

UNIVERSIDAD TECNICA DEL NORTE



FACULTAD DE INGENIERIA EN CIENCIAS APLICADAS

ESCUELA DE INGENIERIA EN SISTEMAS COMPUTACIONALES

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS COMPUTACIONALES

TEMA:

**SISTEMA INFORMÁTICO PARA LA PROMOCIÓN Y PUBLICIDAD
DE LOS PRODUCTOS Y SERVICIOS DE LA COOPERATIVA
ATUNTAQUI UTILIZANDO BLUETOOTH, APLICATIVO
“SISTEMA INFORMÁTICO CON BLUETOOTH”**

AUTOR:

Manuel Jesús Cepeda Arteaga

DIRECTOR:

Ing. Jorge Caraguay Procel, Msc

Ibarra – Ecuador

Diciembre - 2011

CERTIFICACIÓN

El señor egresado Manuel Jesús Cepeda Arteaga, ha trabajado en el desarrollo del proyecto de tesis “Sistema Informático para la Promoción y Publicidad de los Productos y Servicios de la Cooperativa Atuntaqui Utilizando Bluetooth”, previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizándola con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.

Ing. Jorge Caraguay Procel, Msc

DIRECTOR DE TESIS

DEDICATORIA

En primer lugar, gracias a Dios por haberme dado este regalo, que es la capacidad de aprender y crecer, sé que durante toda mi vida ha estado pendiente de mis pasos, sin su bendición no hubiese sido posible.

A mi madre María Cepeda, que es la mayor fuerza para seguir superándome, por haberme apoyado en todo momento, por sus consejos, sus valores, su confianza y todo el amor que siempre me ha brindado, Además de su sacrificio y esmero para que yo pueda culminar con éxito mis estudios.

A mi hermano Fernando Mendoza, por apoyo incondicional y porque siempre he contado con él para todo, y por toda la confianza que siempre nos hemos tenido.

AGRADECIMIENTO

Doy gracias a todas las personas que de forma directa o indirecta me brindaron su ayuda en la elaboración de este trabajo de tesis.

Demuestro mi leal agradecimiento al Ing. Jorge Caraguay Procel, director de tesis y amigo quien siempre estuvo disponible con sus valiosos consejos para que este proyecto se lleve a cabo y a todos mis maestros por su apoyo y motivación para la culminación de mis estudios profesionales.

Además resalto mis agradecimientos al Ing. Guillermo Brucil director del área de marketing de la Cooperativa Atuntaqui, principal artífice para la elaboración de este trabajo que está elaborado para la Cooperativa Atuntaqui.

A la Universidad Técnica del Norte y en especial a la Facultad de Ingeniería en Ciencias Aplicadas y a la Escuela de Ingeniería en Sistemas Computacionales por permitirme ser parte de una generación de triunfadores y gente productiva para nuestro país.

ÍNDICE DE CONTENIDOS

DEDICATORIA	2
AGRADECIMIENTO.....	3
INTRODUCCIÓN	14
2. Antecedentes	15
3. Objetivos	16
4. Alcance	17
5. Beneficios	18
CAPITULO I:.....	19
1. ESTUDIO DE MERCADO	19
1.1. Introducción	19
1.2. Qué es el estudio de Mercado?.....	19
1.3. Definición de Mercado.....	20
1.4. Objetivo del Estudio de Mercado.....	20
1.5. Segmentación del Mercado	21
1.6. Implementación.....	22
1.7. Objetivos del Estudio de Mercado	23
1.8. Levantamiento de la Información	23
1.9. Conclusiones	36
CAPITULO II:	37
2. HERRAMIENTAS	37
2.1. Software Libre.....	37
2.2. Bluetooth.....	37
2.3. API Java Para Bluetooth - BlueCove	48
Entre las principales clases que tenemos en la API BlueCove tenemos:	49
2.4. Tecnología Java.....	49

2.5.	Merapi	50
2.6.	Frameworks RIA	51
2.7.	SQLite	53
2.8.	Metodología	56
CAPITULO III: FASE DE INICIO		57
3. PLAN DE DESARROLLO DE SOFTWARE.....		57
3.1.	Propósito	58
3.2.	Alcance.....	58
3.3.	Vista general del proyecto.....	59
3.4.	Organización del Proyecto	64
3.5.	Plan del Proyecto.....	67
CAPITULO IV: FASE D ELABORACIÓN.....		74
4. ESPECIFICACION DE LOS CASOS DE USO		74
4.1.	Especificación Caso de Uso: Carga de Archivos	74
4.2.	Especificación Caso de Uso: Creación de Eventos (campanías).....	76
4.3.	Especificación Caso de Uso: Asignar Eventos Diarios.....	78
4.4.	Especificación Caso de Uso: Reportes y Estadísticas.....	80
4.5.	Especificación Caso de Uso: Buscar teléfonos celulares.....	82
4.6.	Especificación Caso de Uso: Transferencia de Eventos.....	84
CAPITULO V: FASE DE CONSTRUCCIÓN		86
5. VISTA LÓGICA.....		86
5.1.	Diagrama de Clases.....	86
5.2.	Diagrama Entidad Relación	87
5.3.	Arquitectura del Sistema	88
CAPITULO VI: FASE DE IMPLEMENTACION.....		92
6. ESPECIFICACIONES DE CASOS DE PRUEBA.....		92
6.1.	Especificación de Caso de Prueba: Carga de Archivos.....	92
6.2.	Especificación de Caso de Prueba: Creación de Eventos.....	93

