**DEFINICIÓN DE PROYECTO DE SOFTWARE**

**DOCUMENTO RDP PARA ESMEL-PARQ**

**DOCUMENTO CREADO POR:**

**MELQUI CAMACHO ESPITIA – 814014 - MCAMACHOE@UNAL.EDU.CO**

**ESTEFANIA IMBACHI ESCOBAR – 212529 - EIMBACHIE@UNAL.EDU.CO**

**ASIGNATURA:**

**TÉCNICAS DE PROGRAMACIÓN**

**DOCENTE:**

**JUAN SEBASTIÁN LÓPEZ VILLA**

**UNIVERSIDAD NACIONAL DE COLOMBIA**

**SEDE MANIZALES**

**NOVIEMBRE 05 DE 2015**

**1) Descripción del software:**

1. *ESMEL-PARQ*. Llamado así en primera parte por los nombres de sus diseñadores, quienes decidimos dejar parte de nuestros nombres impregnados en el software. La segunda parte brinda un rápido sentido de su funcionalidad, dando así un buen entendimiento comercial.
2. ESMEL-PARQ está diseñado con el fin de ayudar en las labores de contabilidad de una empresa de parqueaderos, facilitar el registro de los automotores a guardarse en estos, realizar la factura de pago de cada automotor y como valor agregado es capaz de entregar reporte sobre la productividad de los mismos.
3. Al entregar reporte sobre la productividad de los parqueaderos el software está ayudando a cumplir las labores de recolección de datos por parte de los encargados de finanzas en la empresa, lo cual brinda una mayor eficacia y eficiencia en dichas labores pues reduce el tiempo estimado para realizar los cálculos y estos son efectuados por el software dando así la confiabilidad de que los datos y resultados son exactos y sin manipulación alguna.

El generar la factura de pago se cumple con las respectivas leyes del comercio pues el software integra en su configuración las normas establecidas para la adecuada imprenta de las facturas, dando así la fiabilidad de contar con un sistema que libra de repercusiones legales en el área especificada.

1. Los beneficiados con este software van a ser desde el portero del parqueadero hasta el dueño de la empresa debido a que cubre diferentes necesidades en varias labores de los trabajadores de la empresa.  
     
   Al brindar el fácil registro y generar la factura de pago el portero deberá cumplir la labor de ingresar los datos necesarios, al llevar la contabilidad de los parqueaderos los encargados de la contabilidad van a obtener los datos fácilmente para las respectivas evaluaciones y al brindar el reporte de los parqueaderos el dueño o gerente de los mismos podrá evaluar la productividad de cada uno.

**2) Funcionalidades del software:**

ESMEL-PARQ a partir de su codificación realiza las siguientes funciones a especificar:

* **Funcionalidad 1**: Registrar nuevo parqueadero.

Al estar diseñado el software para una empresa con varios parqueaderos esta funcionalidad permite el registro de nuevos parqueaderos sin un límite alguno.

* **Funcionalidad 2**: CRUD parqueadero.

Con esta funcionalidad es posible la búsqueda de cada parqueadero, la muestra de sus datos, la modificación y eliminación de los mismos.

* **Funcionalidad 3**: Registrar automóvil.

Permite el registro de cada automóvil al ser ingresado al parqueadero, de esta manera se monitores el tiempo de servicio y datos del dueño.

* **Funcionalidad 4**: CRUD automóvil.

Como en el caso de los parqueaderos, es posible la búsqueda de cada automóvil, la muestra de sus datos, la modificación y eliminación de los mismos.

* **Funcionalidad 5**: Contabilizar tiempo de servicio.

Al ser ingresado un automóvil se proporciona una hora de llegada y al retirarse el mismo una hora de salida, con estos datos se contabiliza el tiempo de servicio del automóvil.

* **Funcionalidad 6**: Factura automóvil.

Con los datos obtenidos en el ingreso y la contabilidad del servicio e integrándolos con el código de facturación quemado en el software se proporciona la factura correspondiente a cada automóvil según el tiempo y tarifa establecidos.

* **Funcionalidad 7**: Registro parqueaderos.

