UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

Tesis previa a la obtención del Título de Ingeniero de Sistemas

TÍTULO:

"SERVICIO DE MENSAJERÍA INSTANTÁNEA PARA LA COMUNIDAD DEL INSTITUTO TÉCNICO SALESIANO."

AUTORES:

Jorge Luis Calderón Villavicencio
Rolando Efraín Vicuña Méndez

DIRECTOR:

Ing. Diego Quinde Falconí

Cuenca, Agosto del 2012

DECLARACIÓN DE RESPONSABILIDAD

Los conceptos desarrollados en este trabajo, así como todo el estudio e implementación de este Proyecto, son de exclusiva responsabilidad de los Autores y autorizamos a la UNIVERSIDAD POLITECNICA SALESIANA hacer uso de éste para fines académicos.

Los textos de otros autores llevan su correspondiente cita bibliográfica.

Cuenca, 2 de agosto del 2012

Jorge Luis Calderón Villavicencio. Rolando Efraín Vicuña Méndez.

CERTIFICADO

Yo, Diego Marcelo Quinde Falconí, Docente de Carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana, CERTIFICO que he leído íntegramente el presente trabajo de grado, comprobando que se cumplen los objetivos planteados en el proyecto. Así mismo declaro que los autores Jorge Calderón y Rolando Vicuña son los únicos responsables de la originalidad del trabajo desarrollado.

(f)_____
Ing. Diego Quinde Falconí

Dedicatoria de Jorge Luis Calderón Villavicencio

Esta tesis va dedicada a mis padres Jorge Gonzalo y Diana Rebeca porque este título no es solo mío sino de ustedes también por apoyarme incondicionalmente desde siempre en todo realizando grandes esfuerzos, mami, papi ya soy profesional para ustedes este título.

Dedicatoria de Rolando Efraín Vicuña Méndez

La vida es tan corta y misteriosa que cada vez nos llena de sorpresas, hoy estoy culminando una etapa de mi vida la cual me servirá de entrada hacia otra. Dedico este proyecto a la persona más importante en mi vida que es mi madre quien fue la principal promotora de que hoy en día sea lo soy y lo que he sido, dándome todo el apoyo y esfuerzo para lograr culminar con éxito mis metas y objetivos planteados, gracias por todo.

Agradecimientos de Jorge Luis Calderón Villavicencio

Agradezco a Dios por haberme dado fuerzas para terminar esta meta que me propuse, agradezco a mis padres, hermanos a todos mis tíos y familiares porque todos colaboraron de una u otra manera, a todos mis amigos y amigas, a todos quienes alguna vez fueron mis compañeros de aula, a todos los profesores que compartieron su conocimiento, a mi director de tesis, a mi compañero de tesis, al director del departamento de sistemas de la Unidad Educativa Técnico Salesiano y todos quienes se involucraron y dieron ánimos a que salga a flote este proyecto, gracias a todos.

Agradecimiento de Rolando Efraín Vicuña Méndez

Cualquiera podrá extendernos la mano pero no todos con la misma intención, es por ello que agradezco principalmente a DIOS y a todas las personas quienes supieron confiar en mí y de manera muy particular a quienes no lo hicieron, en realidad existen muchas personas que verdaderamente merecen mis agradecimientos entre ellas mi familia en especial mi hermano y mi tío Miguel, quienes han sido las personas que han sabido inculcarme por el camino del conocimiento a lo largo de mi vida, aportándome con sus experiencias y brindándome luz en mi existencia.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN 2
1.1. ANTECEDENTES 2
1.2. OBJETIVOS 3
1.2.1. Objetivo General 3
1.2.2. Objetivos Específicos 3
1.3. ALCANCE 4
1.4. FUNCIONAMIENTO DE MENSAJES DE TEXTO EN CELULARES 5
1.5. TIPOS DE MENSAJES EN CELULARES. 6
1.5.1. SMS 6
1.5.2. EMS 7
1.5.3. MMS 8
2. ANÁLISIS 10
2.1. ANÁLISIS DE LOS SISTEMAS ACTUALES EN LA UNIDAD EDUCATIVA TÉCNICO SALESIANO 10
2.2. ANÁLISIS DE LAS SOLUCIONES DE GATEWAY EXISTENTES 11
2.3. INVESTIGACIÓN DE CELULARES Y CARACTERÍSTICAS 12
2.4. RECOPILACIÓN DE REQUISITOS. 17
2.5 DOCUMENTACION DE REQUISITOS 19
2.6 DIAGRAMAS DE LA APLICACIÓN. 21
2.6.1 DIAGRAMAS DE CASO DE USO. 21
2.6.2 DIAGRAMAS DE SECUENCIA. 22
2.6.3 DIAGRAMA DE COMPONENTES. 23
2.7 DEFINICIÓN DEL ALCANCE. 23
3 DISEÑO 25
3.5 ARQUITECTURA DE LA APLICACIÓN. 25
3.6 REQUERIMIENTOS DE SOFTWARE. 27
3.6.1 REQUISITOS FUNCIONALES. 27
3.6.2 REQUISITOS NO FUNCIONALES 27
3.6.3 REQUISITOS DE RENDIMIENTO 27
3.6.4 SEGURIDAD 28
3.6.5 DISPONIBILIDAD Y FIABILIDAD 29

3.6.6 MANTENIBILIDAD....29
3.6.7 PORTABILIDAD....30
3.6.8 OTROS REQUISITOS....30
3.7 REQUERIMIENTOS DE HARDWARE....31
3.7.1 COMPUTADOR QUE ALOJARA LA APLICACION....31
3.7.2 MODEM GSM....32
3.7.3 JUSTIFICACIÓN HERRAMIENTA A UTILIZAR....33
3.8 MODELO DE DATOS....34
3.8.1 DIAGRAMA DE CLASES....34
3.8.2 DIAGRAMA ENTIDAD RELACIÓN....35
3.9 DESARROLLO....36
4 IMPLEMENTACIÓN....43
4.1 INSTALACIÓN....43
4.1.1 INSTALACIÓN DEL APLICATIVO SMS ENVÍO MASIVO....43
4.1.2 INSTALACIÓN DEL APLICATIVO RECEPCIÓN....44
4.2 CONFIGURACIÓN....44
4.3 PRUEBAS....45
4.3.1 MODULO DE ENVÍO....45
4.3.2 MODULO DE RECEPCIÓN....47
4.4 ANALISIS DE RESULTADOS....48
4.5 MANUALES DE USUARIO....49
4.5.1 MANUAL DE USUARIO DEL MÓDULO DE ENVÍO....49
5.5.2 MANUAL DE USUARIO DEL MÓDULO DE RECEPCIÓN....51
CONCLUSIONES....53
RECOMENDACIONES....54
BIBLIOGRAFÍA....55
ANEXOS....56
ANEXO 1 INSTALACIÓN DE JAVA....57
ANEXO 2 CONFIGURAR LA VARIABLE DE ENTORNO JAVA_HOME....60
ANEXO 3 CONFIGURAR LA VARIABLE DE ENTORNO PATH....61
ANEXO 4 INSTALACIÓN DE APACHE TOMCAT 7....62
ANEXO 5 ENCUESTA RALIZADA A LOS ESTUDIANTES DE LA UNIDAD EDUCATIVA TÉCNICO SALESIANO....65
ANEXO 6 ERRORES DE COMANDOS AT....71

ANEXO 7 FICHA TÉCNICA DEL MODEM HUAWEI-E173 76

ÍNDICE DE FIGURAS

Gráfico 2.1: Solicitud de envío de calificaciones 21
Gráfico 2.2: Envío de SMS informativos o recordatorios. 21
Gráfico 2.3: Solicitud de información. 22
Gráfico 2.4: Envío de SMS informativos o recordatorios. 22
Gráfico 2.5: Diagrama de componentes del sistema. 23
Gráfico 3.1: Arquitectura de la aplicación: Modulo de envío de información. 26
Gráfico 3.2: Arquitectura de la aplicación: Modulo de recepción y envío de respuesta 26
Gráfico 3.3: Diagrama de clases del módulo de recepción /envío 34
Gráfico 3.4: Diagrama de clases de módulo de envío masivo de SMS. 35
Gráfico 3.5: Tablas de la base de datos del usuario SMS 36

ÍNDICE DE TABLAS

Tabla 2.1: Resumen de características con el siguiente cuadro comparativo de Módems 16
Tabla 4.1: Primera prueba del módulo de envío masivo. 46
Tabla 4.2: Segunda prueba del módulo de envío masivo. 46
Tabla 4.3: Tercera prueba del módulo de envío masivo. 47
Tabla 4.4: Primera prueba del módulo de recepción/envío. 47
Tabla 4.5: Segunda prueba del módulo de recepción/envío. 48
Tabla 4.6: Tercera prueba del módulo de recepción/envío. 48
Tabla 4.7: Cuarta prueba del módulo de recepción/envío. 48

CAPITULO 1

INTRODUCCIÓN

1. INTRODUCCIÓN

1.1. ANTECEDENTES

En la actualidad existen numerosas tecnologías que facilitan la comunicación entre los diferentes usuarios, una de las características que ofrecen las diferentes compañías que utilizan la red celular es el envío y recepción de mensajes, sean estos de texto o multimedia siendo el más utilizado los (1)SMS que son mensajes de tamaño corto (160 caracteres). Si bien los SMS fueron creados como parte de un estándar de telefonía móvil, en un principio fueron utilizados exclusivamente por los operadores de red celular para dar a conocer cierta información a todos sus clientes.

Hoy en día la red celular y de manera muy particular el envío y recepción de mensajes entre los clientes ha sido de mucha utilidad en los últimos años, pudiendo hacer uso de ésta para muchas aplicaciones como es el caso de algunos bancos, cooperativas que utilizan la red celular para dar a conocer a sus clientes sus estados de cuenta y otro tipo de información dependiendo de cada entidad.

Las universidades, escuelas, colegios y otras instituciones constantemente necesitan dar a conocer a los padres de familia o personal interno cierta información ya sea para convocación a reuniones o simplemente dar cualquier tipo de aviso. Existen diferentes formas que nos facilitan el desarrollo de este tipo de proyectos, entre ellos la utilización de pasarelas que no son más que intermediarios para el envío y recepción de los mensajes, otra forma de realizar esto es manteniendo todo el control de la aplicación es decir no tener la necesidad de depender de otro tipo de entidad. Este tipo de proyectos se lo puede realizar con diferentes lenguajes de programación entre ellos Pyton, Visual Basic, Java, etc. Tomando como referencia el lenguaje de programación Java, este cuenta con muchas librerías que facilitan de una forma muy importante el desarrollo del mismo.

Por otro lado el hardware necesario para estos proyectos puede variar, se pudiera hacer uso de dispositivos celulares conectados a un PC o de lo contrario la utilización de módems con chip integrado que de igual forma se conectaran al

PC, de cualquiera de las formas antes mencionadas ambos dispositivos utilizan (2)comandos AT[1] para la comunicación entre la PC y el dispositivo y de igual forma Java cuenta con librerías que permiten la manipulación de comandos AT.

1.2. OBJETIVOS

1.2.1. Objetivo General

Desarrollar e implementar el servicio de mensajería instantánea para la comunidad del Instituto Técnico Salesiano.

1.2.2. Objetivos Específicos

Enlazar la página del Instituto Técnico Salesiano con la aplicación que se va a desarrollar.

Desarrollar una aplicación web que corra en la página web del Instituto Técnico Salesiano.

Añadir Entidades a la base de datos del colegio para almacenar los números telefónicos de los diferentes grupos.

Interconectar la aplicación con la base de datos del colegio.

Investigar y recomendar la adquisición de un dispositivo de comunicación que mejor se acople a las necesidades del instituto.

Utilizar tecnología actualizada JAVA AJAX con base HTML5 para el desarrollo.

---

[1] AT: El conjunto de comandos Hayes es un lenguaje desarrollado por la compañía Hayes Communications que prácticamente se convirtió en estándar abierto de comandos para configurar y parametrizar módems.

[2] PAYLOAD: Es la carga de transmisión de datos, es información que se envía junto con la información

1.3. ALCANCE

La comunidad actualmente cuenta con diferentes grupos de personas a las cuales se emiten distintos comunicados, entre estos grupos se encuentran:

- Alumnos
- Padres de familia
- Docentes
- Comunidad Salesiana

Hoy en día es muy común que las personas utilicen celular, y generalmente el uso de mensajes de texto, este proyecto se centra en el desarrollo de un módulo que previo al almacenamiento, nos permita extraer números telefónicos de celulares de las distintas personas agrupadas en la descripción anterior.

El proyecto está enfocado para el envío masivo de mensajes texto a celulares, logrando llegar individualmente a cada una de las personas con un mismo mensaje de interés al grupo al que pertenezca, Cubriendo las necesidades que actualmente el colegio presenta de forma rápida, oportuna y eficaz.

Existe también la posibilidad de que un individuo perteneciente a algún grupo de la comunidad salesiana ya sean estudiantes, profesores, directivos entre otros realicen consultas al sistema informático del colegio y comunidad, entonces el sistema de mensajería instantánea debe responder al número telefónico que solicito la petición, luego del debido proceso según corresponda la consulta solicitada. Por poner un ejemplo quizá una de las consultas que más utilidad tenga luego de la implementación es que un estudiante poseedor de un celular con número 084226723 envíe un SMS con la palabra notas al número telefónico del colegio, entonces el sistema de mensajería instantánea debe detectar la llegada de una petición, realizar la consulta de las notas del estudiante al que pertenece ese número de celular el cual ya está previamente guardado en el sistema y enviar una respuesta al mismo número de celular 084226723 con las notas obtenidas hasta el momento de las materias que este cursando.

1.4. FUNCIONAMIENTO DE MENSAJES DE TEXTO EN CELULARES

Hoy en día el papel que juegan las compañías operadores de la red celular es muy importante y a la vez imprescindible ya que debido a los avances tecnológicos la comunicación en esta época es cada vez más eficiente y rápida. Una de las características que nos brindan estas compañías es el servicio de mensajería, entendiéndose como el envío y recepción de mensajes cortos (SMS). (3)Al hablar de SMS nos referimos al tamaño máximo que pude tener el mensaje este caso 160 caracteres y es texto simple, el mismo que podrá ser transmitido con cualquiera de las operadores que estén a cargo en cada país.

Para realizar el proceso de envío y recepción de los mensajes es necesario de dos operaciones diferentes, la primera es el envío del SMS al SMSC (Short Message Service Center), este último se encarga de manipular, encolar y redireccionar el mensaje al destinatario. No se garantiza el 100% de la entrega del mensaje, se podrían dar algunos inconvenientes y en el peor de los casos la pérdida total del mensaje. Por otro lado el encolado, el orden, redireccionado y por ende la velocidad de entrega del mensaje depende exclusivamente del SMSC.

La primera operación, es decir el envío del mensaje al SMSC se denomina SMS-MO(SMS Mobile Originated).

La segunda operación, es decir el redireccionado del mensaje al destinatario se denomina SMS-MT (SMS Mobile Terminated).

Los SMS nacen como parte del estándar de la tecnología GSM, pero en la actualidad las tecnologías como 3G hacen uso de dicha característica.

