**PROYECTO.-**

**“ESTACIONAMIENTO INTELIGENTE”**

**2- REQUISITOS DEL SISTEMA**

**INTEGRANTES:**

Ortiz Medina Blas David

Duarte Guicho Julio Cesar

**Introducción.**

Este proyecto se hace con fines de mejorar la entrada y salida de un estacionamiento, también se llevara un control para el acceso implementando reglas viales, esta proyecto nombrado como “**estacionamiento inteligente”**,su uso será implementado en las instalaciones del tecnológico de Culiacán con la finalidad de ayudar a dicha institución para que haya una mejor seguridad vial y una conducción correcta sin irrumpir reglas que puedan afectar la fluidez de la vía de entrada y salida del estacionamiento de dicha institución.

**Requisitos funcionales**

|  |  |
| --- | --- |
| Número de requisito | RF1 |
| Nombre de requisito | Registrar las altas de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF2 |
| Nombre de requisito | Registrar las bajas de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF3 |
| Nombre de requisito | Registrar las horas de entrada de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF4 |
| Nombre de requisito | Registrar las horas de salida de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF5 |
| Nombre de requisito | Guardar las horas de entrada de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X  X |

|  |  |
| --- | --- |
| Número de requisito | RF6 |
| Nombre de requisito | Guardar las horas de salida de los usuarios |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

**Requisitos no funcionales**

|  |  |
| --- | --- |
| Número de requisito | RF2.1 |
| Nombre de requisito | Seguridad para no poder modificar la información ingresada por personas ajenas al sistema |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF2.2 |
| Nombre de requisito | El sistema debe tener un buen rendimiento en cuanto a tiempos de espera |
| Tipo | Requisito Restricción  X |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X |

|  |  |
| --- | --- |
| Número de requisito | RF2.3 |
| Nombre de requisito | La disponibilidad del servicio del sistema debe ser continua. |
| Tipo | Requisito Restricción  XX |
| Fuente del requisito | Sistema |
| Prioridad del requisito | Alta/Esencial Media/Deseado Baja/Opcional  X  X |

**Casos de uso**

|  |  |
| --- | --- |
| CU1 | Registrar alta de un usuario. |
| Actor | Administrador |
| Objetivo | Registrar la alta de un usuario |
| Pre-condición | -El administrador se identifica  -El administrador ingresa al sistema |
| Flujo básico | 1.-El usuario llega con los requisitos establecidos para ser registrado.  2.-El administrador recibe los requisitos.  3.-El administrador ingresa al sistema.  4.-El administrador ingresa los datos de usuario (identificación (IFE), número de control, comprobante de inscripción de semestre).  5.-El sistema registra al usuario.  6.-El administrador imprime la identificación de entrada para el usuario.  7.- El usuario se va con su identificación de entrada. |
| Flujo alternativo | 3.-El sistema falla.  3a.-El administrador reinicia el sistema, inicia la sesión y solicita la recuperación al estado anterior.  4.-El administrador ingresa mal los datos de usuario (Identificación (IFE), número de control, comprobante inscripción de semestre) al sistema.  4ª.El administrador vuelve a ingresar los datos  6.-El administrador imprimió mal la identificación de entrada para el usuario.  6ª.-Buscar una falla en la impresora |
| Pos-condición | El usuario es dado de alta en el sistema. |

|  |  |
| --- | --- |
| CU2 | Registrar hora de entrada del usuario |
| Actor | Usuario |
| Objetivo | Registrar la hora de entrada del usuario |
| Pre-condición | 1.-El usuario pertenece a la institución  2.-Estar registrado en el sistema.  3.-Tener la identificación de acceso. |
| Flujo básico | 1.-El usuario llega con su identificación.  2.-El usuario muestra su identificación.  3.-El sistema da acceso al usuario a la institución.  3.-El usuario accede a la institución. |
| Flujo alternativo | 1.-El sistema falla.  1ª.- Se presenta un reporte al administrador del sistema.  2.-El usuario no pertenece a la institución.  2ª.- No puede acceder a la institución.  3.-El usuario olvido su identificación.  3ª.- No tiene acceso a la institución. |
| Pos-condición | 1.-El usuario ingresa a la institución. |