6.3.	Especificación de Caso de Prueba: Asignar Eventos Diarios	94
6.4.	Especificación de Caso de Prueba: Traslencia de Eventos	95
CAPITULO VII:		97
7.	LISTA DE RIESGOS	97
CAPITULO VIII:		98
8.	CONCLUSIONES Y RECOMENDACIONES	98
8.1.	Conclusiones	98
8.2.	Recomendaciones.....	101
9.	GLOSARIO	103
10.	REFERENCIAS	106
ANEXOS.....		108
ANEXO A: Encuesta		109
ANEXO B: Diccionario de Datos		111
B.1.	Nombre de las tablas del Sistema	111
B.2.	Explicación de campos de cada tabla del Sistema.....	111
ANEXO C: Guía de Programación		114
C.1.	Estándares de Programación.....	114
C.2.	Estandarización del Lenguaje de Programación.....	115
ANEXO D: Prototipo de Interfaz de usuario		121
ANEXO E: Manual de Usuario.....		125
E.1.	Instalación de BlueMarking - AIR.....	128
E.2.	Instalación de BlueMarking - Java	129
ANEXO F: Manuel Técnico		141

ÍNDICE DE FIGURAS

Figura1.1. Esquema del Proyecto.....	18
Figura1.2. Población que utiliza celular con Bluetooth.	24
Figura1.3. Desea recibir mensajes.....	25
Figura1.4. Tipo de mensaje que prefiere recibir.	26
Figura1.5. Tipo de Campaña.	27
Figura1.6. Recibe mensajes de ofertas	28
Figura1.7. Empresas que envían mensajes.....	29
Figura1.8. Mensajes recibidos por día.	31
Figura1.9. Edad de la población de ciudad Ibarra.	32
Figura1.10. Nivel de preparación de la población.....	33
Figura1.11. Género de la población de la ciudad de Ibarra.	34
Figura1.12. Actividad de la población de la ciudad de Ibarra.....	35
Figura 2.1. Estructura de una dirección Mac.....	41
Figura 2.2. Piconet.	43
Figura 2.3. Scatternet.	44
Figura 2.4. Arquitectura de Protocolos.	46
Figura 2.5. Datagrama Bluetooth.	48
Figura 3.1. Descripción del Proceso Unificado.....	69
Figura 4.1. UC. Carga de Archivos.....	75
Figura 4.2. UC. Creación de Eventos.....	77
Figura 4.3. UC. Asignar Eventos.	78
Figura 4.4. UC. Reportes y Estadísticas.....	80
Figura 4.5. UC. Buscar teléfonos celulares.	82
Figura 4.6. UC. Traslado de Eventos.	84

Figura 5.1. Diagrama de Clases.	87
Figura 5.2. Diagrama Entidad Relación del sistema BlueMarking.	88
Figura 5.3. Diagrama de Arquitectura.	90
Figura 5.4. Diagrama de Arquitectura Java.	90
Figura 5.5. Diagrama de Arquitectura Adobe AIR.	91
Figura E.1. BlueMarking, ubicación de instalación.	128
Figura E.2. BlueMarking, aplicación instalada.	129
Figura E.3. BlueMarking, ejecutar el archivo BlueMarking.jar.	129
Figura E.4. BlueMarking, pantalla carga de Archivos.	130
Figura E.5. BlueMarking, selección de Archivos.	131
Figura E.6. BlueMarking, tipos de Archivos.	132
Figura E.7. BlueMarking, Agenda dinámica.	132
Figura E.8. BlueMarking, popUp para cada día.	133
Figura E.9. BlueMarking, eventos ingresados para un día.	134
Figura E.10. BlueMarking, asignar eventos para un día.	134
Figura E.11. BlueMarking, incremento de evento a un día.	135
Figura E.12. BlueMarking, asignar eventos en un rango de fechas.	136
Figura E.13. BlueMarking, incremento de evento a un rango de fechas.	136
Figura E.14. BlueMarking, reporte por tipo de Archivos.	137
Figura E.15. BlueMarking, reporte de eventos enviados y no enviados.	137
Figura E.16. BlueMarking, reporte de eventos enviados por mes.	138
Figura E.17. BlueMarking, reporte de eventos enviados por día.	139
Figura E.18. BlueMarking, reporte de eventos enviados agrupados por tipo de Archivo.	139
Figura E.19. BlueMarking, reporte total de eventos enviados.	140
Figura E.20. BlueMarking, exportar los datos de la tabla a Excel.	140
Figura F.1. Términos y condiciones de Adobe AIR.	142
Figura F.2. Selección del Idioma de Adobe Flash Builder.	144
Figura F.3. Términos y condiciones de Adobe Flash Builder.	144