Parámetros:

En el proceso de envío o recepción de mensajes se involucra algunas características o parámetros en su PayLoad[2] que es el cuerpo del mensaje, entre ellos tenemos:

- Fecha de envío
- Validez del mensaje, desde una hora hasta una semana
- Número de teléfono del remitente y del destinatario
- Número del SMSC que ha originado el mensaje

1.5. TIPOS DE MENSAJES EN CELULARES.

1.5.1. SMS

Los SMS que significa Short Message Service, lograron su masificación de su uso desde hace pocos años de la mano de los teléfonos celulares, pese a que existen desde 1985 cuando un grupo de técnicos franceses y alemanes desarrollaron y establecieron un estándar para el servicio de envío de mensajes cortos que hoy utiliza la sociedad en todo el mundo. Y es una forma rápida y la más usada de hacer saber algún comunicado a otro usuario de un teléfono celular.

Este servicio de SMS que en principio se creó para la comunicación entre usuarios y telefónicas se hizo popular por la rapidez y facilidad de uso, entonces fue cuando las operadoras decidieron abrir a los usuarios este sistema de comunicación gratuitamente como alternativa de comunicación a la llamada celular. En consecuencia de tener SMS gratuito y la informalidad de redactar texto para comunicarse con amigos, familia o usuarios de quien conozca el número de teléfono celular, este sistema de SMS salto de ser un mecanismo alternativo a la principal forma de comunicación en todo el mundo.

[2] PAYLOAD: Es la carga de transmisión de datos, es información que se envía junto con la información de interés para el destinatario entiéndase como datos generales o metadatos.

A partir del año 2000 fue la mejor oportunidad para las operadoras telefónicas abriéndose una nueva oportunidad de negocio a explotar, entonces mejoraron el servicio y creando planes de mensajes y promociones, los mensajes ahora tenían un costo por el envío. Esto no hizo que se pierda la popularidad de los SMS.

En el 2003 se crean los sistemas de números cortos para el mejor manejo de marketing por ejemplo: "ENVIA LA PALABRA CHISTE AL 2020". También se agregaron servicios como los que permiten descargar ring tones y demás contenido multimedia.

Para el año 2009 los SMS generaron la mayor ganancia de todos los ingresos de las operadoras en el mundo que ronda los $ 150 mil millones de dólares y promete seguir creciendo a fuerza de nuevas utilidades.

La importancia del mensaje de texto en la telefonía móvil llegó al punto de superar incluso al uso del sistema de voz, que ahora solo se utiliza como un sistema de comunicación más formal.

Ahora bien, el SMS es algo más que simplemente un sistema de comunicación, también trae aparejado prácticas que claramente afectaron nuestra vida en estos 26 años desde su lanzamiento.

1.5.2. EMS

En la década de 1990, cuando se dio el avance tecnológico de la segunda generación de telefonía celular de 2G[3] a 2.5G[4] y la necesidad de mejorar el servicio de mensajería ya que los SMS solo pueden enviar texto simple o texto plano, se creó EMS (Enhanced Message Service), que en español es servicio mejorado de mensajería, presentaba nuevas ventajas respecto a SMS

[3] 2G: Esta tecnología marco el cambio de protocolos de telefonía móvil analógica a digital.
[4] 2.5G: Es una mejora a la tecnología 2G incrementando los servicios o utilidades que la tecnología 2G ofrecía.

como era el texto con formato o texto a colores, algo fácil de notar que un mensaje EMS tiene la capacidad equivalente a tres o cuatro mensajes SMS y la principal característica de EMS fue que se podía agregar iconos en el texto y sonido por ejemplo tonos para timbre de celular entre otros.

1.5.3. MMS

MMS son las siglas de Multimedia Message Service, Funciona mediante GPRS[5] (General Packet Radio Service), se desarrolló con la llegada de la segunda generación de telefonía celular para mejorar EMS; es un estándar que permite compartir contenidos multimedia entre usuarios de teléfonos celulares, esto es enviar o recibir contenido de audio, fotos, videos.

MMS también permite envío de contenidos multimedia a correo electrónico o web blog, ampliando las posibilidades de la comunicación móvil.

[5] GPRS: Sistema de transmisión de datos de alta velocidad mediante paquetes desarrollado para telefonía móvil, utiliza varios canales de comunicación a la vez, abre un nuevo canal para cada transmisión de información ya sea de voz o datos.

CAPITULO 2

ANÁLISIS

2. ANÁLISIS

2.1. ANÁLISIS DE LOS SISTEMAS ACTUALES EN LA UNIDAD EDUCATIVA TÉCNICO SALESIANO

El colegio técnico Salesiano actualmente tiene sus sistemas en 2 plataformas que son Oracle y Fox, la mayor parte de sus sistemas están en funcionamiento con la plataforma de Oracle entre ellos están:

En la parte Académica:

- Gestión Curricular (Oracle)
- Inscripciones y Matriculas (Oracle)
- Calificaciones (Oracle)
- Clubes (Oracle)
- Bitácora Estudiantil (Oracle)
- Orientación Vocacional (Oracle)
- Web para registro de notas (Oracle)

En la parte Financiera:

- Recaudaciones (Oracle)
- Activos Fijos (Oracle)
- Roles de Sueldo (Fox)
- Bancos (Fox)
- Contabilidad (Fox)
- Guarda Almacén (Fox)

A más de esto cabe mencionar que el colegio está en proceso de migración de todos sus sistemas para el entorno web con Oracle APEX.

Por cuestiones técnicas y de políticas del colegio el aplicativo que se desarrollara será totalmente independiente de todos los aplicativos con los que cuenta actualmente.

2.2. ANÁLISIS DE LAS SOLUCIONES DE GATEWAY EXISTENTES

En la actualidad existen muchas opciones de Gateway con sus respectivas ventajas y desventajas para lo cual citaremos algunos de los más populares:

- **KANNEL:**

  Este es un SMS Gateway de software libre que proporciona las características de envío y recepción de SMS con la posibilidad de trabajar con SMS multimedia, es un software bastante robusto y de gran acogida por los usuarios, este SMS Gateway esta entre los más populares del mercado, entre otras de las características tenemos que cuenta con reportes de entrega de mensajes. Por otro lado Kannel está desarrollado para sistemas operativos Linux.

- **ALAMIN:**

  Alamin es un SMS Gateway con licenciamiento GPL posee las características de envío y recepción de SMS sin posibilidad de trabajar con SMS multimedia, está desarrollado sobre una plataforma Linux.

- **PROVATO:**

  Provato es SMS Gateway propietario lo cual es un punto importante a considerar, por otro lado una gran característica que es multiplataforma es decir soporta sistemas operativos Windows y Linux, cuenta con las características de envío y recepción de SMS. Esta construido con tecnologías J2EE y por lo tanto cuenta con la administración directa desde la web. Al estar desarrollado con tecnologías J2EE posibilita la integración con lenguaje de programación Java.

Entre los Gateway mencionados anteriormente todos cumplen con los estándares de envío y recepción de SMS que plantea el proyecto con sus respectivas características que los distinguen que es el punto principal del proyecto a implementar, pero si se llegara a escoger uno de ellos para la integración con el

proyecto, el sistema dependería de aplicaciones terceras en los cuales dificultaría el control total de la aplicación y esto sería una característica muy importante a considerar, por lo cual lo más conveniente luego de conocer los respectivos Gateway existentes se ha llegado a la conclusión que se debería tener el control total de la aplicación y no depender de códigos terceros, por lo tanto se desarrollara una pasarela completamente nueva acoplándose a las necesidades del proyecto.

2.3. INVESTIGACIÓN DE CELULARES Y CARACTERÍSTICAS

Al iniciar el proyecto se planteó como idea utilizar celulares como medio o hardware para el envío y recepción de mensajes pero en el transcurso del desarrollo del proyecto conforme a avanzado la investigación y desarrollo del mismo ya con ideas mucho más maduras llegamos a concluir que un celular tiene muchas más funciones que las que vamos a utilizar, tiene un costo mayor y sobre todo mediante pruebas realizadas los celulares permiten el envío de SMS mediante comandos AT pero no la lectura desde un computador con excepción de celulares antiguos que ya no se encuentran a la venta, por lo que optamos por la búsqueda de otro hardware que se acople de mejor manera optimizando recursos y sobre todo brinde estabilidad como hardware de envío, recepción de mensajes con total control y administración de nuestra parte mediante la aplicación que desarrollaremos, En el mercado ecuatoriano de operadoras celulares existen tres, Movistar, Claro y CNT Alegro que aparte de dar el servicio de cobertura para teléfonos celulares para hacer llamadas también brindan servicios de internet móvil con teléfonos celulares y módems inalámbricos celulares, éstos últimos vienen únicamente con la capacidad de transferencia de datos y no para voz por lo que es ideal utilizar estos para nuestro objetivo de enviar y recibir mensajes sin desperdicio de recursos, completa estabilidad y control.

Ahora con nuestra atención a los módems inalámbricos que funcionan con señal celular realizamos una recopilación de características de cada uno de los módems que las operadoras celulares ofrecen a la venta en nuestro país. La información

que se muestra a continuación es procedente de las páginas web de características de equipos que las operadoras móviles han publicado.

Modem Alcatel X060A (Operadora Claro)

- UMTS 850/1900/2100 MHz
- Descargas hasta 3.6Mpbs
- Micro SD de hasta 16GB
- SMS
- Soporta Windows 2000/XP/Vista; Mac OS X10.4.9
- Funciona en redes GSM en las bandas 850, 900, 1800 y 1900 MHz y en redes WCDMA en las bandas 850, 1900 y 2100 MHz.

Modem E587 (Operadora Movistar)

- Wi-Fi 802.11b/g/n
- DC-HSPA+/HSPA+/HSPA/UMTS
- EDGE/GPRS/GSM
- DC-HSPA+ data service up to 43.2 Mbps
- HSUPA data service up to 5.76 Mbps
- Equalizer and receive diversity microSD Card Slot(Up to 32G)
- Data and SMS Service Plug and play

Modem E369 (Operadora Claro)

- HSPA+/UMTS 2100MHz/1900MHz/AWS/900MHz/850MHz
- GSM/GPRS/EDGE 850MHz/900MHz/1800MHz/1900MHz
- Equalizer and receive diversity
- HSUPA data service of up to 5.76 Mbps
- HSDPA data service of up to 21.6 Mbps
- UMTS PS domain data service of up to 384 kbps
- EDGE packet data service of up to 236.8 kbps
- GPRS packet data service of up to 85.6 kbps
- SMS based on CS/PS domain of GSM and WCDMA

- Plug and play (PnP)
- USSD
- Personal computer/Smart card (PC/SC) Driver.
- Standard USB interface (Type A).
- Supporting easy installation and uninstall of the driver on Windows XP SP2/SP3, Windows Vista SP1/SP2, Windows 7, Mac OS X 10.5 and 10.6 with latest upgrades.
- Mini ID.
- Support CPC, IC (Interference cancellation), APT.

Modem Huawei EC168 (CNT)

- MODEM CDMA EVDO ev. A (download max 3.1Mbps & upload max 1.8 Mbps.
- Conexión a PC por puerto USB 2.0 estándar.
- CDMA 1x (download & upload max 153.6 Kbps.
- Compatible con Windows XP, Windows Vista de 32 bits, Windows Seven y sistemas MAC OS 10.6.

Modem Huawei E173 (Movistar, Claro)

- MODEM 3.5G - HSUPA / HSDPA / EDGE / GPRS.
- Interfaz de administración con opción de envío y recepción de SMS incorporada.
- Slot de memoria externa para micro SD de hasta 8GB.
- Conexión a PC por puerto USB 2.0 estándar.
- Sistemas operativos soportados: Windows XP, Windows Vista de 32 bits, Windows Seven, MAC OS 10.5.5, 10.6.3. y Linux.
- Compatible con Windows XP, Windows Vista de 32 bits, Windows Seven y sistemas MAC OS 10.6.

Modem ZTE MF110 (CNT)

- MODEM 3.5G - HSUPA / HSDPA / EDGE / GPRS.
- Interfaz de administración con opción de envío y recepción de SMS incorporada.
- Slot de memoria externa para micro SD de hasta 4GB.
- Conexión a PC por puerto USB 2.0 estándar.
- Compatible con Windows XP, Windows Vista de 32 bits, Windows Seven y sistemas MAC OS 10.6

Con esta información realizamos un cuadro comparativo para facilitar el análisis y resumir características importantes afines a nuestro objetivo, para poder realizar la compra de los módems que mejor se acoplen según sus características y operadora celular.

	Operadora que ofrece	Bandas	Velocidad de descarga hasta	Velocidad de subida	Plataforma S.O.	Tipo de red	Slot memoria externa	Extra
Modem Alcatel X060A	Claro	850/900/1800/1900/2100 MHz	3.6 Mbps	3.6 Mbps	Windows 2000/XP/Vista; Mac OS X10.4.9	GSM, WCDMA	SD hasta 16GB	
Modem E587	Movistar	850/900/1800/1900 MHz	5.76 Mbps	5.76 Mbps	XP/Vista/7	GSM,GPRS,EDGE	SD hasta 32GB	Wi-Fi 802.11 b/g/n
Modem E369	Movistar	850/900/1800/1900/2100MHz	5.76 Mbps	5.76 Mbps	XP/Vista/7; , Mac OS X 10.5 /10.6	GSM,GPRS,EDGE		
Modem Huawei EC168	CNT	CDMA2000 1xEV-DO	3.1Mbps	1.8Mbps	XP/Vista de 32 bits/7; MAC OS 10.6	CDMA EVDO		
Modem Huawei E173	Movistar, Claro	850/900/1800/1900 MHz	14Mbps	14Mbps	XP/Vista de 32 bits/7; MAC OS 10.5.5, 10.6.3. y Linux	3.5G - HSUPA / HSDPA / EDGE / GPRS	MicroSD hasta 8GB	
Modem ZTE MF110	CNT	850/900/1800/1900 MHz	14Mbps		XP/Vista de 32 bits/7; MAC OS 10.6	3.5G - HSUPA / HSDPA / EDGE / GPRS	MicroSD hasta 4GB	

Tabla 2.1: Resumen de características con el siguiente cuadro comparativo de Módems

Más adelante se especificara el o los módems que se usarían en el proyecto.

2.4. RECOPILACIÓN DE REQUISITOS.

Mediante una entrevista con el encargado del manejo administrativo y director del departamento de sistemas del Colegio Técnico Salesiano se recopilo la información que ya expresamos en la propuesta al inicio de este proyecto y mencionado en el capítulo 1, sin embargo aquí describimos las necesidades que se nos indicó en la entrevista, La extracción de las ideas principales son las siguientes:

- Existe la necesidad de hacer llegar comunicados o esquelas a los representantes legales de los estudiantes de esta institución de forma inmediata por lo que es necesario el desarrollo de un sistema que envié un SMS a los teléfonos celulares de dichas personas.

- La institución quiere innovar la manera en que los estudiantes realizan la consulta de sus notas, se quiere que tanto los padres de familia o representantes de los estudiantes o los mismos estudiantes no tengan que esperar a la reunión de curso para recibir los certificados de notas de las pruebas o exámenes rendidos en clases, si no se quiere que sea inmediato con una consulta que interactúe con el sistema a desarrollarse y leer las notas y promedios desde su celular mediante un SMS.

- Se requiere un medio de recordatorio de reuniones o avisos no sólo para los estudiantes sino también para profesores, directores de área, miembros de cierto departamento, comité central de padres de familia, grupos pertenecientes a la comunidad salesiana dependientes de la Unidad Educativa Técnico Salesiano, estudiantes que pertenezcan a cierto conjunto como es equipo de futbol o algún otro deporte, banda de guerra, animadores de batucada, grupo musical, etc.

