|  |  |
| --- | --- |
| Universidad Autónoma de Coahuila |  |
| **Security Guard Gelocator**  SGC v1.0 |
| Propuesta de Negocio. | |

TABLA DE CONTENIDO

1. Historial de Cambios. 2

2. Propósito del Documento. 3

3. Antecedentes. 3

4. Áreas de Oportunidad de la Situación Actual. 3

5. Propósito del Proyecto. 3

5.1 Metas. 3

5.2 Objetivos. 3

6. Alcance del Proyecto. 3

6.1 <objetivo 1>. 3

6.2 <objetivo 2>. 3

6.3 <objetivo …>. 4

7. Calendario de Actividades. 4

8. Plataforma de Desarrollo. 4

8.1 Ciclo de Vida. 4

8.2 Metodología. 4

8.3 Software. 4

9. Plan de Comunicación. 4

10. Licenciamiento. 5

11. Póliza de Soporte Anual. 5

12. Condiciones Generales del Proyecto. 5

13. Garantía. 6

14. Entregables. 6

15. Inversión. 6

15.1 Costos. 6

15.2 Flujo de Pagos. 6

16. Glosario. 6

# Historial de Cambios.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Rev.** | **Pág.** | **Sección** | **Resumen del cambio** | **Responsable del cambio** | **Aprobó** | **Fecha** |
| PA1 |  |  | Primer borrador del documento | Sergio Armando Suarez Villanueva |  | 18/19/2020 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# Propósito del Documento.

El presente documento tiene como propósito dar a conocer la propuesta de negocio para el desarrollo e implementación del Security Guard Gelocator para la Universidad Autónoma de Coahuila.

En el que tiene por objetivo desarrollar una aplicación móvil con el uso de la localización en tiempo real con doble verificación mediante código QR esto para verificar si los guardias cumplen con éxito sus rondines para salvaguardar la integridad de los estudiantes.

# Antecedentes

Actualmente en el campus Arteaga de la Universidad Autónoma de Coahuila, el personal de seguridad lleva una bitácora acerca del seguimiento y la verificación de los rondines en el perímetro de dicho campus.

# Áreas de Oportunidad de la Situación Actual.

Actualmente el jefe de los guardias no tiene la capacidad de cerciorarse que los guardias de seguridad cumplan con la programación de los rondines, por lo que no hay certeza de que lo guardias cumplan con su trabajo.

# Propósito del Proyecto.

## Metas.

1. 1. Disminuir los incidentes de seguridad en un 100%.
2. 2. Conocer la ubicación en tiempo real de los guardias de seguridad.

## Objetivos.

|  |  |  |
| --- | --- | --- |
| **ID** | **Objetivo** | **Prioridad** |
| 1 | Desarrollar una web responsiva que pueda leer códigos QR para verificar el cumplimiento de los rondines. | Alta |
| 2 | Desarrollar una web responsiva para dispositivos. | Alta |
| 3 | El jefe de los guardias puede llevar un control de los datos de los rondines. | Baja |
| 4 | Deberá llevar una bitácora de incidentes de seguridad | Media |
| 5 | El sistema deberá llevar un registro de los escaneos. | Media |
| 6 | El sistema deberá administrar usuarios y contraseñas | Alta |

# Alcance del Proyecto.

## Desarrollar una web responsiva.

Desarrollar una web responsiva que tenga la capacidad de cumplir como una herramienta que facilite el trabajo de los guardias de seguridad de la UADEC Campus Arteaga.

## Debe ser una web responsiva para todos los dispositivos.

La web responsiva debe funcionar en los diferentes dispositivos de cada uno de los guardias de seguridad, así como en los de los encargados que necesiten ver la información de la web.

## El encargado puede llevar un control de los datos.

El encargado debe de poder ver la información requerida de los rondines de los guardias de seguridad en el campus dentro de la base de datos, en cuanto él la necesite.

## Los datos de incidentes deben ser almacenados.

La web contara con una bitácora que muestre todos los incidentes de seguridad.

## El sistema debe registrar los escaneos

En el sistema deberá almacenar todos los registros QR generados por los recorridos de los guardias.

## Administración de usuarios y contraseñas

El sistema deberá ser capaz de tener un control de registro de usuarios y contraseñas.

# Calendario de Actividades.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Objetivo** | **SEPTIEMBRE** | | | **OCTUBRE** | | | | **NOVIEMBRE** | | | |  | | | | |
| Desarrollar una web responsiva. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Debe ser una web responsiva para todos los dispositivos. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| El encargado puede llevar un control de los datos. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Los datos de incidentes deben ser almacenados. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| El sistema debe registrar los escaneos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Administración de usuarios y contraseñas |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Plataforma de Desarrollo.

## Ciclo de Vida.

El ciclo de vida que aplica para este proyecto es el modelo en cascada o clásico

## Metodología.

La metodología que se utilizará para desarrollar los componentes de software está basada en el modelo MAPRODESv1.0 (Modelo Académico de Procesos para el Desarrollo de Software).