Figura F.4. Seleccionamos la ubicación de Eclipse.	145
Figura F.5. Resumen de las opciones por instalar.	145
Figura F.6. Resumen de la instalación.	146
Figura F.7. Seleccionar la perspectiva Flash.	146
Figura F.8. IDE para el manejo de la base de datos SQLite.....	147
Figura F.9. Diagrama Entidad Relación del sistema BlueMarking.....	148
Figura F.10. Estructura del módulo BlueMarking-AIR.	150
Figura F.11. Estructura del módulo BlueMarking-Java.	151

ÍNDICE DE TABLAS

Tabla 1.1. Resultados población que utiliza celular con Bluetooth.	24
Tabla 1.2. Resultados desea recibir mensajes.	25
Tabla 1.3. Resultados tipo de mensaje que prefiere recibir.....	26
Tabla 1.4. Resultados tipo de campaña.	27
Tabla 1.5. Resultados recibe mensajes de ofertas.	28
Tabla 1.6. Resultados empresas que envían mensajes.	29
Tabla 1.7. Resultados mensajes recibidos por día.....	30
Tabla 1.8. Resultados edad de la población de ciudad Ibarra.	31
Tabla 1.9. Resultados nivel de preparación de la población.	32
Tabla 1.10. Resultados género de la población de la ciudad de Ibarra.....	33
Tabla 1.11. Resultados actividad de la población de la ciudad de Ibarra.....	34
Tabla 1.12. Análisis Oferta Demanda del Estudio de Mercado.	35
Tabla 2.1. Clasificación de dispositivos Bluetooth por potencia y alcance.	38
Tabla 3.1. Roles y Responsabilidades.....	66
Tabla 3.2. Plan de Fases.....	67
Tabla 3.3. Plan de Fases: Hitos.....	69
Tabla 3.4. Roles y Responsabilidades.....	71
Tabla 3.5. Roles y Responsabilidades.....	72
Tabla 4.1. UC. Carga de Archivos.	76
Tabla 4.2. UC. Creación de eventos.....	78
Tabla 4.3. UC. Asignar Eventos.....	79
Tabla 4.4. UC. Reportes y Estadísticas.	82
Tabla 4.5. UC. Asignar Eventos.....	83
Tabla 4.6. UC. Traslado de Eventos.....	85

Tabla 6.1. Lista de Riesgos	98
Tabla B.2.1. Tabla de Archivos.....	112
Tabla B.2.2. Tabla de Dispositivos	112
Tabla B.2.3. Tabla de Eventos	113
Tabla B.2.4. Tabla de Eventos Diarios.....	114
Tabla B.2.5. Tabla de Eventos Enviados.....	114
Tabla C.1. Tabla de tipos de identificadores	117
Tabla C.2. Tabla de tipos de datos primitivos	120
Tabla D.1. Prototipo interfaz, Ingreso de Archivos.	122
Tabla D.2. Prototipo interfaz, Crear Nuevo Evento.	122
Tabla D.3. Prototipo interfaz, Asignar Eventos Diarios.....	124
Tabla D.4. Prototipo interfaz, Reportes y Estadísticas.....	125

RESUMEN

El aporte principal de este proyecto de tesis es el desarrollo de un sistema de Marketing por Bluetooth llamado BlueMarketing, que consiste en una aplicación de escritorio que envía información de publicidad y propaganda a través de Bluetooth a teléfonos celulares que sean compatibles con Bluetooth. Para esto hemos analizado detalladamente la especificación Bluetooth y se hizo un estudio de los productos disponibles en el mercado, el impacto y la aceptación de la población frente a la utilización del Bluetooth como medio para enviar publicidad y propaganda. Además con el conocimiento adquirido se construyó el sistema con dos módulos, el primer módulo lo realizamos con Adobe AIR, que se utilizó para crear una interfaz rica de usuario y el segundo módulo lo realizamos con Java que se utilizó en el manejo de la tecnología Bluetooth.

Este sistema permite a la Cooperativa Atuntaqui, obtener una herramienta informática que facilite la publicidad y propaganda de sus servicios y productos de una manera automática y a un bajo costo.

En este documento se describe con detalle su funcionamiento, al igual que se presenta sus diagramas, características y materiales empleados. Con el sistema desarrollado hicimos pruebas de desempeño y de funcionamiento logrando verificar y analizar el correcto funcionamiento del mismo.

SUMMARY

The main contribution of this thesis project is the development of a system of Marketing for called Bluetooth BlueMarketing that consists on a desk application that it sends information of publicity and propaganda through Bluetooth to mobile devices that are compatible with Bluetooth. For this we have analyzed the specification detailedly Bluetooth and a study of the available products was made in the market, the impact and the population's acceptance in front of the use of the Bluetooth like means to send publicity and propaganda. Also with the acquired knowledge the system was built with two modules, the first module carries out it with Adobe AIR that was used to create user's rich interfaz and the second module we carry out it with Java that was used in the handling of the technology Bluetooth.

This system allows to the “Atuntaqui Cooperative” to obtain a computer tool that facilitates the publicity and propaganda of its services and products in an automatic way and at a low cost.

In this document it is described with detail their operation, the same as it is presented their diagrams, characteristics and materials employees. With the developed system we made acting tests and of operation being able to verify and to analyze the correct operation of the same one.