- Al ser esta una institución educativa privada por cada estudiante se cobra una mensualidad, se requiere enviar un SMS a los estudiantes para alertar la fecha límite de pago, o hacer llegar un comunicado de mora o los mismos padres de familia realizar la consulta al sistema si está adeudando o no alguna mensualidad.

- Cada SMS que se envíe desde el sistema en desarrollo deberá ser almacenado en una base de datos para tener registrado un historial de qué texto se envió a quienes se envió, la fecha, hora y si realizo el envío con éxito o no.

2.5 DOCUMENTACION DE REQUISITOS

<table>
<tr><td colspan="2">INSTITUCIÓN:
Unidad Educativa Técnico Salesiano</td></tr>
<tr><td>PROCESO:
Consulta de notas o pensiones.</td><td>USUARIOS INTERVIENEN:
Estudiante, representante del estudiante.</td></tr>
<tr><td colspan="2">USUARIO RELATA EL PROCESO:
Director del departamento de sistemas.</td></tr>
<tr><td colspan="2">DESCRIPCIÓN:
- El solicitante envía un SMS al número que posteriormente se adquirirá con el número de cédula, un espacio y la palabra clave que puede ser “notas” o “pensiones”.
- El sistema recibe el mensaje, realiza la consulta a la base de datos con los parámetros recibidos en el mensaje.
- Responde el mensaje con los datos obtenidos de la base de datos al solicitante con las notas obtenidas hasta el momento o pensiones pendientes.</td></tr>
<tr><td>FIRMA RELATA:

_____
Ing. Pablo Durazno</td><td>FIRMA ANLISTA:

_____ _____
Jorge Calderón V. Rolando Vicuña M.</td></tr>
</table>

<table>
<tr><td colspan="2">INSTITUCIÓN:
Unidad Educativa Técnico Salesiano</td></tr>
<tr><td>PROCESO:

Envío de SMS para una instancia.</td><td>USUARIOS INTERVIENEN:

Instancias:
Administrativa -> Rector
Académica -> Vicerrector, Inspector
Alumnos -> Presidente del consejo estudiantil
Padres de familia -> Presidente de comité central de padres de familia
Institución -> director de institución.</td></tr>
<tr><td colspan="2">USUARIO RELATA EL PROCESO:

Director del departamento de sistemas.</td></tr>
<tr><td colspan="2">DESCRIPCIÓN:

- El rector o persona autorizada envía una solicitud al encargado de envío de SMS adjuntando el texto que debe ir en el mensaje y a que grupo o instancia va dirigido el mensaje
- encargado de envío de SMS ingresa el texto del recordatorio verificando que no exceda los 160 caracteres.
- Selecciona la instancia a quienes se enviara el mensaje.
- Envía el mensaje, y se almacena en la base de datos el mensaje y destinatarios.</td></tr>
<tr><td>FIRMA RELATA:

_____
Ing. Pablo Durazno</td><td>FIRMA ANLISTA:

_____ _____
Jorge Calderón V. Rolando Vicuña M.</td></tr>
</table>

2.6 DIAGRAMAS DE LA APLICACIÓN.

2.6.1 DIAGRAMAS DE CASO DE USO.

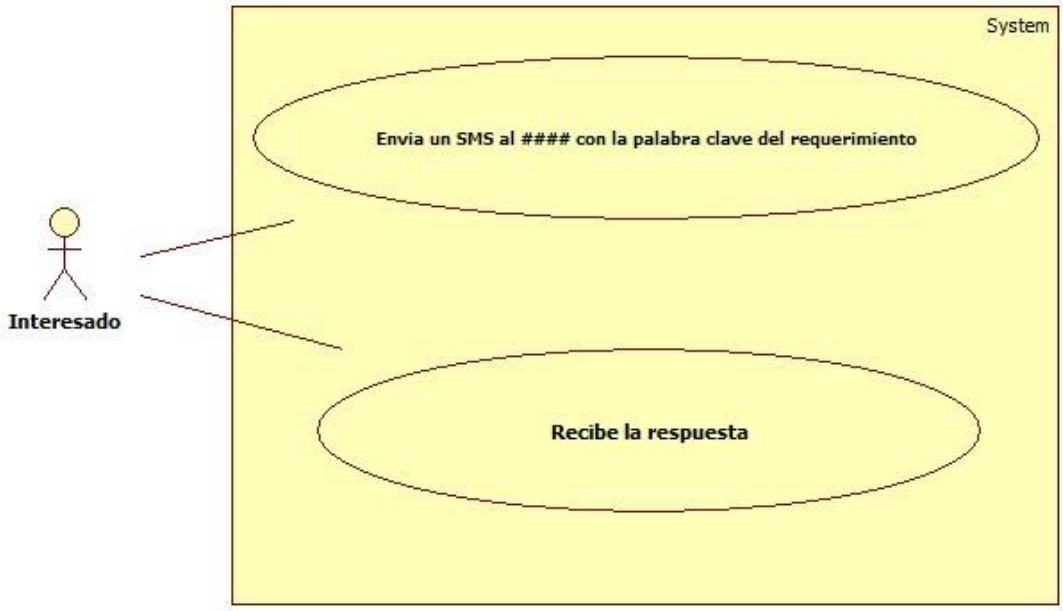


Gráfico 2.1: Solicitud de envío de calificaciones

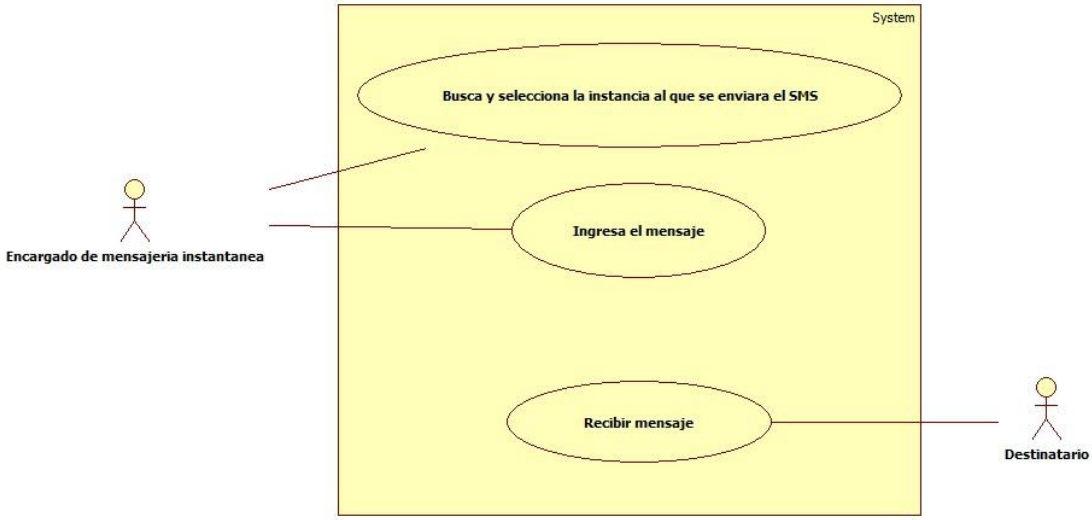


Gráfico 2.2: Envío de SMS informativos o recordatorios.

2.6.2 DIAGRAMAS DE SECUENCIA.

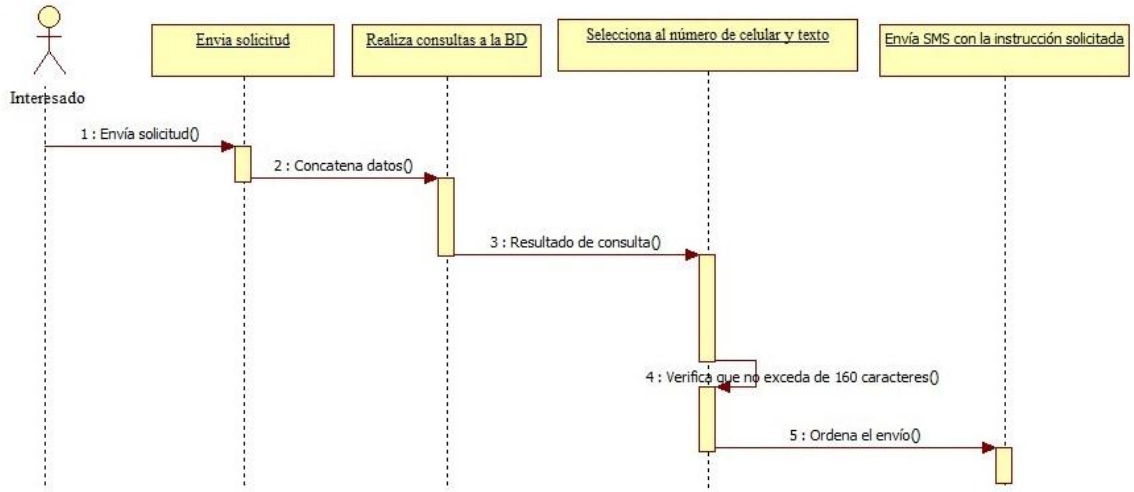


Gráfico 2.3: Solicitud de información.

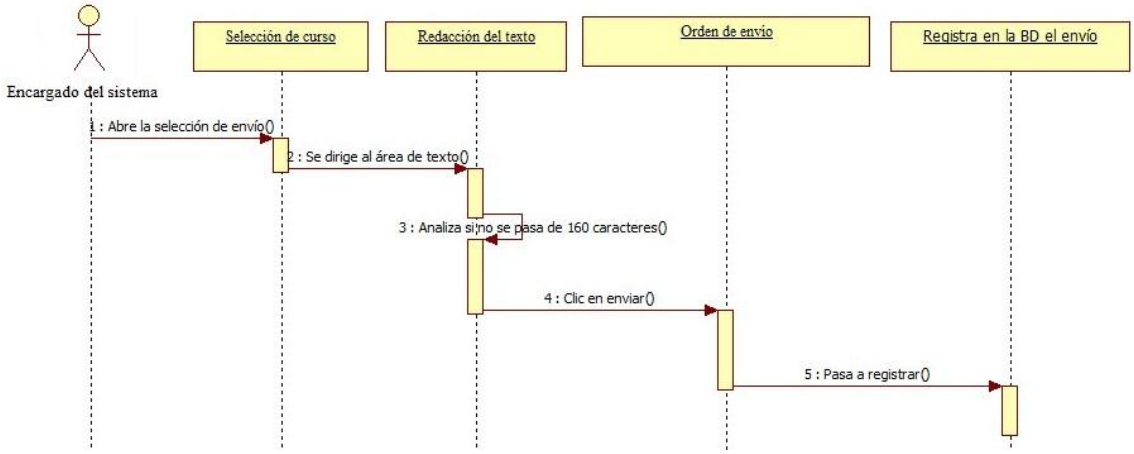


Gráfico 2.4: Envío de SMS informativos o recordatorios.

2.6.3 DIAGRAMA DE COMPONENTES.

Componentes del sistema de mensajería instantánea.

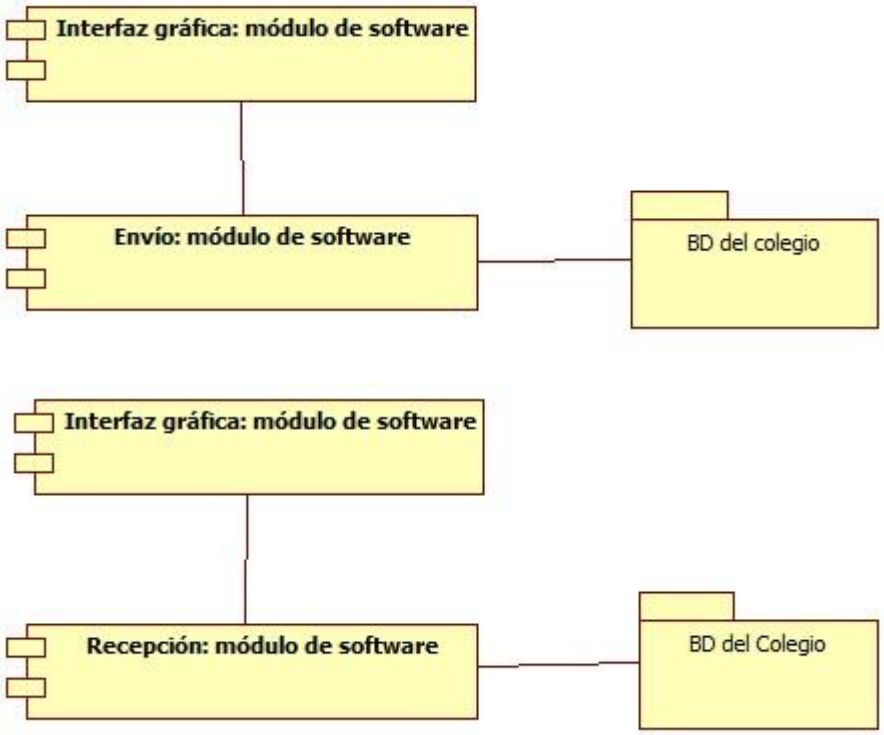


Gráfico 2.5: Diagrama de componentes del sistema.

2.7 DEFINICIÓN DEL ALCANCE.

El alcance ya ha sido especificado en el capítulo anterior.

CAPITULO 3

DISEÑO

3 DISEÑO

3.5 ARQUITECTURA DE LA APLICACIÓN.

El desarrollo de este proyecto es una aplicación distribuida de dos niveles, entonces la arquitectura quedara definida por un modelo cliente-servidor, la hemos dividido en dos módulos que básicamente hacen lo mismo la única diferencia es que el primero solo envía SMS y el segundo módulo recibe a través de SMS peticiones los procesa y genera una respuesta enviando otro SMS.

Específicamente para el envío de SMS el cliente está en una interfaz desarrollada en Oracle APEX que la unidad educativa se encargara de hacer mediante el trabajo de estudiantes del colegio que realizan una tesis para bachillerato; dicha interfaz enviara los datos a una URL que esta publicada por un servidor web Apache Tomcat, una vez accedida a la interfaz mediante un browser se procede a editar el texto y especificar cuál será la instancia de destinatarios de los mensajes.

En el computador que se designe estarán alojadas las clases Java desarrolladas, así como el JDK, librerías Java, DLLs y drivers que permitirán el correcto funcionamiento e interacción con el hardware, en éste caso un modem GSM. Este computador accederá mediante la red al servidor de base de datos para almacenar las transacciones realizadas para llevar un historial de los mensajes enviados.

Para un mayor entendimiento expondremos la arquitectura de envío de mensajes con el siguiente gráfico 3.1.