Cada facturación que se realice es registrada en un archivo de extensión “.dat” para que no pueda ser leído ni modificado fuera del programa.

* **Funcionalidad 8**: Reportar ganancias.

Con esta funcionalidad es posible obtener el total de las ganancias de un parqueadero y el número de carros y motos a los que se les ha proporcionado el servicio, esto en un rango de tiempo proporcionado en días.

* **Funcionalidad 9**: Seguridad de los datos.

Para poder ingresar o modificar la información de los parqueaderos es necesario proporcionar una clave de acceso, la cuál será puesta a disposición del dueño de la empresa.

**3) Mockup del software – Diseño de interfaz gráfica de usuario: mín 4 páginas, máx 8 páginas**

Debe incluir boceto de interfaz gráfica de usuario a implementar, agregando pantallazos de diseño en ***QtDesigner*** y describiendo las diferentes pantallas y/o widgets, su respectivo uso y contenido, relacionados directamente con las funcionalidades que tiene el software.

En el ejemplo del sistema básico de control de acceso, se podrían definir las siguientes pantallas y su relación con las diferentes funcionalidades:

* Pantalla de autenticación: permite registrar un ingreso o una salida. Adicionalmente muestra si la persona está autorizada para ingresar a la zona. Muestra los diferentes accesos de todos los trabajadores en las diferentes zonas en una pestaña. Está relacionada con las funcionalidades 1 y 2.
* Pantalla de CRUD de personas: permite crear, listar, eliminar o actualizar la información de las personas que trabajan en la empresa. Está relacionada con la funcionalidad 3.
* Pantalla de reportes: permite generar y visualizar los diferentes tipos de reportes de control de acceso y registro de trabajo de personal. Está relacionada con las funcionalidades 4 y 5.
* Pantalla de asignación de permisos: permite definir qué personas están autorizadas en cada zona de la empresa. Está relacionada con la funcionalidad 6.
* Pantalla CRUD de zonas de empresa: permite crear, listar, eliminar o actualizar la información de las zonas de trabajo de la empresa. Está relacionada con la funcionalidad 7.
* Pantalla guardar reporte: permite seleccionar la información a guardar en el reporte .txt ya se de registro de trabajo o de intentos de acceso no autorizados. Está relacionada con la funcionalidad 8.

Se debería adjuntar una imagen por cada pantalla que tenga el software, como pantallazo del archivo de diseño .ui del QtDesigner. Utilizar combinación de teclas ***Alt+ImprPant*** para imprimir sólo la ventana visible después de utilizar combinación de teclas ***Ctrl+R*** en QtDesigner.

**4) Listado de entidades del modelo del software – mín 1 página, máx 2**

Debe contener un listado con las entidades (clases) que harán parte del modelo del software, incluyendo su descripción y justificando su necesidad de uso.

En el proyecto de ejemplo del sistema básico de control de acceso, se podrían definir las entidades:

* **Persona**: Entidad que representa a los miembros de la planta de producción y tiene los diferentes roles:
  + **Administrador**: Tiene acceso a todas las funcionalidades del software.
  + **CoordinadorProduccion**: Coordinador de la producción de la planta.
  + **CoordinadorNomina**: Coordinador de nómina de la planta.
  + **Empleado**: Trabajador de la planta.
* **Zona**: Zona de producción en la planta.
* **RegistroAcceso**: Almacena la información para un ingreso o una salida de una zona de trabajo.
* **PermisoZona**: Relaciona cada Persona con cada Planta y define el permiso que tiene, el cual puede ser: PERMITIDO, NO\_PERMITIDO, JEFE. Una Persona sólo puede tener un permiso a la vez.

**5) Cronograma, actividades y costos de ejecución de proyecto de software – mín 2 páginas, máx 4 páginas**

Listado de actividades necesarias para implementar el proyecto de software, asignadas para cada integrante del equipo o de manera individual, incluyendo el tiempo estimado para cada actividad y su responsable. Al final del proyecto deben estimar el costo de la implementación del proyecto. Cada tipo de actividad o cada etapa deben tener un valor por hora asignado y al final definir cuál es el costo total del proyecto. Las fechas del cronograma deben ser reales.