El modelo en cascada o clásico se basa en un enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software esto sugiere una aproximación sistemática secuencial hacia el proceso de desarrollo del software, que se inicia con la especificación de requerimientos del cliente y continua con la planeación, el modelado, la construcción y el despliegue para culminar en el soporte del software terminado.

## Software.

Para el desarrollo de la aplicación web se utilizará software tal como visual studio code donde se realizaran todos los scripts de html, css, javascrript con respectivas herramientas como bootstrap que nos permitan diseñar el mejor sitio responsivo, esto para la parte del front-end también utilizaremos el DBMS de MySQL y PHP para el back-end.

# Plan de comunicación.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo de Comunicación** | **Método / Herramienta** | **Frecuencia** | **Información** | **Participantes / Responsables** |
| **Comunicación Interna:** | | | | |
| Minutas de Revisión de Avance | MS Word,  Teams | Semanal | Estado actual del proyecto referente a las actividades | Equipo de trabajo |
| Notificaciones, acuerdos y planes | MS Word,  Teams | Diario | Comunicación actual referente a todo el proyecto | Equipo de trabajo |
| Documentación Técnica | MS Word,  Repositorio Central de Documentos | Cuando se requiera | Toda la documentación técnica de soporte | Equipo de trabajo |
| **Comunicación Externa y Reportes:** | | | | |
| Reporte de Progreso | MS Word,  Correo Electrónico | Mensual | Estado actual del proyecto con desviaciones de tiempo | Equipo de trabajo |

# Licenciamiento

El proyecto incluye el licenciamiento que ampara la instalación, uso y modificaciones por parte de la Universidad Autónoma de Coahuila, sin límite de usuarios y computadoras enlazadas al sistema.

La adquisición y uso de las licencias de terceros, tales como el Sistema Operativo, MySQL, son responsabilidad de la Universidad Autónoma de Coahuila.

# Póliza de Soporte Anual.

Esta propuesta no incluye ninguna póliza de soporte anual. Cualquier solicitud posterior a la aceptación por parte de la Universidad Autónoma de Coahuila será cotizada por la Universidad Autónoma de Coahuila y en su caso previamente autorizada por la misma Universidad.

# Condiciones Generales del Proyecto.

Se otorgará al cliente una aplicación funcional en el tiempo acordado, podrá realizar pruebas de funcionamiento y de ser necesario se podría modificar la aplicación en caso de presentar errores u omisión de algún requerimiento si el tiempo nos lo permite.

En la implementación se instalará el sistema y se capacitará a los usuarios indicados por el cliente, para su correcto uso.

# Garantía.

El producto final está respaldado por un periodo de garantía de hasta 1 año contra errores. Un error se define como una desviación en el comportamiento del sistema, que difiere de las funciones contenidas en el documento de requerimientos firmado por el cliente. En caso de que exista un supuesto error reportado por el cliente, y éste sea causa de un mal manejo del sistema, el costo de la corrección será imputable al cliente, previa autorización del mismo.

# Entregables.

Al final del proyecto la siguiente lista de productos será entregada:

* Software ejecutable del sistema actualizado.
* Código del Sistema
* Documentación electrónica actualizada para la operación del Sistema.
* Guías de Instalación y Configuración del Sistema

# Inversión.

## Costos.

El costo total de los servicios ofertados es de:

|  |  |
| --- | --- |
| **Concepto** | **Cantidad** |
| Desarrollo del Proyecto | $ 0.00 |

Las cantidades no incluyen el Impuesto al Valor Agregado

## Flujo de Pagos.

La siguiente tabla muestra la forma propuesta para el pago del proyecto:

|  |  |  |
| --- | --- | --- |
| **Etapa** | **Cantidad** | **Fecha** |
| Anticipo (50%) | $0.00 | Al firmar el contrato |
| A la entrega del proyecto (50%) | $ 0.00 | Al finalizar los trabajos |
| **TOTAL** | **$ 0.00** |  |

# Glosario.

SW Denominación que reciben los programas informáticos.

Análisis Proceso de estudiar las necesidades del usuario para obtener una

definición detallada para los productos.

Ciclo de vida Describe el desarrollo de software, desde la fase inicial hasta la fase final. El propósito de este programa es definir las distintas fases intermedias que se requieren para validar el desarrollo de la

aplicación, es decir, para garantizar que el software cumpla los

requisitos para la aplicación y verificación de los procedimientos de

desarrollo: se asegura de que los métodos utilizados son los apropiados.

Ingeniería de Es la aplicación de un enfoque sistemático, disciplinado y

SW cuantificable al desarrollo, operación y mantenimiento de software y el estudio de estos enfoques, es decir, la aplicación de la ingeniería al software.

API Una API es una interfaz de programación de aplicaciones (application programming interface), un conjunto de funciones o métodos usados para acceder a cierta funcionalidad.

Aplicación

cliente

Una aplicación (cliente) es cualquier programa de computadora que usa las interfaces de programación del Modelo de Objetos del Documento proporcionadas por la implementación anfitrión para realizar trabajo útil. Ejemplos de aplicaciones cliente son los scripts de un documento HTML o XML.

Software de

dominio

publico

Programa donado para uso público por su propietario o diseñador y que se podrá utilizar, copiar y distribuir con total libertad.