero en caso de que no se encuentre coincidencia con una persona registrada.~~
- ~~Las distintas entidades deben tener un log de cambios almacenado en una base de datos NO relacional, es decir, cada cambio que se realice en los datos de un edificio, salón, etc. debe quedar logueado.~~
- ~~El sistema debe contar con un portal de noticias público donde se publiquen las novedades y el calendario de uso de los salones de cada edificio.~~
- ~~Mensualmente el sistema cargará una deuda a cada cliente (institución) por el uso del servicio. Cada institución tiene asociado un producto (que es determinado por el SUPERADMIN) que tiene a su vez una lista de precios con fecha de validez y por tanto la deuda se crea con el precio del producto correspondiente a la fecha.~~
- ~~El sistema le debe mostrar al ADMINISTRADOR de la institución la cuenta corriente con sus facturas y pagos.~~
- ~~Por el momento no se integrará con ningún sistema de factura electrónica, por lo que por cada factura se generará un id interno.~~
- Una vez generada la deuda, los ADMINISTRADORES podrán pagar la misma mediante algún medio electrónico (paypal, mercadopago o similar)
- Se solicita enviar notificaciones vía email en todas las acciones que corresponda (por ejemplo indicando que la factura del mes está disponible para pago). El envío de notificaciones vía email es necesario utilizando alguna plataforma profesional como puede ser twilio, sendinblue o similar.
- Los componentes que correspondan deben estar deployados en azure(web apps, db, apis) si la subscripción de estudiantes lo permite, de lo contrario se brindará acceso a otra plataforma.

## Base de Datos Relacional

Se utilizará Microsoft SQL Server. Para desarrollo se puede utilizar la versión SQL Server Express.

## Back-End

Una de las componentes principales del sistema será el back-end que expondrá los servicios necesarios para el funcionamiento de las interfaces de usuario así como para la conectividad con sistemas externos. Esta componente debe estar diseñada utilizando una arquitectura en capas siguiendo las buenas prácticas de diseño, incluyendo una capa de **acceso a datos**, una capa de **lógica de negocios**, una capa con las **entidades de negocio**, una capa que exponga los **servicios** necesarios, y adicionalmente los **testings unitarios**.

## Front-End

El sistema deberá disponer de una aplicación Web multiusuario que disponga tanto de las funcionalidades de gestión de la empresa como el propio portal público. La tecnología de frontend queda a libre elección.

## Requerimientos Opcionales

- Implementar pipelines de deploy continuo con una herramienta a elección (gitlab, aure devops).

# Entregas y evaluación

- 21/09/2021 Se debe entregar el modelo de datos, un diagrama de arquitectura y un análisis de los casos de uso considerados críticos.
- El trabajo se realizará en grupos de 3 integrantes.
- Deberán realizar la entrega final el día 07/12/2021 (tentativa), dicha entrega contendrá:
  - El paper del trabajo.
  - Capturas de pantalla y un mini video del funcionamiento del sistema.
  - Código fuente de la solución.

# Evaluación del Curso

La aprobación del curso requiere de obtener un 60% del puntaje global compuesto de la siguiente forma:

- 20% Evaluación Continua
- 60% Laboratorio Final
- 20% Presentación del Laboratorio Final