INTRODUCCIÓN

En la actualidad, la mayor parte de personas llevan consigo teléfonos celulares y demás dispositivos móviles a todas partes, esto debido a la gran demanda de movilidad que requiere la vida diaria junto con la necesidad de mantenerse siempre en contacto. La habilidad de poder enviar información de interés en un lugar específico hace factible la interacción con la gente en instantes donde normalmente no sería posible alcanzar, permitiendo así que dicho contacto sea realmente productivo para ambas partes, tanto para el emisor del mensaje o contenido como para aquel que lo recibe. El poder crear un entorno donde sea posible el envío de información publicitaria a los teléfonos celulares de manera responsable abre nuevas formas de comunicación y negocio entre personas.

Este trabajo cimienta las bases para conocer y aprender como la tecnología inalámbrica Bluetooth¹ permite desarrollar un sistema de marketing de proximidad en puntos dentro de la ciudad de Ibarra y así brindar un nuevo enfoque en la manera de promocionar servicio y desarrollo tecnológico se refiere.

En este documento se presenta el análisis y diseño previo al desarrollo del sistema de marketing con la tecnología Bluetooth, con el fin de poner a disposición esta información a personas que investiguen acerca de aplicaciones de escritorio que funciones en una PC y que se comuniquen a través de una red inalámbrica Bluetooth con teléfonos celulares.

¹ **Bluetooth:** Es una especificación para redes inalámbricas, permite la transmisión de voz y datos entre distintos dispositivos mediante una radiofrecuencia segura (2,4 GHz).

2. Antecedentes

La motivación para desarrollar un sistema informático en el cual sea posible aplicar tecnología inalámbrica Bluetooth y aprovechar las ventajas que tiene dicha tecnología ofrece, para solventar las estrategias de marketing actuales, estimulo el interés por desarrollar el sistema de marketing que utilizando la tecnología inalámbrica Bluetooth envía información de publicidad a teléfonos celulares en un determinado rango, con la ayuda de un API² de Java BlueCove construimos toda arquitectura y manejo del Bluetooth y transferencia de Eventos y para la interfaz de usuario utilizamos Adobe AIR.

La Cooperativa de Ahorro y Crédito Atuntaqui Limitada, en su afán de crecer y solidificarse como la pionera en el norte del País, le apuesta al uso de esta tecnología inalámbrica Bluetooth, para la utilización de una herramienta que le permita mostrarse y estar más cerca de los usuarios.

Este tipo de proyectos aporta muchas ventajas en comparación a los métodos de marketing actual, ya que automatizan todos los procesos de marketing y a un costo muy reducido.

² **API:** API (Application Programming Interface) es un conjunto de librerías de código JAVA compilado que se incluyen en la Java Runtime Environment.

3. Objetivos

General

Analizar y desarrollar un sistema de Marketing mediante el empleo de la tecnología inalámbrica Bluetooth, con la capacidad de enviar paquetes de mensajes con contenido publicitario de productos y servicios de la Cooperativa Atuntaqui a los usuarios que tengan un dispositivo móvil con la funcionalidad Bluetooth activa.

Específicos

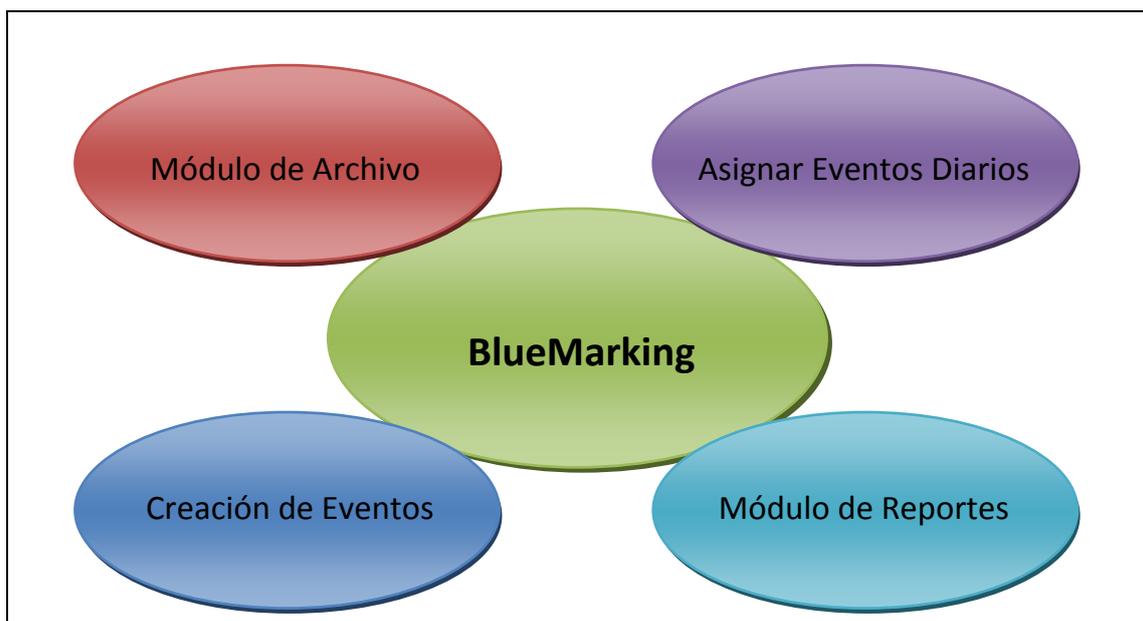
- Estudiar los fundamentos básicos y herramientas necesarias para la creación del presente sistema informático utilizando Bluetooth.
- Estudiar e integrar las herramientas Open Source que están a nuestro alcance para el desarrollo del sistema informático de marketing por Bluetooth.
- Convertir el Sistema de Marketing por Bluetooth en el principal elemento de publicidad y marketing en las empresas.
- Impulsar el manejo de herramientas informáticas al área de marketing de las empresas.
- Simplificar, agilizar y adecuar los procesos de publicidad y propaganda de las empresas.
- Generar reportes, estadísticas sobre la difusión de los productos y servicios de la cooperativa Atuntaqui.

