Ticket de Turno

**Requerimientos de desarrollo WEB para Ticket de Turno, individual.**

(Entrega y exposición el lunes 28 de OCTUBRE del 2024 7:00 p.m.) Parcial 2

El proyecto se evaluará con exposición y pruebas de la aplicación de software, que consiste en el desarrollo WEB, denominada "Ticket de Turno".

Ticket de turno ya tiene avances importantes durante las tareas.

**Lo desarrollarán de forma individual. El proyecto se entregará en digital (se sube en Teams en la fecha y hora indicada, TODOS) y se expondrá al profesor de forma presencial los primeros 5 alumnos que decidan exponerlo en clase el día de la entrega, con esto obtendrán un plus de 10 puntos extras, siempre y cuando cubran todos los requerimientos, que inclusive pueden ser acumulables en el proyecto final. Al ser de forma individual, no es posible exposición de todos ya que se llevaría mucho tiempo, a los que no expongan se evalúa el proyecto por parte del maestro y otorga calificación por teams, no hay resta de puntos por no exponerlo.**

Los puntos de revisión para determinar calificación Parcial 2 de la aplicación " Ticket de turno " serán:

1. Totalmente Orientado a Objetos. **(-30)**
2. Arquitectura en MVC, Frameworks como Angular o React, son aceptados. **(-30)**
3. Se aceptan otros lenguajes para el backend(Go, Phyton, C#, Express, Java, .NET etc.) y/o frameworks de Full stack e inclusive otras bases de datos diferentes a MySQL, SQL o NOSQL. **No usar PHP.**
4. Deberá ser responsivo. **(-30)**

Requisitos operacionales:

1. Deberá tener un login de acceso(usuario y contraseña) con control captcha para administradores. **(5 pts.)**
2. El formulario de solicitud de ticket de turno(público) deberá tener la capacidad de registrar y/o modificar el registro en la parte pública por parte de los usuarios, para modificar solicite el CURP y el número de turno que se le asignó. En cada solicitud generé un comprobante de datos básicos con el turno correspondiente de la solicitud registrada, que sea en PDF con formato aceptable. El turno asignado deberá iniciar en 1 por cada municipio, cada municipio llevará su control interno de atención de 1 hasta n. **(10 pts.)**
3. Es muy importante que el comprobante pdf del usuario lleve un código QR, donde se identifique la CURP del alumno solicitante. **(5 pts.)**
4. Tener una opción de consulta por CURP o por nombre, para administradores y que puedan eliminar, modificar, registrar y colocar un estatus de “Resuelto” y “Pendiente”. **(5 pts.)**
5. Si hay catálogos en el modelo de datos, realizar el CRUD correspondiente y operable ara usuarios finales con Front End. **(5 pts.)**
6. Opción para cerrar sesión (Seguridad de la aplicación, parte privada) y Menú para navegar entre opciones. **(5 pts.)**
7. Deberá contener al menos la implementación del uso de estrategia de una API Rest, donde el equipo considere aplicarla. **(10 pts.)**
8. Tablero(dashboard) donde se muestre por medio de gráficas los estatus de las solicitudes (“Resuelto” y “Pendiente”) que pueda ser filtrado por municipio o por el total de solicitudes. **(10 pts.)**
9. Uso de repositorio Git, donde se demuestre la actividad en las tareas asignadas a desarrollar en base a los commits/push/merge etc. **(5 pts.)**
10. Deberá utilizar un ORM para las operaciones sobre la base de datos. **(10 pts.)**
11. Deberá utilizar al menos 1 patrón de diseño, recomendaciones son: “Singleton”,”Adapter”,”Observer” o ”Factory”. **(10 pts.)**

**Se realizará la entrega vía Teams (Angular, React o Vue solo subir la URL del deploy, los demás en ZIP o RAR), evaluación y exposición de forma individual del proyecto WEB.**