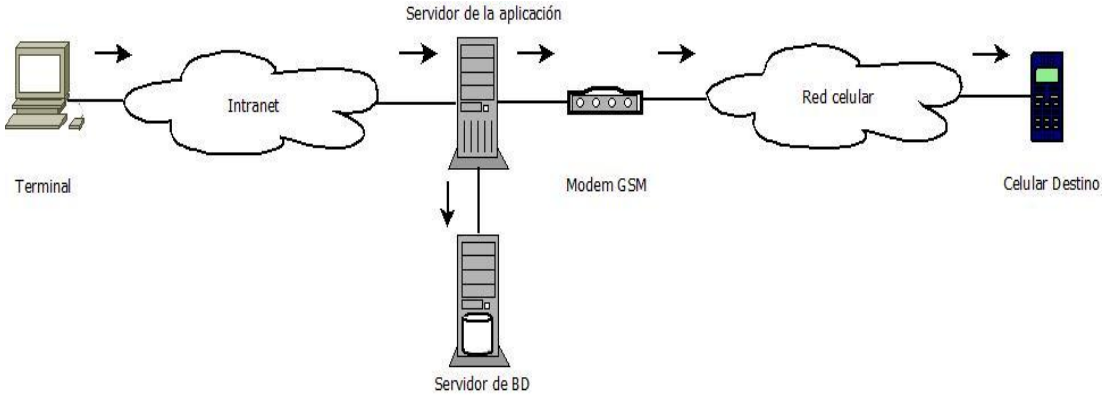


Gráfico 3.1: Arquitectura de la aplicación: Modulo de envío de información.

De igual manera para explicar el módulo de recepción y envío de SMS nos apoyaremos del siguiente gráfico.

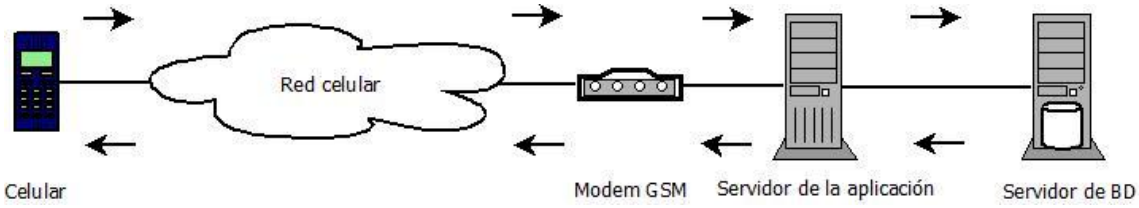


Gráfico 3.2: Arquitectura de la aplicación: Modulo de recepción y envío de respuesta

Un individuo con su celular manda un SMS con la palabra clave (palabras clave que se definieron en el capítulo 2 ítem 2.5 Documentación de requisitos) y número de cédula del estudiante de quién se desea obtener información, el sistema hecho en Java alojado en el computador de la aplicación estará escuchando eventos que se den en el modem en espera de la llegada de mensajes, para cuando llegue alguno, procesarlo, realizar la respectiva consulta a la base de datos, generar una respuesta, esta repuesta retorna a la aplicación y envía mediante un SMS al número celular del interesado que inició este proceso.

3.6 REQUERIMIENTOS DE SOFTWARE.

3.6.1 REQUISITOS FUNCIONALES.

- El sistema debe permitir enviar mensajes de texto a celulares desde un computador.

- El sistema debe tener la capacidad de recibir solicitudes de consultas a la base de datos transparente al usuario a través de mensajes de texto; un usuario puede consultar las notas o las pensiones, las palabras clave deben ser “notas” o “pensiones” y la respuesta de dicha consulta debe llegar al solicitante mediante un mensaje que envié el sistema.

- El sistema debe administrar autónomamente el hardware que envíe los mensajes.

- El sistema no tendrá interfaz gráfica, debe recibir parámetros que otro sistema desarrollado por la Unidad Educativa le enviara por ejemplo un número de celular del destinatario y el texto que contendrá el mensaje.

3.6.2 REQUISITOS NO FUNCIONALES

- El sistema debe estar siempre en marcha para que los usuarios puedan hacer uso de éste en cualquier momento.

3.6.3 REQUISITOS DE RENDIMIENTO

El sistema de SMS deberá soportar una carga de al menos una consulta por estudiante cada dos meses, que es cuando la Unidad Educativa realiza evaluaciones de desempeño académico y seguramente los estudiantes o padres de familia verificaran o harán seguimiento de sus notas. Existen 1799 estudiantes; según estadísticas del colegio que se realizó con otro motivo, se descubrió ésta necesidad no cubierta, que es de un sistema de mensajería instantánea; y mediante observación se determina que un 90% de estudiantes tienen celular; para más detalles de dichas estadísticas, los resultados tabulados de la encuesta se encuentran adjuntas en ANEXOS. Es menester

aclarar que la Unidad Educativa Técnico Salesiano y la Escuela Carlos Crespi utilizan el mismo sistema para los dos establecimientos y este está manejado por el departamento de sistemas del colegio, estos establecimientos son Salesianos manejados por una misma política y por lo tanto todo sistema que se implemente para uno de los establecimientos es para uso de ambos, por lo tanto existe más beneficiarios del sistema de mensajería instantánea ya que la escuela tiene 1084 alumnos sumados en total son 2883 estudiantes que interactuarán directa o indirectamente con el sistema de mensajería masiva del que se trata este proyecto.

3.6.4 SEGURIDAD

El computador que procesará la aplicación de mensajería no tendrá una IP pública ni tendrá acceso a internet, no se podrá tener acceso directo a este computador desde internet, esto por seguridad y resaltando que no existe la necesidad por que la aplicación se comunica con los usuarios finales mediante red celular y mediante intranet con la base de datos y la interfaz gráfica de Oracle APEX.

Aparte de las seguridades que brinde el sistema operativo también instalamos un antivirus para brindar seguridad al equipo, el antivirus instalado es “Avast Free Antivirus” versión 7.0.1426 que es una versión gratuita de libre descarga en internet del sitio web de los creadores.

Nuestro software para la comunicación con el modem tan solo abrirá un puerto COM para comunicación directa con un modem, para lectura o envió de mensajes; La comunicación entre el software y el modem es cifrada a código binario mediante comandos AT.

3.6.5 DISPONIBILIDAD Y FIABILIDAD

El sistema siempre estará a la escucha de eventos que pueden darse de dos tipos:

1. Que un sistema externo mande parámetros para enviar un mensaje (esto para el módulo de envío masivo de mensajes).

2. Que un usuario realice una consulta de notas o pensiones al sistema (esto para el módulo de envío, recepción).

Para el primero existirá instalado un servidor WEB Apache Tomcat para que el sistema externo que mande los parámetros de mensajes apunte a esa dirección URL que tendrá el equipo de alojamiento de la nuestra aplicación; esto se hará de esta manera por que la Unidad Educativa tendrá la interfaz gráfica del sistema de mensajería hecha en Oracle APEX y este sólo puede apuntar a URLs para enviar información a otro equipo. Por lo tanto el sistema será fiable y no existirán errores de tipo de falla de software mientras el servicio de Apache Tomcat este alzado para que el sistema esté 100% disponible cuando se requiera su uso.

El sistema estará funcional, completamente disponible y se garantizara la fiabilidad de éste mientras exista saldo o paquetes de mensajes activos en la cuenta celular del SIM que esté instalado en el modem; Vigilar esto es responsabilidad de la persona a quien delegue la Unidad Educativa como administrador de éste sistema, más detalles de esto lo explicamos en el punto.

3.6.6 MANTENIBILIDAD

Las tareas de mantenimiento las realizara el administrador del sistema de mensajería masiva que será un miembro del departamento de sistemas y será designado por el director de dicho departamento.

En cuanto al software si se llegase a tener que hacer un mantenimiento sería en cuanto a configuración de puertos si en algún momento cambian de lugar en los puertos físicos del computador o aumentan o disminuyen los módems.

Como complemento al mantenimiento se hará una alarma que indique al administrador si el paquete de mensajes esta por acabarse, entonces cuando solo falten pocos SMS (número de SMS configurable por el administrador) en el browser del cliente saldrá una ventana de aviso y luego de eso se detendrá el módulo.

3.6.7 PORTABILIDAD

El Lenguaje de programación es Java y es multiplataforma para su funcionamiento necesita antes tener instalado la JVM (máquina virtual de Java).

El software estará instalado en un sistema operativo Microsoft Windows XP Profesional por que el computador que nos entrega la Unidad Educativa tiene instalado esta plataforma (esto está justificado con la debida documentación que adjuntamos en ANEXOS). Al estar desarrollado en ésta plataforma nuestro sistema es 100% compatible con otras plataformas Windows pero no con sistemas operativos basados en GNU/Linux, Unix, Mac OS u otras, porque al estar en Windows utilizamos DLLs que sólo funcionan en ésta plataforma, además en otras plataformas existe la complejidad de tener que utilizar otros drivers para los módems (con la compra del módems solo vienen drivers para Windows), diferente forma de abrir puertos, diferente manejo para dar permisos de seguridad entre otros, y para que sea portable a otro sistema necesitaría ser pensado desde un principio de su desarrollo en la plataforma escogida.

3.6.8 OTROS REQUISITOS

Ya que el proyecto estará desarrollado en lenguaje de programación Java, se necesitara tener instalado la JVM (máquina virtual de Java) para interpretar y ejecutar las instrucciones escritas para la aplicación.

También es necesario Instalar el driver del modem GSM para permitir el funcionamiento y por ende la comunicación y transferencia de datos con el computador.

Y por último tener instalado el Apache Tomcat para publicar en la red el módulo de Envió masivo.

3.7 REQUERIMIENTOS DE HARDWARE

3.7.1 COMPUTADOR QUE ALOJARA LA APLICACION

Para que la aplicación se ejecute con normalidad no es necesario grandes requerimientos de hardware ya que se desarrollara usando la tecnología de (4)Java que es muy liviana en cuanto a consumo de memoria RAM para el funcionamiento de la JVM (Máquina Virtual de JAVA), ya que en los casos más críticos los mensajes se encolarán, se enviarán o recibirán uno a la vez, es decir que las variables y consumo de memoria que se genere en un mensaje luego de ser procesado se liberará, dando espacio al nuevo proceso que generará el siguiente mensaje, en Microsoft Windows XP la JVM (Máquina Virtual de JAVA) necesita físicamente:

Memoria RAM:	64MB
Memoria ROM:	espacio libre de 98MB

Según la (5)página oficial de Apache Tomcat los requerimientos mínimos para su funcionamiento son:

Memoria RAM:	128MB
Memoria ROM:	espacio libre de 24MB

El computador que nos entregaran será exclusivo para la aplicación de SMS y el sistema operativo que está instalado en éste y alojara el software de mensajería es Microsoft Windows XP Profesional que por experiencia de

manejo de esa plataforma sabemos que un (6)requisito mínimo para su buen funcionamiento es de:

Velocidad del Procesador:	300MHz
Memoria RAM:	64MB
Memoria ROM:	espacio libre de 3.2GB.

Por lo tanto como requerimientos mínimos recomendamos una máquina que tenga:

Velocidad del Procesador:	800MHz o mayor
Memoria RAM:	512MB o mayor
Memoria ROM:	espacio libre de 2GB.
Puertos USB:	1 puerto USB por modem GSM.
Puerto LAN:	1 puerto LAN para conexión con la red.

El computador que nos entrega la Unidad Educativa que alojara nuestra aplicación tiene las siguientes características:

Procesador:	Intel(R) Pentium(R) 4 / 2.80GHz
Memoria RAM:	512MB
Memoria ROM:	40GB
Puertos USB:	6
Puerto LAN:	1
Puerto Paralelo:	1
Puerto Serial:	1
Unidad de Disquete:	1
Unidad de CD/DVD:	1

3.7.2 MODEM GSM

El modem GSM que se use debe soportar comandos AT para poder establecer la comunicación con el computador que alojará la aplicación y al cual estará conectado éste, también debe tener una memoria interna para almacenar los

SMS ya que es allí donde llegan en primera instancia los mensajes de petición de información al sistema, generalmente todo modem cuenta con esta memoria interna, y como es normal un soquet donde se inserta una tarjeta SIM para tener una identificación al acceder a la red celular de las operadoras que dan este servicio.

El modelo del modem adquirido para este proyecto es HUAWEY Mobile Broadband E173, elegimos este modelo y marca porque es un Modem que no tiene bloqueo alguno en el registro de la operadora que la vende, es decir que puede trabajar con SIM de MOISTAR o SIM de CLARO independientemente de donde se haya adquirido; el tamaño de hardware es relativamente pequeño a comparación con otros módems esto es importante ya que si es un modem muy grueso y los puestos USB del computador vienen juntos al conectar varios módems se obstaculizan entre ellos y no es posible conectar varios; Previo a la adquisición se realizó una prueba con dicho modem y si soporta los comandos AT necesarios para este proyecto.

3.7.3 JUSTIFICACIÓN HERRAMIENTA A UTILIZAR

En un principio se hizo la propuesta por parte de la Unidad Educativa de desarrollar este sistema en tecnología Oracle APEX, la que luego de indagar, la rechazamos porque dicha tecnología fue desarrollada pensada para el rápido acceso a base de datos y en general visualización de ventanas para gestión de datos y ésta no permite abrir puertos y comunicación con hardware. Entonces utilizamos el lenguaje de programación Java por ser una tecnología que permite abrir comunicación a bajo nivel, con el hardware, abriendo puertos en este caso necesitamos abrir el puerto COM, y sobre todo la razón principal es que la Unidad Educativa nos propuso que ya que no se puede trabajar directamente con modem desde Oracle APEX, trabajar en lenguaje Java, disposición que aceptamos luego de recibir un documento que esta adjunto en ANEXOS.

El software desarrollado no es un aplicativo de uso delicado en cuanto a seguridad ni hace transacciones delicadas que afecten el sistema completo de la Unidad Educativa que puedan causar una caída, detener parcial o totalmente el sistema; por lo que utilizar un servidor para instalar el software de nuestro proyecto y manejo de módems GSM sería desperdiciar recursos por que sólo alojara clases Java que se conectarán con un servidor de base de datos, es decir toda la carga en realidad estará en la consulta que realiza nuestra aplicación al servidor de base de datos; Por ello se nos entregó un computador instalado con la plataforma Windows XP Profesional que es donde quieren que este nuestro software; Otro motivo por el que se designa dicha maquina es que no quieren que los módems GSM que se van a utilizar estén conectados en los servidores de la institución y prefieren tener por separado como un subsistema y no parte de la totalidad del sistema del establecimiento.

3.8 MODELO DE DATOS

3.8.1 DIAGRAMA DE CLASES

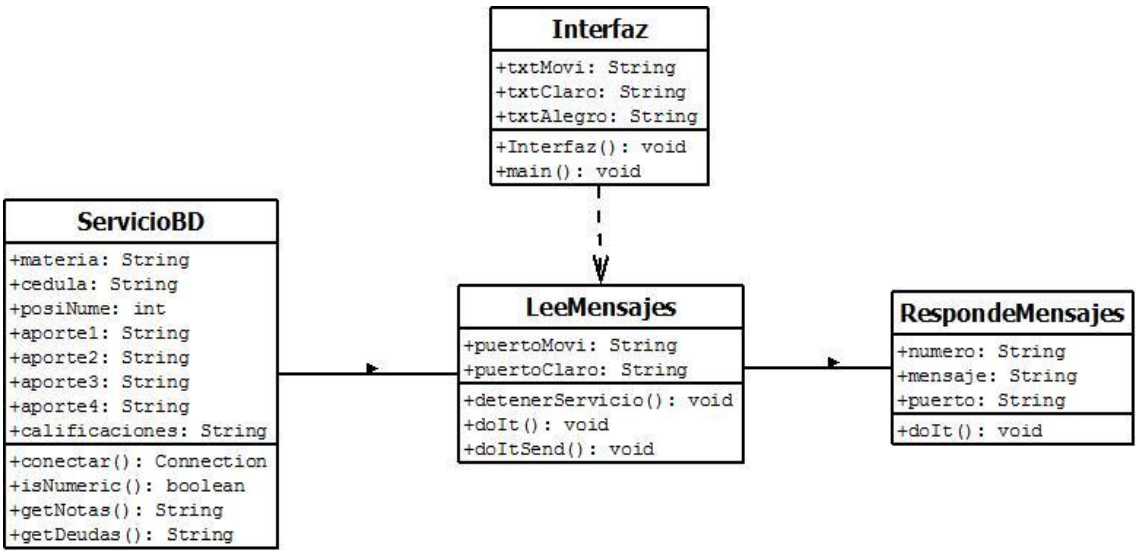


Gráfico 3.3: Diagrama de clases del módulo de recepción /envío

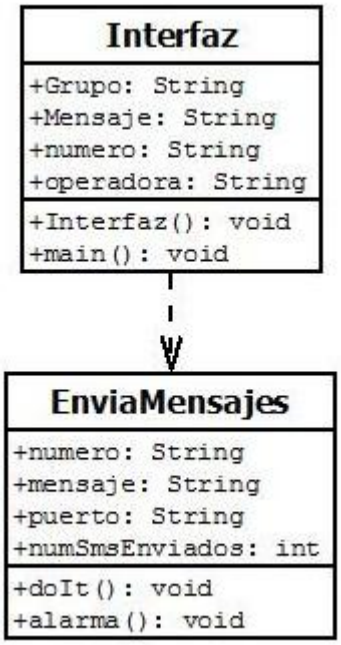


Gráfico 3.4: Diagrama de clases de módulo de envío masivo de SMS.