4. Alcance

Desarrollar e implementar una aplicación informática de escritorio que permita enviar mensajes multimedia a teléfonos celulares utilizando la tecnología inalámbrica Bluetooth. Además el ámbito geográfico donde se centrara el estudio de la aplicación será dentro de la ciudad de Ibarra.

El campo de acción del sistema de marketing de proximidad descrito en este trabajo solo se centrará sobre los teléfonos móviles (celulares) y no abarcará al resto de teléfonos celulares empleados hoy en día como laptops, PDAS o agendas electrónicas por cuestiones de simplificación del estudio y ya que los teléfonos celulares cuentan con mayor presencia y son más comunes en nuestra ciudad.

En el desarrollo del sistema BlueMarketing se han identificado los siguientes aspectos que son tomados en cuenta en el diseño y desarrollo del mismo.



Fuente: Propia

Figura1.1. Esquema del Proyecto

5. Beneficios

- El sistema BlueMarketing se basa principalmente en los teléfonos celulares que tienen disponible la tecnología Bluetooth, para el envío y recepción de información multimedia, totalmente gratuita.
- La posibilidad de mensajes de envío son múltiples: imágenes, videos, audio y texto, sin costo de envío. Por lo que las campañas de publicidad suponen una menor inversión y una reducción considerable de los gastos.
- Al ser el envío de forma dinámica, los resultados son rápidos, efectivos y puede estar funcionando la 24 horas del día y 7 días a la semana.
- El presente sistema BlueMarketing simplifica la comunicación con el cliente.
- Se evita el uso indiscriminado del papel y con ello contribuimos a la lucha ambiental.
- Se consigue mayor impacto en cliente, ya que solo acepta el mensaje aquel que está interesado y por ello, presentará mayor atención al contenido. Esto es algo que no se consigue mediante ningún otro tipo de publicidad.
- Tiene una interfaz agradable y muy elegante para el usuario además de ser fácil de usar.
- Agilita la toma de decisiones, ya que con una simple acción se pueden conocer datos y estadísticas de los eventos y tipos de archivos enviados.

CAPITULO I:

1. ESTUDIO DE MERCADO

1.1. Introducción

En una época de globalización³ y de alta competitividad⁴ de productos y servicios, es necesario estar alerta a las exigencias y expectativas del mercado, para ello es de vital importancia asegurar el éxito de las empresas mediante el uso de técnicas y herramientas que les ayude a crecer en el mercado, una de estas técnicas es llevar a cabo el estudio de mercado.

A través del estudio de mercado, es posible verificar la existencia de mercado para el producto o servicio que pretendemos desarrollar, proporcionando los elementos necesarios para determinar la demanda, así como la aceptación del sistema y a través de un análisis de los competidores, establecer estrategias que nos permita incursionar favorablemente en el mercado.

1.2. Qué es el estudio de Mercado?

El estudio de mercado es un proceso sistemático de recolección y análisis de datos e información acerca de los clientes, competidores y el mercado. Sus usos incluyen ayudar a

³ **globalización:** Es una interdependencia económica creciente entre los países, provocada por el aumento del volumen y la variedad de las transacciones de bienes y servicios.

⁴ **competitividad:** Capacidad de generar la mayor satisfacción de los consumidores al menor precio.

crear un plan de negocios, lanzar un nuevo producto o servicio, mejorar productos o servicios existentes y expandirse a nuevos mercados.^[1]

El estudio de mercado es utilizado para determinar que porción de la población comprara un producto o servicio, así como la aceptación del mismo, basado en variables como el género, la edad, ubicación y nivel de Instrucción. Esta es una herramienta que nos ayudara a ver las cosas racionalmente, a minimizar riesgos y verificar el potencial de del presente sistema.

