|  |  |
| --- | --- |
| **Integrantes** | |
| Esteban Dalel R | 201227078 |
| <nombre y apellidos> | <código> |

# Estructura máxima 5 HOJAS

# Introducción

Realice una breve explicación de que hicieron y cómo llegaron a la solución propuesta.

# Derechos de propiedad

Nosotros Esteban Dalel R y <nombre2> garantizamos que el presente trabajo y todos sus entregables son de nuestra propiedad o usados con los correspondientes permisos, sin incurrir en ningún tipo de falta de derechos de propiedad.

# Requerimientos Funcionales

Identifique los 3 requerimientos funcionales principales para el sistema propuesto y documéntelos siguiendo el siguiente formato:

|  |  |
| --- | --- |
| Nombre | Recibir PQRs |
| Resumen | Al entrar una llamada de un cliente, se le pregunta si es una Petición, Queja o Reclamo, y, si no hay uno abierto, se abre un caso para resolver su inquietud. |
| Entradas | |
| Tipo: P|Q|R  Descripción del caso  Datos del cliente con el requerimiento  Fecha de entrada (DDMMAA HH:MM:SS) | |
| Resultados | |
| Se abre un caso, se pone en base de datos.  Si el cliente no existe, se crea. | |

# Modelo del mundo

Presente de forma clara su modelo del mundo usando la notación UML y explique los diferentes elementos del que consiste (para cada parte del proyecto). Indique como **las estructuras que usted creó,** las cuales deben expresarse en el modelo, satisfacen los requerimientos funcionales propuestos por el cliente al igual como los no funcionales.

Así mismo, indique cual fue el aporte de cada estudiante en el proyecto (quién hizo qué).

# Análisis de complejidad

De los requerimientos propuestos tome los 3 más importantes, copie el fragmento de código que satisface dicho requerimiento y haga un análisis de complejidad para cada uno. Indique el porqué de su respuesta, si piensa que no es la solución óptima explique porqué tomó dicha decisión y cuál podría ser una mejor solución (si es la óptima indique porqué lo es).

Public boolean recibirLlamada(String datosCliente, String PQR, Date fechaProblema, String razonLlamada)

{

//guardar datos

for(i=0;i<clientes.size;i++)

{

If(datosCliente==clients[i])

{

Caso ultimaLlamada=new Caso(clients, PQR, fechaProblema);

Return true;

}

}

}

Cliente esteCliente= new Cliente (datosCliente);

Caso ultimaLlamada=new Caso(esteCliente, PQR, fechaProblema);

}