3.8.2 DIAGRAMA ENTIDAD RELACIÓN

Por políticas del departamento de sistemas y del establecimiento no podemos conectarnos directamente al usuario administrador de la base de datos por seguridad de contraseñas, evitar divulgación de información, uso indebido, intencional o mal intencionado del sistema; por ello se creará un usuario exclusivo para el subsistema de mensajería instantánea en el que se alojarán dos tablas que son un resumen de otras tablas que contienen los datos necesarios de los estudiantes a los cuales se necesita acceder para realizar las consultas tanto de notas como pensiones; las tablas resumen se llenan con datos automáticamente con triggers y procedimientos almacenados que el DBA(Administrador de Base de Datos) de la Unidad Educativa realizo para replicar o actualizar los datos tanto en el usuario administrador como en el usuario de mensajería instantánea.

Las tablas del usuario de mensajería instantánea son:

SMS_PENS		
SMS_PENS_CEDULA_ALUMNO	Varchar2(15)	NN
SMS_PENS_MES_PAGO	Number(x,y)(2,0)	NN
SMS_PENS_VALOR	Number	
SMS_PENS_NOMBRE_MES	Varchar2(3)	NN
SMS_PENS_ESTADO_PUBLICACION	Varchar2(1)	
SMS_PENS_FECHA_CREACION	Date	
SMS_PENS_USUARIO_CREACION	Varchar2(30)	
SMS_PENS_FECHA_MODIFICADO	Date	
SMS_PENS_USUARIO_MODIFICADO	Varchar2(30)	

SMS_NOTAS		
SMS_NOTAS_ESPECIALIDAD	Varchar2(45)	NN
SMS_NOTAS_CURSO	Varchar2(30)	NN
SMS_NOTAS_PARALELO	Varchar2(2)	NN
SMS_NOTAS_MATERIA	Varchar2(75)	NN
SMS_NOTAS_ABREVIATURA	Varchar2(10)	NN
SMS_NOTAS_ESTUDIANTE	Varchar2(60)	NN
SMS_NOTAS_CEDULA	Varchar2(15)	NN
SMS_NOTAS_APORTE1	Number(x,y)(2,0)	
SMS_NOTAS_APORTE2	Number(x,y)(2,0)	
SMS_NOTAS_APORTE3	Number(x,y)(2,0)	
SMS_NOTAS_APORTE4	Number(x,y)(2,0)	
SMS_NOTAS_ESTADO_ACADEMICO	Varchar2(1)	
SMS_NOTAS_ESTADO_FINANCIERO	Varchar2(1)	
SMS_NOTAS_ESTADO_PUBLICACION	Varchar2(1)	
SMS_NOTAS_FECHA_CREACION	Date	
SMS_NOTAS_USUARIO_CREACION	Varchar2(30)	
SMS_NOTAS_FECHA_MODIFICADO	Date	
SMS_NOTAS_USUARIO_MODIFICADO	Varchar2(30)	

Gráfico 3.5: Tablas de la base de datos del usuario SMS

3.9 DESARROLLO

Como ya se explicó en los puntos anteriores de éste y otros capítulos cual será la arquitectura y las tecnologías a utilizar, ahora implementaremos el código de programación utilizando el IDE (*Integrated Development Environment*) Eclipse que es software de libre distribución y código abierto.

Se explicará a continuación una parte del código que se puede considerar como la base fundamental de la aplicación desarrollada en el módulo de envío:

En primer lugar se establece las dependencias necesarias para que pueda funcionar con éxito el proyecto:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Date;
```

A continuación se detalla la implementación de la Clase que provee los servicios de la base de datos:

```
public class ServicioBD {
        public static Statement s;
        static Connection con;

        public static Connection conectar()
        {
           String url = "jdbc:oracle:thin:@localhost:1521:XE";
                        try
                        {
                                Class.forName("oracle.jdbc.driver.OracleDriver");
                                 con =DriverManager.getConnection(url,"system","cuervo");
                                 s = con.createStatement();
                 }
                        catch (SQLException e)
                        {
                                        System.out.println("Error SQL ");
                                        System.out.println(e.getMessage());
                }
                        catch (Exception ea)
                        {
                System.out.println("Error Exception al conectar");
                        }
        return con;
        }
```

Así mismo para poder realizar las validaciones de la cédula se implementa el siguiente método:

```
private static boolean isNumeric(String cadena){

                try {
                        Integer.parseInt(cadena);
                        return true;
                } catch (NumberFormatException nfe){
                        return false;
                }
        }
```

Otro método importante es el que devuelve las notas de un estudiante pasando como parámetro el número de cedula:

```
        public static String  getNotas(String ced) {
                int posiNume=0;
                for(int i=0;i<=ced.length()-1;i++){
                        System.out.println(ced.substring(i, i+1));
                        if(isNumeric(ced.substring(i, i+1))){
                                posiNume=i;
                                break;
                        }
                }
                String cedula=ced.substring(posiNume, posiNume+11);
                System.out.println("cedula->"+cedula);
```

```
Connection con=conectar();
        try{

String  calificaciones="";

PreparedStatement p = con.prepareStatement( "select sms_notas_abreviatura,
sms_notas_aporte1, sms_notas_aporte2, sms_notas_aporte3, sms_notas_aporte4 "
+
" from sms_notas where sms_notas_cedula= '"+cedula+"'");

ResultSet r = p.executeQuery();

while (r.next()){
                String materia=r.getString("sms_notas_abreviatura");
                String aporte1=r.getString("sms_notas_aporte1");
                String aporte2=r.getString("sms_notas_aporte2");
                String aporte3=r.getString("sms_notas_aporte3");
                String aporte4=r.getString("sms_notas_aporte4");
                        calificaciones=calificaciones+materia+" "+aporte1+"
"+aporte2+" "+aporte3+" "+aporte4+"; ";
        }
                System.out.println(calificaciones.length());
                System.out.println("Sus calificaciones son->"+calificaciones);
                return calificaciones;

        }
        catch(Exception e){
                e.printStackTrace();
                return null;
        }
}
```

Otra información solicitada por los usuarios son las deudas que tienen los alumnos con la institución, esto se consigue mediante la implementación del siguiente método:

```
public static String  getDudas(String ced) {

        java.util.Date fecha = new Date();

        int mes_actual=fecha.getMonth()+1;
        String deuda="";
        mes_actual=getMes_actual(mes_actual);//extrahemos el mes actual

        if(mes_actual!=0){

        Connection con=conectar();

        try{

        PreparedStatement p = con.prepareStatement( "select SMS_PENS_MES_PAGO,
        SMS_PENS_VALOR as valor, SMS_PENS_NOMBRE_MES as mes "
        +
        " from SMS_PENSIONES where SMS_PENS_CEDULA_ALUMNO='"+ced+"' and
        SMS_PENS_MES_PAGO <='"+mes_actual+"'");
```

```
ResultSet r = p.executeQuery();

while (r.next()){
double valor=Double.parseDouble(r.getString("valor"));
String mes=r.getString("mes");
deuda=deuda+mes+"->"+valor+" ";
//System.out.println("valor->"+valor);
//System.out.println (mes+"->"+valor);
//deudas="Su deuda es->"+"$"+r.getString("deuda");
}
}
catch(Exception e){
e.printStackTrace();
return null;
}
System.out.println(deuda);
return deuda;
}
else {
deuda="NO SE ENCUENTRA EN PERIODO DE CLASES";
System.out.println("NO SE ENCUENTRA EN PERIODO DE
CLASES");
return deuda;
}
}
public static int getMes_actual(int mesSys){
int mes=0;
if(mesSys==9)
mes=1;
if(mesSys==10)
mes=2;
if(mesSys==11)
mes=3;
if(mesSys==12)
mes=4;
if(mesSys==1)
mes=5;
if(mesSys==2)
mes=6;
if(mesSys==3)
mes=7;
if(mesSys==4)
mes=8;
if(mesSys==5)
mes=9;
if(mesSys==6)
mes=10;
return mes;
}
```

Por último un método muy sencillo es el que ocupamos para para mostrar un mensaje en la pantalla

```
public static void mensaje(String men){
Interfaz i=new Interfaz();
i.mensaje(men);
}
}
```

Otra parte muy importante de la aplicación es la que se usa para iniciar los servicios en el módulo de Recepción y Envío, en primer lugar se detalla el método que inicializa o prepara los modem

```
public void doIt(String puertoMovis, String puertoClaros) throws Exception
    {
```

declaramos las variables para los modem de movistar y claro

```
        puertoMovi=puertoMovis;
        puertoClaro=puertoClaros;

        try
        {
```

colocamos las características del Modem claro

```
            gateway1 = new
            SerialModemGateway("modem."+puertoMovi.toLowerCase(),
            puertoMovi, 115200, "Huawei", "E173");
           gateway1.setProtocol(Protocols.PDU);
            gateway1.setInbound(true);
            gateway1.setOutbound(true);
            gateway1.setSimPin("0000");
```

fin del modem claro

colocamos las características del Modem movistar

```
           gateway2 = new SerialModemGateway("modem."+puertoClaro.toLowerCase(),
           puertoClaro, 115200, "Huawei", "E173");
            gateway2.setProtocol(Protocols.PDU);
            gateway2.setInbound(true);
            gateway2.setOutbound(true);
            gateway2.setSimPin("0000");
```

fin del modem movistar

Añadimos los servicios que provee el paquete smsLib

```
            Service.getInstance().setInboundMessageNotification(inboundNotification);
            Service.getInstance().setCallNotification(callNotification);
            Service.getInstance().setGatewayStatusNotification(statusNotification);

        Service.getInstance().setOrphanedMessageNotification(orphanedMessageNotificati
        on);
            Service.getInstance().addGateway(gateway1);
            Service.getInstance().addGateway(gateway2);
```

Iniciamos los modem de movistar y claro

```
    Service.getInstance().startService();
            System.out.println("gateway1->"+gateway1.getStatus());
            System.out.println("gateway2->"+gateway2.getStatus());
            System.out.println("SERVICIO->"+Service.getInstance().getServiceStatus());
            System.out.println("Escuchando...");
        }
        catch (Exception e)
        {
            e.printStackTrace();
        }

    }
```

Y por último el método neurálgico de todo el sistema se detalla a continuación, este sirve para enviar los SMS a través de los módems.

```
public void doItSend(String numero, String mensaje, AGateway gateway) throws Exception