1.3. Definición de Mercado

Es el lugar donde se lleva a cabo las transacciones económicas, es decir, es el lugar donde concurren demandantes y ofertantes.^[2]

1.4. Objetivo del Estudio de Mercado

El estudio de mercado nos sirve para tener una noción clara de la cantidad de consumidores que habrán de adquirir el producto o servicio que se piensa vender, dentro de un espacio definido, durante un periodo de mediano plazo y a qué precio están dispuestos a obtenerlo.^[3]

Adicionalmente, el estudio de mercado nos ayuda a indicar si las características y especificaciones del sistema corresponden a las que desea utilizar el cliente o si va a tener aceptación en el mercado. Nos dirá igualmente que tipo de clientes son los interesados en el sistema, y el lugar donde va a tener mayor aceptación, lo cual servirá para orientar el desarrollo del sistema.

Finalmente, el estudio de mercado nos dará la información sobre la aceptación del presente sistema en el mercado, que tipo de información prefiere recibir acerca de nuestros

^[1] **Estudio de Mercado, "Emprendedores"**, <http://www.blog-emprendedor.info/que-es-el-estudio-de-mercado>.

^[2] **Mercado, "Proyectos de inversión"**, <http://www.mailxmail.com/curso-proyectos-inversion/estudio-mercado>.

^[3] **Objetivo del Estudio de Mercado, "Proyectos de inversión"**, <http://www.mailxmail.com/curso-proyectos-inversion/estudio-mercado>.

productos y servicios y si existe alguna empresa que ofrece los mismos servicios que nuestros.

1.5. Segmentación del Mercado

Es un proceso que consiste en dividir el mercado total de un bien o servicio en varios grupos e internamente homogéneos⁵ (compuestos por clientes con características similares), y luego, seleccionar de entre dichos mercados resultantes, el mercado o los mercados que sean más atractivos para incursionar.^[4]

La razón de segmentar el mercado, es que el mercado total que existe para nuestro sistema es tan amplio y variado, que no es rentable adaptar nuestro sistema para cada tipo de cliente que pueda existir en dicho mercado, lo más razonable es identificar tipos de clientes con características similares, seleccionar el grupo más atractivo para el presente sistema, y enfocarnos sólo en él, especializarnos en él, y dirigir nuestro producto o servicio sólo hacia él, logrando así una mayor efectividad.

Por tanto, nuestros clientes objetivos se pueden agrupar en los siguientes tipos:

- El primer grupo de clientes serán los que se podrían considerar demandantes del sistema de Marketing por Bluetooth, como por ejemplo: Hoteles, Edificios públicos, Aeropuertos, estaciones, Centros Comerciales, etc.
- Otro grupo lo constituyen las empresas y particulares interesados en instalar el presente sistema de Marketing por Bluetooth, como por ejemplo: Promotores, Arquitectos, clientes finales, etc.
- Por otro lado estarán grandes y medianas empresas con la necesidad de promocionar sus productos y servicios a todos los posibles clientes.

Nuestros clientes son por lo tanto todo tipo de empresas que deciden renovar o incorporar nuevas tecnologías de marketing, en la publicidad de sus productos y servicios, para poder competir en un mundo empresarial orientado a la mayor satisfacción del cliente.

⁵ **homogéneos:** Es aquello que pertenece o que está relacionado a un mismo género.

^[4] **Segmentación del Mercado, "Marketing",** <http://www.monografias.com/trabajos57/segmentacion-mercado/segmentacion-mercado.shtml>

1.6. Implementación

- **Selección de la muestra a analizar.-** Para seleccionar la muestra a analizar vamos a tomar las 60.000 personas que es el total de usuarios que tiene la Cooperativa Atuntaqui.

Aplicación de la formula Distribución Normal para ($n < 100.000$)

$$n = \frac{N d^2 Z^2}{(N - 1) E^2 + d^2 Z^2}$$

Donde:

n = Tamaño de la muestra, número de unidades a determinarse.

N = Universo o población a estudiarse.

d² = Varianza de la población respecto a las principales características que se van a representar. Es un valor constante que equivale a 0.25, ya que la desviación típica tomada como referencia es = 0.5.

N-1 = Corrección que se usa para muestras mayores a 30 unidades.

E = Limite aceptable de error de muestra que varían entre 0.01 – 0.09 (1% y 9%).

Z = Valor obtenido mediante niveles de confianza o niveles de significancia con el que se va a realizar el tratamiento de la estimaciones. Es un valor constante que si se lo toma en relación al 95% equivale a 1.96.

Calculo:

$$n = \frac{N d^2 Z^2}{(N-1) E^2 + d^2 Z^2}$$

$$n = \frac{(60.000) (0,25) (1,96)^2}{(60.000-1) (0,05)^2 + (0,25) (1,96)^2}$$

$$n = 381,72$$

Análisis de Resultado

Una vez investigado la cantidad del universo y realizado el cálculo respectivo, obtuvimos que el tamaño de la muestra es de 382, por lo tanto este será el número de encuestas que se deben tomar en cuenta para realizar el estudio de mercado.

1.7. Objetivos del Estudio de Mercado

General

Determinar el impacto y la aceptación del Sistema Informático utilizando Bluetooth para la promoción de publicidad de los productos y servicios de la Cooperativa Atuntaqui, en la ciudad de Ibarra.

Específicos

- Identificar el número de personas que tiene un teléfono celular con tecnología Bluetooth.
- Conocer si existe otra empresa que realice marketing por Bluetooth.
- Identificar el tipo de mensaje que desearía recibir el posible cliente.