En esta sección se deben incluir 3 tablas:

1. ***Tabla de actividades*** para las fases 2 y 3 del proyecto con la estimación en tiempo de cada actividad y lo que planea hacer cada miembro del equipo.
2. ***Tabla de cronograma*** para las fases 2 y 3 del proyecto con la definición de las fechas en las cuales realizarán las actividades del proyecto cada miembro del equipo.
3. ***Tabla de costos*** para todas las fases del proyecto, indicando cuánto se le debe pagar a cada miembro del equipo por su participación en el proyecto de software, discriminando cuánto gana por cada tipo de actividad. Esta será la estimación inicial, el presupuesto inicial, al final se convalidará con los registros de trabajo en las bitácoras de cada miembro del equipo.

Para cada etapa se proponen los siguientes costos de referencia:

|  |  |
| --- | --- |
| **Nombre Actividad** | **Costo de Referencia por Hora (Pesos)** |
| Creación de documentos de requisitos, diseño y planeación de proyecto de software. | $25,000 |
| Refinamiento de documento de requisitos del proyecto de software. Creación de repositorio de código en github.com. | $25,000 |
| Diseño de modelo de clases y/o datos, diseño de interfaz gráfica de usuario. | $32,000 |
| Implementación de modelo de clases y/o datos, implementación de interfaz gráfica de usuario. | $23,000 |
| Implementación de funcionalidades -> 6 funcionalidades Parejas. 3 funcionalidades Individual. | $23,000 |
| Realización de pruebas de correcto funcionamiento de los casos de uso. Una prueba por cada caso de uso. | $14,000 |
| Refinamiento de implementación. | $23,000 |
| Documentación para usuario final de proyecto de software. | $12,000 |
| Empaquetamiento de software. | $15,000 |

Al final se debe informar cuánto vale el desarrollo del proyecto de software y cuánto se le debe pagar a cada miembro del equipo.

Realizar la división y asignación de los tiempos para la ejecución del proyecto y agendarlos en el cronograma. Cada vez que se realice un avance de trabajo, deberá ser registrado en la bitácora de trabajo del proyecto, la cual debe ser individual. Al final se comparará el tiempo estimado con el tiempo empleado por cada miembro del equipo.

Basarse en los ejemplos suministrados en el archivo de Excel.

**6) Diseño de pruebas de proyecto de software – mín 1 página, máx 2 páginas**

Definir qué pruebas se realizarán para verificar el cumplimiento de los requerimientos y funcionalidades especificadas. Para el proyecto de ejemplo se tienen programas las siguientes pruebas:

* **Prueba 1:** Ingreso y Salida de un trabajador. El software debe calcular de manera automática el tiempo laborado. Si no se da ingreso, debe generar inasistencia y advertencia. Si no se da salida, debe generar advertencia.
* **Prueba 2:** Registro exitoso de zona de producción en la planta. Asignación de zona a cada empleado.
* **Prueba 3**: Creación exitosa de empleado, asignación de permisos.
* **Prueba 4**: Creación exitosa de coordinadores. Visualización de la interfaz de usuario basado en el rol de cada usuario.
* **Prueba 5**: Verificación que se genera una advertencia y registro de acceso no autorizado.
* **Prueba 6**: Verificación de reportes generados tanto de horas laboradas como de accesos no autorizados.

**7) Tabla de contenido de la documentación de usuario – mín 1 página, máx 1**

Incluir la tabla de contenido propuesta que tendrá el documento de usuario del proyecto de software. La tabla de contenido en la documentación del proyecto de ejemplo sería:

1. Descripción de ControlMais.
2. Funcionalidades de ControlMais.
3. Cómo crear usuarios y asignarles roles con ControlMais? Guía para el administrador.
4. Cómo registrar sesión de trabajo con ControlMais? Guía para el trabajador.
5. Cómo determinar los tiempos de trabajo de un empleado con ControlMais? Guía para el coordinador de nómina.
6. Cómo definir las zonas de producción de la planta y con ControlMais? Guía para el coordinador de producción.
7. Cómo generar y visualizar los reportes generados por ControlMais? Guía para los coordinadores de producción y de nómina.
8. Mecanismos de soporte del usuario de ControlMais.