{
```

Cargamos el número celular y el cuerpo del mensaje

```
        OutboundMessage msg = new OutboundMessage(numero, mensaje);
        Service.getInstance().addGateway(gateway);//utilizamos un modem para enviar
        Service.getInstance().sendMessage(msg);//enviamos el mensaje
}
```

CAPITULO 4

IMPLEMENTACIÓN

4 IMPLEMENTACIÓN

4.1 INSTALACIÓN

4.1.1 INSTALACIÓN DEL APLICATIVO SMS ENVÍO MASIVO

Copiamos el archivo .war que se encuentra ubicado en la carpeta envió masivo en la dirección del servidor de aplicaciones web Apache Tomcat dentro de la carpeta wepapps por ejemplo C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps

Una vez alzado el servicio de Apache Tomcat y verificado el funcionamiento del servidor web como se indica en anexos dentro de instalación de Apache Tomcat, colocamos la siguiente dirección en el browser http://localhost:8080/enviomasivo/ si los pasos anteriores se han realizado con éxito deberíamos poder observar y tener en funcionamiento el módulo de envío masivo.

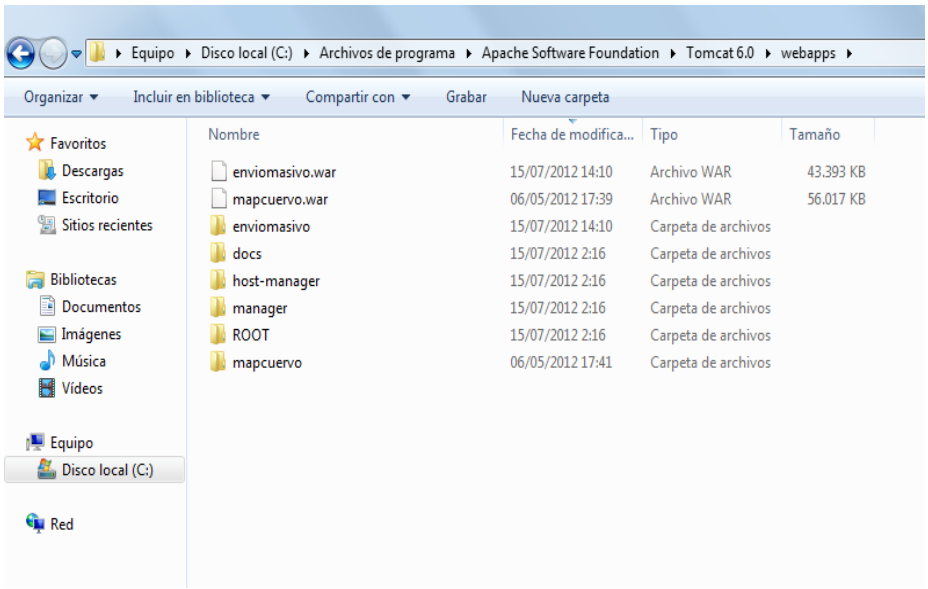


Gráfico 4.1: Muestra de la carpeta donde se coloca el archivo enviomasivo.war

4.1.2 INSTALACIÓN DEL APLICATIVO RECEPCIÓN

Este módulo se entregará en un archivo .exe por lo cual no se deberá tener ningún tipo de instalación previa, bastara con copiar la carpeta sms_recepción en cualquier ubicación del disco; dentro de esta carpeta se encontrara disponible el archivo .exe que se llamará smsRecepcion.exe; al dar clic sobre este archivo se abrirá la interfaz para el módulo de recepción SMS.

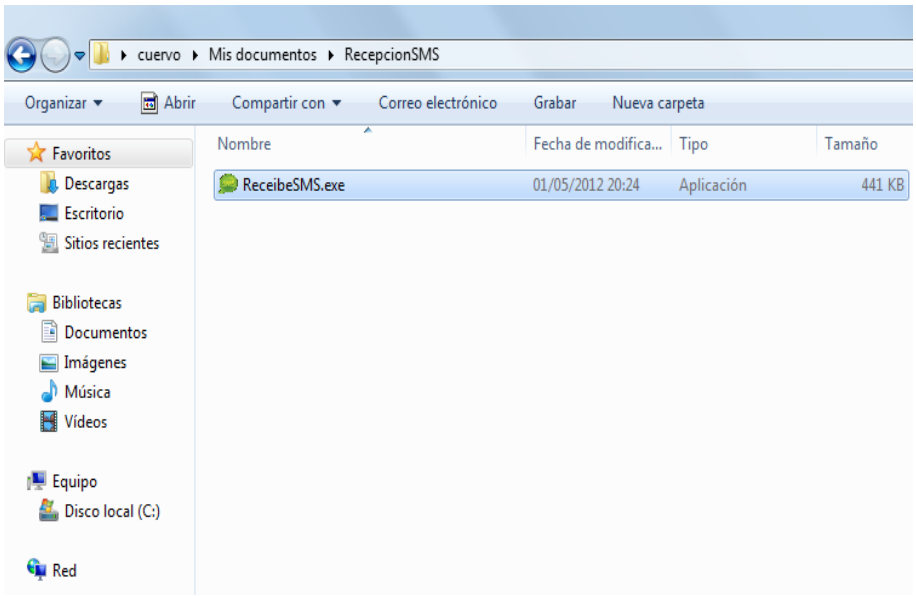


Gráfico 4.2: Muestra del archivo ejecutable recibeSMS.exe

4.2 CONFIGURACIÓN

(7)Es necesario instalar Java Comm API para el sistema operativo de Windows, es necesario tener los archivos:

- comm.jar
- win32com.dll
- javax.comm.properties

Son tres pasos que hay que realizar y se describen a continuación:

1. Guardar la DLL llamada win32com.dll en el directorio de JRE en nuestro caso en la dirección: C:\Program Files\Java\jre6\bin\
2. Guardar el archivo comm.jar que es el JAR para comunicaciones en la siguiente dirección: C:\Program Files\Java\jre6\lib\ext\
3. Guardar el archivo llamado javax.comm.properties en la siguiente dirección: C:\Program Files\Java\jre6\lib\

4.3 PRUEBAS

Para probar el rendimiento de los módulos desarrollados, utilizamos (8)pruebas de carga que nos darán resultados que permitirán evaluar tiempos de respuesta en condiciones extremas de uso del sistema para realizar una tabla comparativa de los resultados obtenidos versus los esperados.

Seleccionamos las pruebas de carga porque el software justamente trabajará en condiciones constantes de peticiones de información de parte de los usuarios y envío de información desde el sistema a los mismos, esto decimos ya que luego de instalado éste sistema estará en funcionamiento ininterrumpido.

4.3.1 MODULO DE ENVÍO

Sintetizamos las pruebas de este módulo en tres partes, la primera consta en enviar dos mensajes a la vez, uno mediante el modem de Movistar y otro por el de Claro, las lecturas se muestran a continuación.

Operadora	Número de SMS enviados	Número éxitos	Número de fracasos	Tiempo(seg)
Movistar	1	1	0	7:45
Claro	1	1	0	12:20
Totales	2	2	0	-

Tabla 4.1: Primera prueba del módulo de envío masivo.

Para la segunda prueba enviamos diez SMS, cinco para destinatarios Movistar y cinco para destinatarios Claro desde los respectivos módems los resultados mostramos a continuación.

Operadora	Número de SMS enviados	Número éxitos	Número de fracasos	Tiempo(seg)
Movistar	5	5	0	24:69
Claro	5	5	0	17:99
Totales	10	10	0	-

Tabla 4.2: Segunda prueba del módulo de envío masivo.

Para la tercera prueba de este módulo planificamos enviar 200 mensajes de texto, y tomar el tiempo que toma terminar esta acción, luego contactar a la mayoría de destinatarios posibles para que indiquen si recibieron el SMS y finalmente verificar los mensajes consumidos del paquete de mensajes adquirido para estas pruebas de esta manera confirmar si les llego el mensaje a todos los destinatario que no logremos contactar, en la siguiente tabla se indican los resultados obtenidos.

Operadora	Número de SMS enviados	Número éxitos	Número de fracasos	Tiempo(min)
Movistar/Claro	200	200	0	16:00

Tabla 4.3: Tercera prueba del módulo de envío masivo.

4.3.2 MODULO DE RECEPCIÓN

Al igual que el módulo anterior realizaremos pruebas de carga para ver si el sistema en algún momento pierde su normal estabilidad; realizamos varias pruebas en distintas circunstancias y tomamos nota de lo ocurrido para su posterior análisis. Aclaramos que los tiempos que se tomaron para este módulo fueron de la siguiente forma; el cronómetro arranca cuando pulsamos el botón enviar del celular luego de escrita la petición y paramos el cronómetro luego de que llegue el SMS de contestación al mismo celular.

La primera prueba la hicimos enviando una petición con un celular de Movistar y otro con un celular de Claro, cada uno a su respectivo modem, los resultados están en la siguiente tabla:

Operadora	Número de solicitudes	Número éxitos	Número de fracasos	Tiempo(seg)
Claro	1	1	0	35:37
Movistar	1	1	0	25:83

Tabla 4.4: Primera prueba del módulo de recepción/envío.

La segunda prueba la hicimos aumentando la carga, para esto mandamos de cuatro celulares al mismo tiempo, dos peticiones al modem de Movistar, uno pidiendo notas y otro pidiendo las pensiones; y otras dos peticiones al modem de Claro, así mismo, una pidiendo notas y otra pidiendo pensiones; Los resultados del sistema fueron los siguientes:

Operadora	Número de solicitudes	Número éxitos	Número de fracasos	Tiempo(min)
Movistar/Claro	4	4	0	01:30

Tabla 4.5: Segunda prueba del módulo de recepción/envío.

Para la tercera prueba tratamos de incrementar la carga al módulo con un celular más; Ahora haremos cinco peticiones al mismo tiempo tanto el modem movistar como al de Claro, consultando tanto notas como pensiones, estas pruebas se realizaron en horas de la mañana, obtuvimos los siguientes resultados.

Operadora	Número de solicitudes	Número éxitos	Número de fracasos	Tiempo(min)
Movistar/Claro	5	5	0	01:40

Tabla 4.6: Tercera prueba del módulo de recepción/envío.

La tercera prueba la repetimos para tomar nuevamente los tiempos y ver si el sistema se mantiene estable y también verificar los resultados, estas pruebas lo realizamos en horas de la noche.

Operadora	Número de solicitudes	Número éxitos	Número de fracasos	Tiempo(min)
Movistar/Claro	5	5	0	01:20

Tabla 4.7: Cuarta prueba del módulo de recepción/envío.

4.4 ANALISIS DE RESULTADOS

El análisis para el módulo de envío masivo es positivo, fue totalmente satisfactorio no existieron errores o fracasos en ninguna de las tres pruebas realizadas, en ningún momento se saturó ningún componente del sistema ya sea la programación realizada, el modem o la base de datos.

Se enviaron doscientos mensajes en 16 minutos lo que nos da una idea al realizar una ponderación que cuando enviemos 800 SMS, al sistema le tomaría más o menos una hora.

Para el módulo de recepción no tuvimos inconveniente alguno, tratamos de desestabilizar al sistema sobrecargándolo de peticiones enviando cinco consultas concurrentes y tuvimos excelentes resultados con cero fracasos y un tiempo de respuesta dentro de los parámetros aceptables; cabe indicar que en producción es poco probable que se dé una situación similar. La velocidad de respuesta de las pruebas depende de que tan congestionada esté la red celular, no es lo mismo realizar una prueba en el día o en la noche, pero tampoco se proyectan tiempos que llamen la atención o afecten a usuarios del sistema.

4.5 MANUALES DE USUARIO

4.5.1 MANUAL DE USUARIO DEL MÓDULO DE ENVÍO

Es fácil de explicar el funcionamiento del módulo de envió masivo y lo describimos a continuación:

- Abrir un Browser y escribir en la barra de direcciones lo siguiente: http:// localhost:8080/enviomasivo y aparecerá algo similar a la imagen que se muestra a continuación.

Gráfico 4.3: Vista del browser de la interfaz de inicio y configuración del módulo de envío masivo.

- Llenar los campos con el nombre del puerto COM que se haya abierto cuando se conectaron los módems al computador (para saber cómo se llama el puerto que se abrió al conectar el modem se puede utilizar un software de monitoreo de puertos o ver en Inicio > Panel de Control > Sistema > Administrador de Dispositivos para el sistema operativo Windows).

- Luego de completar el llenado de los nombres de los puertos dar clic en "Cargar", El browser cargara una nueva interfaz como la que se muestra a continuación y ya está listo el sistema para enviar mensajes:

Gráfico 4.4: Vista del browser de la interfaz principal del módulo de envío masivo.

- Para enviar un mensaje primero seleccionamos el grupo al que deseamos hacer llegar el SMS para esto desplegamos en combo box "GRUPO" y seleccionamos el ítem de la instancia deseada.

- Se podrá observar como automáticamente se carga en el panel de la derecha los números y a que operadora pertenecen (esos

son los números de teléfono celular de las personas de la instancia seleccionada en el combo box “Grupo”).

- Seguidamente se debe escribir el texto que ira en el SMS en el campo de área de texto ubicado a la izquierda que esta con título de “MENSAJE”.

- Terminada la redacción en el área de texto dar clic en el botón “ENVIAR” y empezara el proceso de envío automáticamente a todos los destinatarios.

5.5.2 MANUAL DE USUARIO DEL MÓDULO DE RECEPCIÓN

El sistema es fácil de explicar, no cuenta con largos procedimientos de configuración ni gran cantidad de botones u objetos de control que disparen eventos que desentiendan al usuario. Incluso una persona que no tenga conocimientos de sistemas puede administrar esta aplicación. A continuación una descripción de como operar este software:

- Para abrir el programa hacer clic en el icono:

Gráfico 4.5: Icono del módulo de recepción.

- Se abrirá la siguiente pantalla:

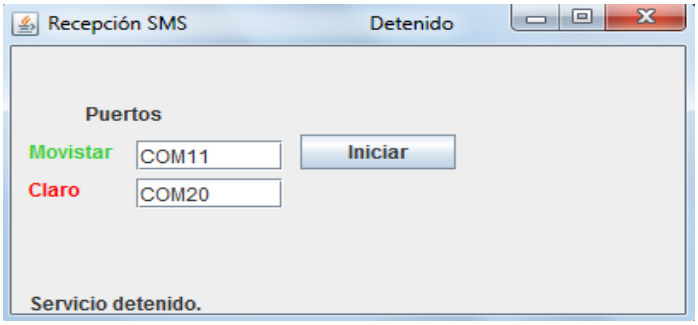


Gráfico 4.6: Ventana de inicio y configuración de puertos del módulo de recepción.

- En la ventana del grafico 4.6 llenar los campos con el nombre del puerto COM que se haya abierto cuando se conectaron los módems al computador (para saber cómo se llama el puerto que se abrió al conectar el modem se puede utilizar un software de monitoreo de puertos o ver en Inicio > Panel de Control > Sistema > Administrador de Dispositivos para el sistema operativo Windows).

- Una vez llenado los puertos, dar clic en el botón "Iniciar", entonces saldrá la siguiente ventana:

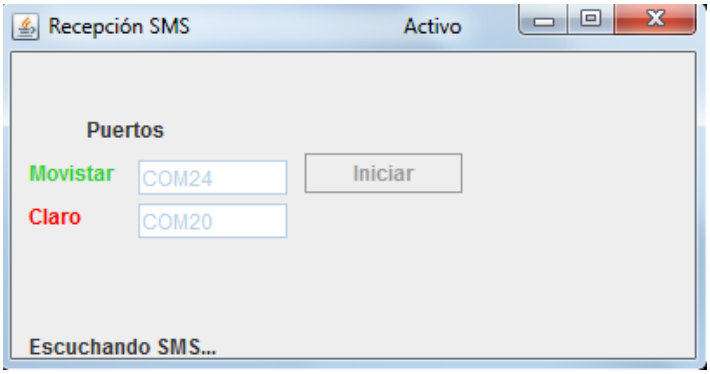


Gráfico 4.7: Ventana del módulo de recepción, iniciado el servicio.

- Una vez hecho esto nos podemos fijar que cambio el texto en la barra de título de "Recepción SMS Detenido" a "Recepción SMS Activo" y en la parte baja de la ventana cerca del borde inferior en la izquierda también cambio el texto de una etiqueta de "Servicio detenido" a "Escuchando SMS…" , esto nos indica que el módulo de recepción ya está en funcionamiento.

CONCLUSIONES

El desarrollo de este sistema de software resulto ser “eco-amigable” ya que elimino la necesidad de imprimir en papel documentos de no tanta trascendencia a nivel público ya que son de manejo interno del establecimiento, en general elimino el manejo de documentos físicos como certificados de notas, pagos pendientes de los estudiantes, convocatorias a reuniones, recordatorios a profesores, padres de familia, administrativos y/o distintos grupos que forman parte de la comunidad del Técnico Salesiano; también implica un ahorro de recursos económicos y de tiempo en la movilidad para la entrega de los documentos antes mencionados.

Específicamente para el módulo de envío masivo el tiempo total de envío de cierta cantidad de mensajes SMS, variará dependiendo de qué tan saturado esté el sistema de red celular por los usuarios de la operadora que esté en uso, es decir este tiempo no depende de la aplicación desarrollada sino del tráfico en la red de las operadoras.