1.8. Levantamiento de la Información

La presente investigación tiene como finalidad determinar la utilización del servicio Bluetooth para el envío de mensajes con información de promociones y ofertas de la cooperativa Atuntaqui, este estudio fue realizada en la provincia de Imbabura, ciudad

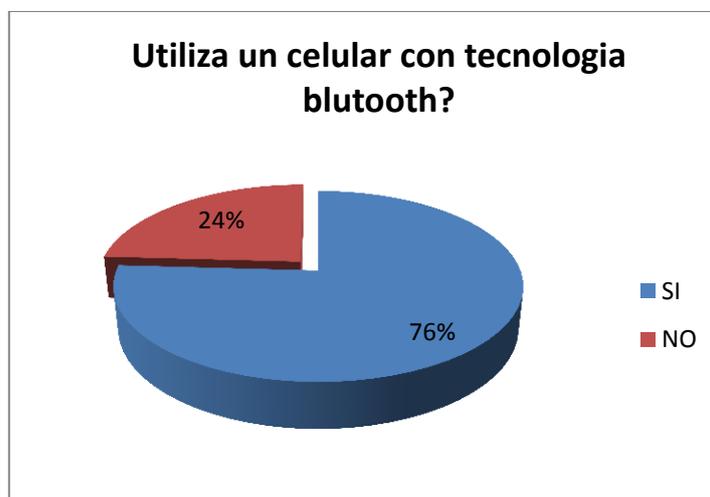
Ibarra entre el 19 de Septiembre del 2011 al 01 de Octubre del 2011, los lugares que se tomaron en cuenta para realizar esta encuesta fueron: La cooperativa Atuntaqui y el centro de la ciudad de Ibarra.

1.- Utiliza usted un teléfono con tecnología Bluetooth?

Variable	Frecuencia	Porcentaje
SI	290	75,90%
NO	92	24,10%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.1. Resultados población que utiliza celular con Bluetooth.



Fuente: Propia

Figura1.2. Población que utiliza celular con Bluetooth.

Como resultado tenemos que en la ciudad de Ibarra, 45540 personas tienen un teléfono celular con tecnología Bluetooth. Lo que significa que este es un potencial mercado para la cooperativa Atuntaqui, para el envío de mensajes vía Bluetooth en la ciudad de Ibarra.

2.- Le gustaría recibir mensajes con información de nuestros productos, a través de Bluetooth?

Variable	Frecuencia	Porcentaje
SI	330	86,40%
NO	52	13,60%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.2. Resultados desea recibir mensajes.



Fuente: Propia

Figura1.3. Desea recibir mensajes.

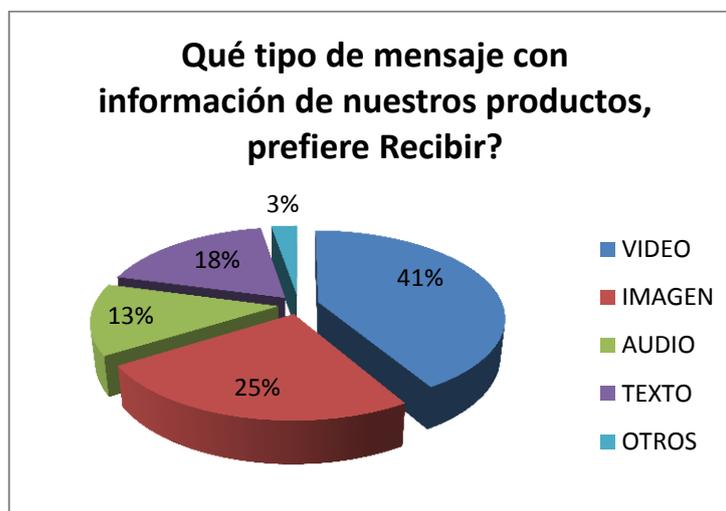
El potencial de la población que está interesada y le gustaría recibir mensajes con información de nuestros productos, a través de Bluetooth es de 51840 personas, en la ciudad de Ibarra.

3.- Qué tipo de mensaje con información de nuestros productos, prefiere recibir?

Variable	Frecuencia	Porcentaje
VIDEO	186	41,52%
IMAGEN	110	24,55%
AUDIO	59	13,17%
TEXTO	81	18,08%
OTROS	12	2,68%
TOTAL	448	100,00%

Fuente: Propia

Tabla 1.3. Resultados tipo de mensaje que prefiere recibir.



Fuente: Propia

Figura 1.4. Tipo de mensaje que prefiere recibir.

En la ciudad de Ibarra, 24912 personas desearían recibir un video como información de nuestros productos, 14730 personas desearían recibir una imagen como información de

nuestros productos, 7902 personas desearían recibir audio como información de nuestros productos, 10848 personas desearían recibir texto como información de nuestros productos, y 1608 personas desearían recibir otro tipo de información de nuestros productos.

4.- Qué tipo de campaña promocional le gustaría recibir?