**Caso de uso extendido**

|  |  |
| --- | --- |
| CU3 | Guardar las horas de entrada de los usuarios |
| Actor | Usuario |
| Objetivo | Guardarlas horas de entrada de los usuarios |
| Resumen | El usuario llega a la entrada, muestra su identificación y el sistema guarda la hora de entrada del usuario y le da acceso a la institución. |

|  |  |
| --- | --- |
| Acción del actor | Respuesta del sistema |
| 1.-El usuario llega con su identificación. | 2.-El sistema despliega la opción de registro. |
| 3.-El usuario muestra su identificación al sistema. | 4.-El sistema reconoce el código de la identificación. |
| 5.-El usuario espera la autorización de acceso del sistema. | 6.-El sistema guarda la hora de entrada del usuario yle da acceso a la institución. |
| 7.-El usuario accede a la institución. |  |

**2.4 - ACTORES DEL SISTEMA Y ROLES:**

Alumnos: Los alumnos llegaran a la entrada del estacionamiento a solicitar entrada al estacionamiento con su permiso, tienen que ver si se le autoriza la entrada en caso de no tener alguna infracción o si el permiso a caducado.

Empleados: Los maestros tendrán que pasar el permiso de estacionamiento por el scanner y se determinara si puede pasar al estacionamiento, los maestros también pueden ser sancionados en caso de infracción al reglamento.

Servicios Generales: Llevan el control de los datos en un catálogo, ellos toman la captura de las sanciones que se hayan cometido, las cuales serán enviadas desde el sistema en la entrada del por las autoridades de la institución, llevaran un registro de hora, número de sanciones, datos personales, altas y bajas en el sistema.

**3- ARQUITECTURA DEL SISTEMA.**

Arquitectura a implementar: Hemos decidido implementar la arquitectura web, ya que nos proporciona la ventaja de que se puede llevar a cabo en distintas capas, la capa de presentación, la capa de negocio y la capa de datos.

La ventaja principal de esta arquitectura es el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio. En el diseño de sistemas informáticos actuales se suele usar las arquitecturas multilineal o Programación por capas. Además, permite distribuir el trabajo de creación de una aplicación por niveles; cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

API (Application Programming Interface) es el conjunto de funciones y procedimientos o métodos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Ejemplos:

* Microsoft Framework .NET
* OpenGL
* API for SCSI device interfacing
* Javascript-C de Mozilla
* Symfony para PHP

**CAPAS Y NIVELES**

Capa de presentación:

Esta capa es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso.

Aquí mostraremos una interfaz al usuario con la cual tiene una fácil interacción, los usuarios entrarán de manera sencilla al ingresar su número de control en un apartado del sistema, nuestra interfaz le mostrará si es válida su entrada, no podrá ingresar al estacionamiento si el número de control a sido repetido o en caso de que aun no salga del estacionamiento.

Se mostrarán unas pequeñas opciones en la parte posterior de la interfaz, allí se mostrará el tipo de usuario que es, los administradores y usuarios normales tienen distintas opciones en la interfaz.

Los administradores pueden dar de alta y baja de manera sencilla a usuarios, y tienen la opción de crear nuevos administradores, y solo los administradores tienen derecho a ver las horas de entrada y salida de cada uno de los usuarios que ingresaron al estacionamiento.

Capa de negocio:

Aquí es donde, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.

En esta capa el administrador recibe los datos de un nuevo usuario (tanto trabajador como estudiante de la institución), una vez proporcionado los datos validos se le dará de alta en nuestro sistema, en caso de que se proporcionen datos erróneos o que el usuario que esté registrado no se validara dicho registro. Se dará de baja a un usuario si ya no pertenece a la institución o no cumple con las reglas establecidas por la institución.

Capa de datos:

En esta capa residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Nuestra aplicación tiene como sistema gestor de base de datos a MongoDB, es un sistema NoSQL con el cual nos apoyaremos para la realización de nuestra aplicación.