El proyecto se realizó con éxito alcanzándose todos los objetivos planteados tanto generales como específicos, éste proyecto será sin lugar a dudas muy beneficioso para la comunidad de la Unidad Educativa Técnico Salesiano ya que permitirá un ahorro de tiempo y esfuerzo cuando exista la necesidad de enviar un comunicado a cualquiera de los grupos de personas que la misma maneja.

RECOMENDACIONES

Para el módulo de recepción se puede incrementar los servicios y no limitarse tan solo a solicitud de notas o pensiones pendientes, es decir este módulo es sumamente escalable, como ejemplo en un futuro se puede incrementar los servicios de descarga de horarios de exámenes, horario de clases, estados de solicitudes o trámites internos en general.

Dependiendo de qué tanta aceptación tenga el sistema y en específico el módulo de envió masivo, las reuniones de entrega de certificados de notas o libretas podrían ser sustituidas por el envío de mensajes a los padres de un SMS con el contenido de las mismas.

Es posible que en un futuro se incrementen los estudiantes que utilicen como operadora celular de "CNT Alegro" por lo tanto el sistema también tendría que expandirse para aceptar módems de esta operadora.

Para el buen manejo del sistema es muy importante que el administrador del mismo esté al tanto de la disponibilidad de mensajes sobrantes para poder ser enviados, cabe recalcar que el sistema notificara cuando el paquete de mensajes este próximo a terminarse, pero esta alerta debe ser considerada por el administrador para así evitar contratiempos.

Por último es muy importante que el administrador o el encargado de ingresar una nueva persona a los diferentes grupos constate de que su número de celular pertenezca a una de las diferentes operadoras de celular, pudiendo ser MOVISTAR o CLARO, esto será provechoso para el sistema ya que permitirá trabajar con un sistema de paquetería de mensajes para las diferentes operadoras antes mencionadas.

BIBLIOGRAFÍA

1. **ALEGSA.** *sitio web de alegsa.com.ar.* 2008. [Recuperado el: 11 de Febrero de 2012.] http://www.alegsa.com.ar/Dic/sms.php.

2. **DEVELOPERSHOME.** developershome.com. *sitio web de developershome.* 2008. [Recuperado el: 11 de Abril de 2012.] http://www.developershome.com/sms/resultCodes.asp.

3. **NAVEGANTE2.ELMUNDO.ES.** *sitio web de elmundo.es.* 2009. [Recuperado el: 12 de Abril de 2012.] http://navegante2.elmundo.es/navegante/2009/05/04/gadgetoblog/1241407484.html.

4. **BONILLA, David.** Blog Personal 2009. [Recuperado el: 15 de 03 de 2012.] http://sixservix.com/blog/david/2009/08/21/gestion-de-memoria-en-java/.

5. **APACHE.** *sitio web de Apache Tomcat 7.* 2012. [Recuperado el: 5 de Julio de 2012.] http://tomcat.apache.org/tomcat-7.0-doc/introduction.html.

6. **WIKIPEDIA.** es.wikipedia.org. *sitio web de wikipedia.org.* 2012. [Recuperado el: 2 de Julio de 2012.] http://es.wikipedia.org/wiki/Windows_XP.

7. **NEGMA CIRCUIT.** *sitio web de Negma Circuit.* 2007. [Recuperado el: 20 de Julio de 2012.] http://circuitnegma.wordpress.com/2007/02/07/how-to-install-the-java-communications-api-in-a-windows-environment/.

8. **LONDOÑO ABAD, Jorge Hernan.** Ingeniería de Software. 2005. [Recuperado el: 5 de Julio de 2012.] http://ing-sw.blogspot.com/2005/04/tipos-de-pruebas-de-software.html.

9. **ORACLE CORPORATION.** *sitio web de java.com.* 2012. [Recuperado el: 20 de Junio de 2012.] http://www.java.com/es/download/help/windows_manual_download.xml.

10. **RODRÍGUEZ, Alex y SAGÁSTEGUI, Walter.** *aprenderaprogramar.com.* 2012. [Recuperado el: 20 de Junio de 2012.] http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=389:configurar-java-en-windows-variables-de-entorno-javahome-y-path-cu00610b&catid=68:curso-aprender-programacion-java-desde-cero&Itemid=188.

11. **VITTALITO.** *sitio web de Vittalito.* 2011. [Recuperado el: 23 de Marzo de 2012.] http://vittalito.blogspot.com/2011/03/instalar-apache-tomcat7.html.

ANEXOS

ANEXO 1 INSTALACIÓN DE JAVA

La instalación de Java es muy sencillo y para explicar esto que mejor que un pequeño tutorial de la sitio WEB oficial de Java entonces aquí un están los pasos según (9)
El procedimiento de instalación de Java consiste, a grandes rasgos, en:

1. Descarga e instalación
2. Probar la instalación
3. Requisitos del sistema Windows

El tamaño normal de la descarga es de 10 MB, que es la descarga mínima. El tamaño puede aumentar si se seleccionan funciones adicionales.

Nota: el usuario debe tener permisos de administrador para descargar e instalar Java en el sistema.

4. Descarga e instalación

Se recomienda, antes de proceder con la instalación en línea, desactivar el cortafuegos de Internet. En algunos casos, la configuración del cortafuegos predeterminado se establece para rechazar todas las instalaciones automáticas o en línea, como la instalación en línea de Java. Si el cortafuegos no se configura correctamente, podría impedir la operación de descarga/instalación de Java en determinadas circunstancias. Consulte las instrucciones del manual específico del cortafuegos de Internet para desactivarlo.

- Haga clic en Windows en línea.
- Aparecerá el cuadro de diálogo Descarga de archivos y le pedirá que ejecute o guarde el archivo descargado

  - Para ejecutar el instalador, haga clic en Ejecutar.
  - Para guardar el archivo y ejecutarlo más tarde, haga clic en Guardar. Seleccione la ubicación de la carpeta y guarde el archivo en el sistema local.
    Sugerencia: guarde el archivo en una ubicación conocida de su equipo; por ejemplo, en el escritorio.

Haga doble clic en el archivo guardado para iniciar el proceso de instalación.

- Se iniciará el proceso de instalación. Haga clic en el botón Instalar para aceptar los términos de la licencia y continuar con la instalación.

- Oracle colabora con empresas que ofrecen distintos productos. Es posible que el instalador le ofrezca la opción de instalar estos programas como parte de la instalación de Java. Una vez seleccionados los programas que desee, haga clic en el botón Siguiente para proseguir con el proceso de instalación.
- Se abrirán varios cuadros de diálogo con información para completar las últimas etapas del proceso de instalación; haga clic en Cerrar en el último cuadro de diálogo. Con esta acción se completará el proceso de instalación de Java.

NOTA: quizá deba reiniciar (cerrar y abrir) su navegador para habilitar la instalación de Java en su navegador.

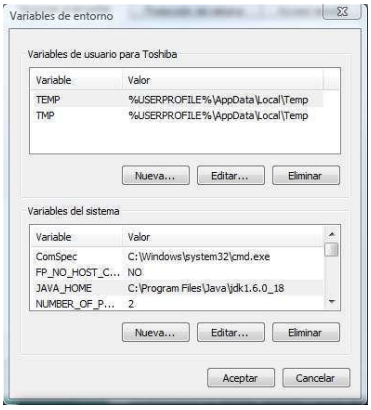
Java Home

De la misma manera en que en la instalación de Java para explicar la instalación de la variable de entorno JAVA_HOME nos apoyaremos en la explicación ya existente en un sitio WEB publicada en internet; Esto se indica en dos pasos, tomado de (10)

ANEXO 2 CONFIGURAR LA VARIABLE DE ENTORNO JAVA_HOME

JAVA_HOME, es una variable de entorno del sistema que informa al sistema operativo sobre la ruta donde se encuentra instalado Java. Seguiremos la siguiente secuencia de pasos para configurar esta variable:

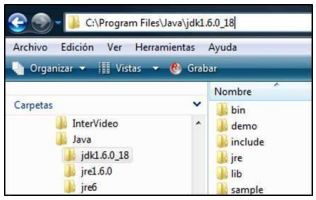
Abrimos el explorador de Windows o pulsamos sobre “Mi Pc”. Pulsamos sobre Equipo y con botón derecho del ratón o buscando el icono -> Propiedades -> Configuración avanzada / Cambiar configuración -> Opciones avanzadas -> Variables de entorno -> Nueva (Variables del sistema).

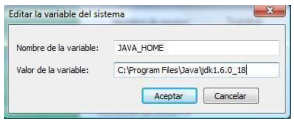


Escribiremos en las cajas de texto que se muestran lo siguiente:

Nombre de variable: JAVA_HOME

Valor de variable: escribiremos aquí la ruta en que se haya instalado Java. Puedes consultarla en el propio explorador de Windows buscando la carpeta en que se ha instalado Java, que normalmente será del tipo C:\Program Files\Java\jdk1.6.0_26 ó C:\Program Files (x86)\Java\jdk1.6.0_26. Fíjate en la barra superior donde aparece la ruta y cópiala tal y como aparece ahí.





ANEXO 3 CONFIGURAR LA VARIABLE DE ENTORNO PATH

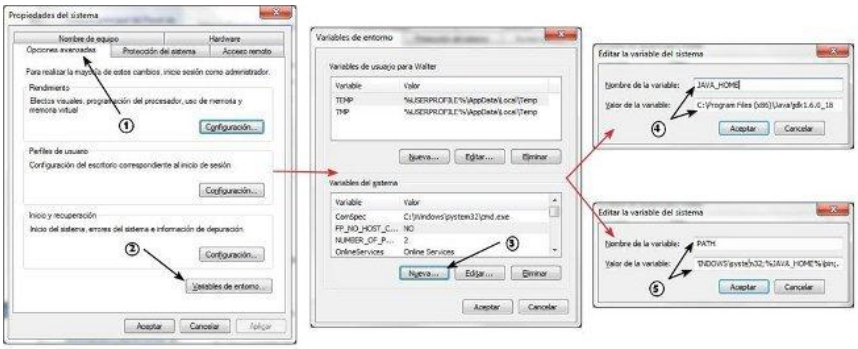
PATH es una variable de entorno del sistema que informa al sistema operativo sobre la ruta de distintos directorios esenciales para el funcionamiento del ordenador. Vamos a añadir al contenido de la variable PATH el lugar donde se encuentran los ficheros ejecutables de Java necesarios para su ejecución, como el compilador (javac.exe) y el intérprete (java.exe). Seguiremos la siguiente secuencia de pasos para configurar esta variable.

Abrimos el explorador de Windows o pulsamos sobre “Mi Pc”. Pulsamos sobre Equipo -> Propiedades y con botón derecho del ratón o buscando el icono -> Configuración avanzada / Cambiar configuración -> Opciones avanzadas -> Variables de entorno -> Editar (Variables del sistema). Luego al final del contenido que ya exista, añadiremos un punto y coma y el texto %JAVA_HOME%\bin. No deben quedar espacios intermedios. Nos quedará similar a esto:

Nombre de variable: PATH

Valor de variable: C:\WINDOWS;C:\WINDOWS\system32;%JAVA_HOME%\bin

A modo de resumen de todo el proceso de configuración: hemos creado una variable de entorno llamada JAVA_HOME y hemos añadido una expresión a la variable PATH.



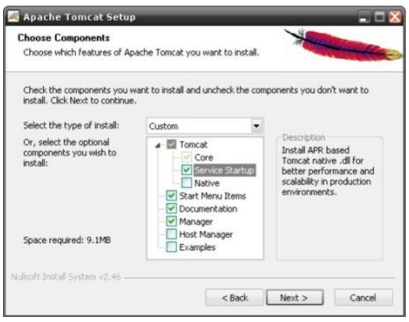
Asegúrate de que todo ha ido bien cerrando todas las ventanas y entrando a “Variables de entorno” para comprobar que aparece todo como hemos indicado. Si es así, ya tenemos Java instalado y configurado en nuestro ordenador.

ANEXO 4 INSTALACIÓN DE APACHE TOMCAT 7

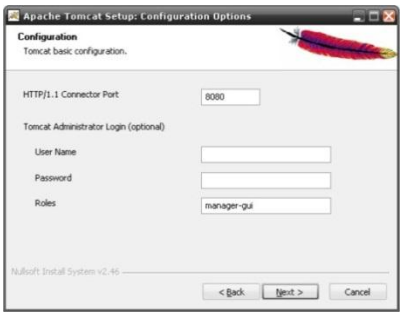
Para la explicación de cómo instalar Apache Tomcat nos apoyaremos en un blog publicado en internet que nos pareció fácil de entender en (11), a continuación detallare los pasos a seguir para instalar Apache Tomcat 7 (servidor de aplicaciones Web). Esta instalación específicamente será hecha en Windows XP (también es válida para Windows Vista, 7).

Primero: Descargar el instalador 32-bit/64-bit Windows Service Installer (pgp, md5) Apache Tomcat 7.0.11 .

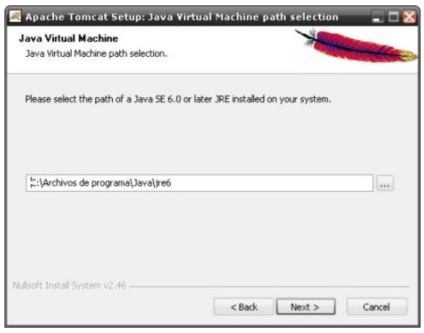
Segundo: Procedemos a realizar la instalación. habilitamos la opción *Service Startup* asi nuestro servidor se iniciara junto con nuestro sistema operativo.



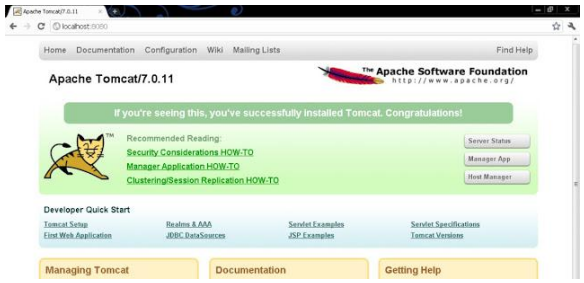
en esta parte podemos configurar el puerto de conexión así como el nombre de usuario y su respectiva contraseña (Datos de acceso al servidor), para fines del caso dejaremos tal y cual aparece en la siguiente imagen.



Tercero: Indicaremos la ruta donde se encuentra instalado el JRE (Java Runtime Environment) el cual previamente habremos instalado.



Después de realizar estos pasos finalizamos la instalación del servidor y procedemos a realizar una pruebas para ello abriremos nuestro navegador de Internet, ingresamos http://localhost:8080/



después de esto estaremos listos para realizar algún proyecto web, y para el cual debemos tener en cuenta lo siguiente la ruta de instalación de nuestro servidor esta localizada en

C:\Archivos de programa\Apache Software Foundation\Tomcat 7.0

existe varios carpetas que acontinuación habremos un descripción resumida de cada una de ellas

/bin – arranque, cierre, y otros scripts y ejecutables.
/temp – archivos temporales.
/conf – archivos XML y los correspondientes DTD para la configuración de apache-tomcat el más importante es server.xml.
/logs – archivos de registro (log) de apache-tomcat.
/webapps – directorio que contiene las aplicaciones web, sera aquí donde podremos colocar nuestros proyectos para que pueda ser desplegado por nuestro servidor.
/work – almacenamiento temporal de ficheros y directorios.

Vamos a realizar nuestro clásico "Hola Mundo" para ello nos ubciamos en *C:\Archivos de programa\Apache Software Foundation\Tomcat*

7.0\webapps, dentro de ella crearemos una carpeta en este caso se llamara "appHolaMundo" dentro del cual crearemos un archivo HolaMundo.html dentro del archivo escribiremos lo siguiente:

```
<html>
    <head>
    </head>
    <body>
        Hola Mundo
    </body>
</html>
```

Guardamos y por ultimo accedemos desde nuestro navegador a la siguiente dirección:

Probemos: http://localhost:8080/appHola/HolaMundo.html

ANEXO 5 ENCUESTA RALIZADA A LOS ESTUDIANTES DE LA UNIDAD EDUCATIVA TÉCNICO SALESIANO

Esta encuesta fue realizada por estudiantes de la unidad educativa durante el periodo lectivo actual 2011- 2012, información que nos facilitó el departamento de sistemas de la unidad educativa. La transcribimos a continuación:

Tabulación de Datos.

1. ¿Sabe usted que significa las siglas SMS?

Respuesta	Frecuencia	Porcentaje
Si	25	63%
No	15	37%
Total	40	100%

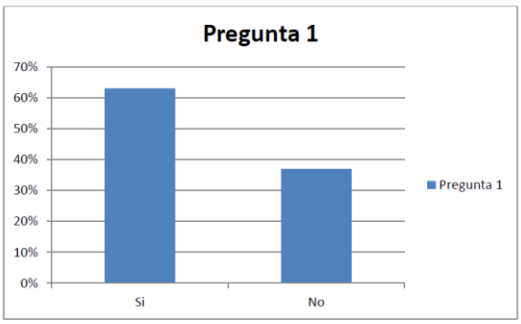


Ilustración 4 Diagrama de Estadística Pregunta 1

2. ¿Con que cree que se relaciona SMS?

Respuesta	Frecuencia	Porcentaje
Mensajería	25	63%
Materia	5	13%
Comida	2	5%
No sabe	8	20%
Total	40	100%

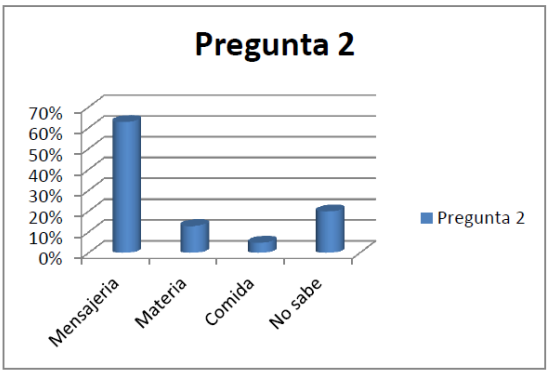


Ilustración 5 Diagrama de Estadística pregunta 2

3. ¿Por qué medio se comunica frecuentemente?

Respuesta	Frecuencia	Porcentaje
Mensaje de texto	12	30%
Chat	20	50%
Llamada	8	20%
Correo	0	0
Total	40	100%

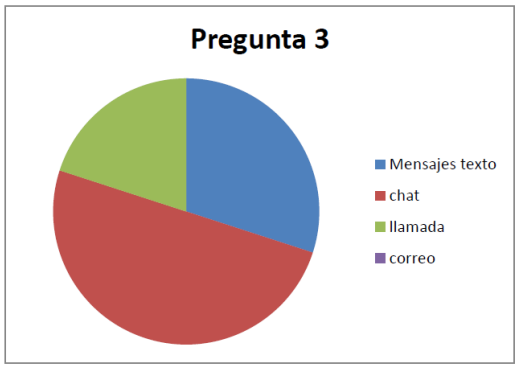


Ilustración 4 Diagrama de estadística pregunta 3

4. ¿En que aparato electrónico cree usted que se utiliza la tecnología SMS?

Respuesta	Frecuencia	Porcentaje
Celular	25	63%
TV	3	7%
Radio	10	25%
DVD	2	5%
Total	40	100%

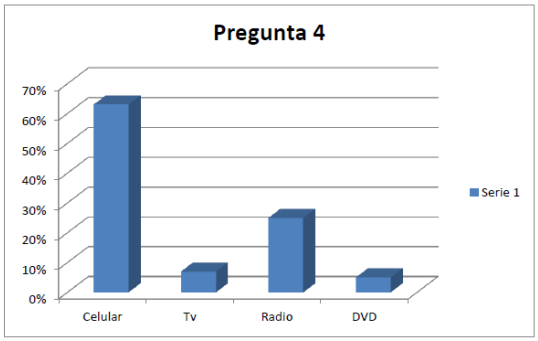


Ilustración 6 Diagrama de Estadística Pregunta 4

5. ¿Conoce alguna empresa en nuestro medio que se dedique a este sistema?

Respuesta	Frecuencia	Porcentaje
Si	10	25%
No	30	75%
Total	40	100%

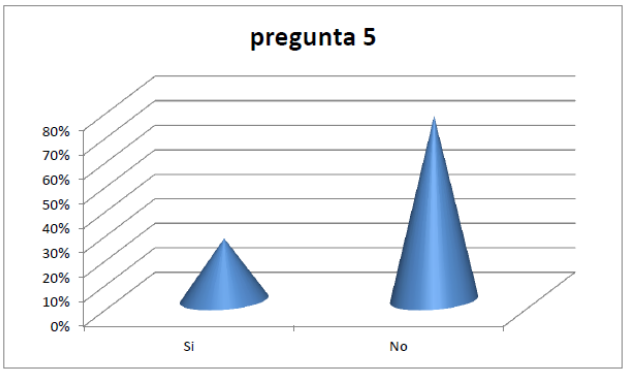


Ilustración 7 Diagrama de Estadística Pregunta 5

6. ¿Cada qué tiempo aproximadamente envía un mensaje de texto?

Respuesta	Frecuencia	Porcentaje
30 min o menos	21	53%
1 hora	17	42%
3 horas	2	5%
No lo utiliza	0	0%
Total	40	100%

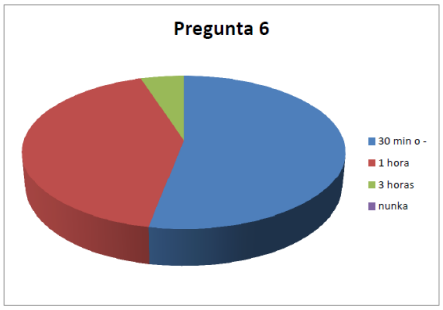


Ilustración 8 Diagrama de Estadistica pregunta 6

7. ¿Cree que sería necesario implementar un sistema que envié directamente a su celular mensajes importantes de cualquier motivo que tenga que ver con su colegio?

Respuesta	Frecuencia	Porcentaje
SI	35	88%
NO	5	12%
Total	40	100%

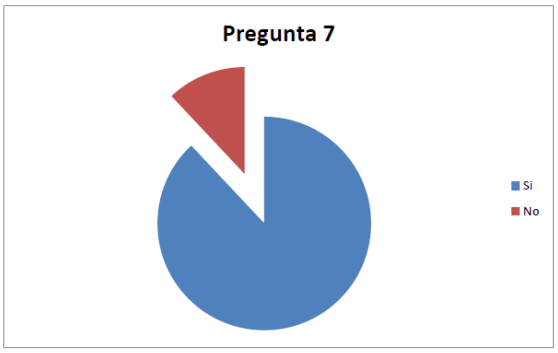


Ilustración 10 Diagrama de Estadistica pregunta 7

8. ¿ Cual piensa que debería ser el costo de este servicio?

Respuesta	Frecuencia	Porcentaje
Muy Caro	14	35%
Caro	11	27%
Medio	12	30%
Gratuito	3	8%
Total	40	100%

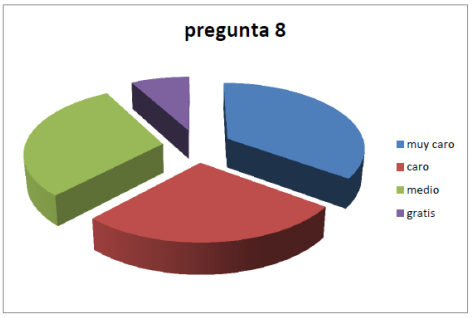


Ilustración 8 Diagrama de Estadística pregunta 8

9. ¿Estaría dispuesto a pagar por este servicio?

Respuesta	Frecuencia	Porcentaje
Si	37	92%
No	3	8%
Total	40	100%

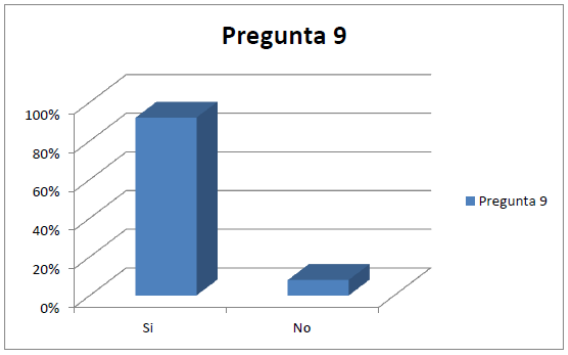


Ilustración 9 Diagrama de Estadística Pregunta 9

10. ¿Que puntaje le pone a este proyecto en una escala del 1 al 10?

Respuesta	Frecuencia	Porcentaje
1 a 3	0	0%
4 a 6	1	3%
7 a 9	20	50%
10	19	47%
Total	40	100%

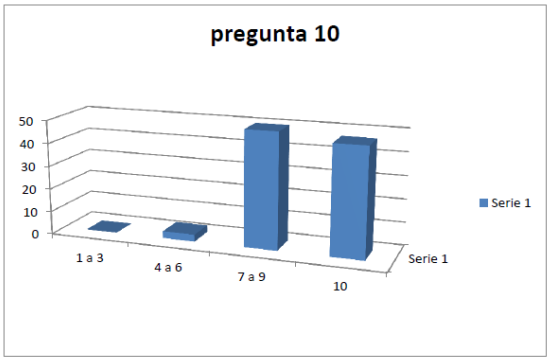


Ilustración 10 Diagrama de Estadística pregunta 10

ANEXO 6 ERRORES DE COMANDOS AT

El contenido que se muestra a continuación fue tomado de la página: http://www.developershome.com/sms/resultCodes2.asp recuperado el 23-07-2012

The +CMS ERROR final result code is returned when a message service failure occurs. An error code is provided for programmers to check what causes the message service failure. The +CMS ERROR final result code is specific to SMS AT commands, i.e. the +CMS ERROR final result code will only be outputted by AT commands that are used to perform tasks related to SMS messaging. Below are the SMS AT commands that may output the final result code +CMS ERROR:

- +CMGC (command name in text: Send Command)
- +CMGD (command name in text: Delete Message)
- +CMGL (command name in text: List Messages)
- +CMGR (command name in text: Read Message)
- +CMGS (command name in text: Sending Message)
- +CMGW (command name in text: Write Message to Memory)
- +CMSS (command name in text: Send Message from Storage)
- +CNMA (command name in text: New Message Acknowledgement to ME/TA)
- +CNMI (command name in text: New Message Indications to TE)
- +CPMS (command name in text: Preferred Message Storage)
- +CRES (command name in text: Restore Settings)
- +CSAS (command name in text: Save Settings)
- +CSMS (command name in text: Select Message Service)

The syntax of the +CMS ERROR final result code is:

<CR><LF>+CMS ERROR: *error_code<CR><LF>*

Just as the final result codes OK and ERROR, the +CMS ERROR final result code always starts and ends with a carriage return character and a linefeed character. *error_code* is an integer that is associated to a certain error. A list of some error codes and their meanings can be found in "Table of +CMS Error Codes and Their Meanings".

As mentioned earlier, after the execution of a command line, only one final result code is returned. Hence, when an error occurs, you will not find both +CMS ERROR and ERROR in the command response. For errors related to SMS messaging, the +CMS ERROR final result code is returned. For other errors such as invalid command syntax and unsupported AT command, the ERROR final result code is returned as usual.

Below shows some common causes of +CMS errors:

- A SIM card is not present in the GSM/GPRS modem or mobile phone.
- The SIM card requires a password (e.g. PIN, PIN2, PUK and PUK2) but you have not entered it.
- An invalid memory index is assigned to an AT command.
- The memory of the GSM/GPRS modem, mobile phone or SIM card for storing SMS messages is full.
- The SMSC address is unknown or incorrect.

Following is an example that demonstrates the usage of the +CMS ERROR result code. Let's say there is only one SMS text message stored on our Nokia 6021 and it is stored in the memory location at index 1. If we enter the command line "AT+CMGR=11" (it means "to read the SMS message at memory index 11"), Nokia 6021 will return a +CMS error:

AT+CMGR=11<*CR*>
<*CR*><*LF*>+CMS ERROR: 321<*CR*><*LF*>

As mentioned earlier, when a terminal program such as HyperTerminal of Microsoft Windows sees a carriage return character, it moves the cursor to the beginning of the current line. When it sees a linefeed character, it moves the cursor to the same position on the next line. Hence, the command line you entered, together with the response returned, will be displayed like this in a terminal program such as HyperTerminal of Microsoft Windows:

AT+CMGR=11
+CMS ERROR: 321

To find out the meaning of the +CMS error code 321, go to "Table of +CMS Error Codes and Their Meanings". From there, we know that the read message operation failed because an invalid memory index was assigned to the AT command +CMGR.

Note that after the occurrence of a +CMS error, the GSM/GPRS modem or mobile phone will not process the remaining AT commands in the command line. Thus, if the command line sent to Nokia 6021 is "AT+CMGR=11;+CGMI" (+CGMI is the AT command for retrieving the manufacturer name of the GSM/GPRS modem or mobile phone), you will get the following result in Windows' HyperTerminal:

AT+CMGR=11;+CGMI
+CMS ERROR: 321

But if the positions of the two AT commands in the command line are exchanged, Nokia 6021 will output the result of the execution of the AT command +CGMI before outputting the +CMS ERROR result code. Below is the result displayed in Windows' HyperTerminal:

AT+CGMI;+CMGR=11
Nokia

+CMS ERROR: 321

The following table lists some of the +CMS error codes and their meanings.

+CMS error code	**Meaning**
300	Mobile equipment (ME) failure. Mobile equipment refers to the mobile device that communicates with the wireless network. Usually it is a mobile phone or GSM/GPRS modem. The SIM card is defined as a separate entity and is not part of mobile equipment.
301	SMS service of mobile equipment (ME) is reserved. See +CMS error code 300 for the meaning of mobile equipment.
302	The operation to be done by the AT command is not allowed.

+CMS error code	Meaning
303	The operation to be done by the AT command is not supported.
304	One or more parameter values assigned to the AT command are invalid. (For PDU mode)
305	One or more parameter values assigned to the AT command are invalid. (For Text mode)
310	There is no SIM card.
311	The SIM card requires a PIN to operate. The AT command +CPIN (command name in text: Enter PIN) can be used to send the PIN to the SIM card.
312	The SIM card requires a PH-SIM PIN to operate. The AT command +CPIN (command name in text: Enter PIN) can be used to send the PH-SIM PIN to the SIM card.
313	SIM card failure.
314	The SIM card is busy.
315	The SIM card is wrong.
316	The SIM card requires a PUK to operate. The AT command +CPIN (command name in text: Enter PIN) can be used to send the PUK to the SIM card.
320	Memory/message storage failure.
321	The memory/message storage index assigned to the AT command is invalid.
322	The memory/message storage is out of space.
330	The SMS center (SMSC) address is unknown.
331	No network service is available.
332	Network timeout occurred.
340	There is no need to send message acknowledgement by the AT

+CMS error code	Meaning
	command +CNMA (command name in text: New Message Acknowledgement to ME/TA).
500	An unknown error occurred.

ANEXO 7 FICHA TÉCNICA DEL MODEM HUAWEI-E173

La siguiente ficha técnica es tomada de http://es.shoppydoo.com/precio-modem-huawei_e173.html recuperada el 23-07-2012.

Características generales	
Brand	Huawei
Device type	Wireless cellular modem
Compatibility	PC, Mac
Dimensions (W x H x D) [cm]	2.6 x 1.3 x 7.1 cm
Weight [gr]	40 gr
Form factor	External
Modem	
Connectivity technology	Wireless
Max transfer rate [Kbps]	7200 Kbps
Max transfer rate (upload) [Kbps]	5760 Kbps
Interface	Hi-Speed USB
Antenna	Internal integrated
Number of antennas	1
Protocol optimization for cell lines	GSM, GPRS, UMTS, EDGE, HSDPA, HSUPA
Cellular band	GSM 850/900/1800/1900, UMTS 850/1900/2100

WCDMA/UMTS frequency	Quad band
Expansión / Conectividad	
Interfaces	1 x Hi-Speed USB - 4 pin USB Type A
Expansion slots	1 (total) / 1 (free) x microSD, 1 (total) / 1 (free) x SIM card
Audio y Video	
DVB-T	No
Parámetros ambientales	
Min operating temperature [°C]	-10 °C
Max operating temperature [°C]	45 °C
Humidity range operating [%]	5 - 95 %