Variable	Frecuencia	Porcentaje
CREDITO DE VIVIENDA	186	41,52%
CREDITO DE CONSUMO	110	24,55%
MICROCREDITO	59	13,17%
CREDITO COMERCIAL	81	18,08%
TOTAL	448	100,00%

Fuente: Propia

Tabla 1.4. Resultados tipo de campaña.



Fuente: Propia

Figura 1.5. Tipo de Campaña.

En la ciudad de Ibarra, 19950 personas desearían recibir una campaña promocional acerca de Crédito de Vivienda, 15600 personas desearían recibir una campaña promocional acerca

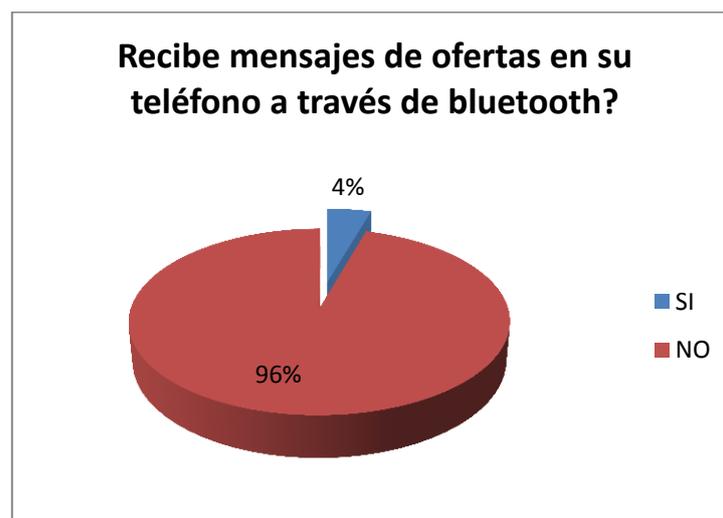
de Crédito de Consumo, 9600 personas desearían recibir una campaña promocional acerca de Microcrédito y 14850 personas desearían recibir una campaña promocional acerca de Crédito Comercial.

5.- Recibe mensajes de ofertas en su teléfono a través de Bluetooth?

Variable	Frecuencia	Porcentaje
SI	17	4,45%
NO	365	95,55%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.5. Resultados recibe mensajes de ofertas.



Fuente: Propia

Figura1.6. Recibe mensajes de ofertas

Como resultado tenemos que en la ciudad de Ibarra, 57330 personas no han recibido mensajes de ofertas en su teléfono celular o través de Bluetooth, lo que significa que en la

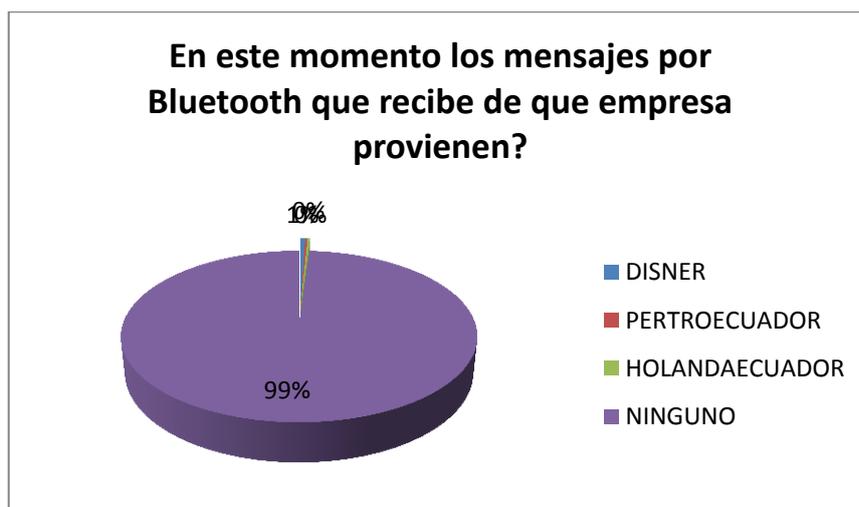
ciudad no se está utilizando esta herramienta para realizar publicidad de productos y servicios. Tan solo 2670 personas han recibido mensajes de publicidad a través de Bluetooth lo que significa, que esta es la oferta que se encontró en el mercado local.

6.- En este momento los mensajes por Bluetooth que recibe de que empresa provienen.

Variable	Frecuencia	Porcentaje
DISNER	2	0,52%
PETROECUADOR	1	0,26%
HOLANDAECUADOR	1	0,26%
NINGUNO	378	98,95%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.6. Resultados empresas que envían mensajes.



Fuente: Propia

Figura1.7. Empresas que envían mensajes.

En la ciudad de Ibarra, 624 personas han recibido algún tipo de mensajes a su teléfono celular con información de ofertas a través de Bluetooth.

7.- En el día cuantos mensajes de ofertas ha recibido a través de Bluetooth?

Variable	Frecuencia	Porcentaje
NINGUNO	378	98,95%
1 msj	2	0,52%
2 msj	2	0,52%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.7. Resultados mensajes recibidos por día.



Fuente: Propia

Figura 1.8. Mensajes recibidos por día.

En la ciudad de Ibarra, 624 persona han recibido entre 1 y 2 mensajes a su teléfono celular con información de ofertas a través de Bluetooth.

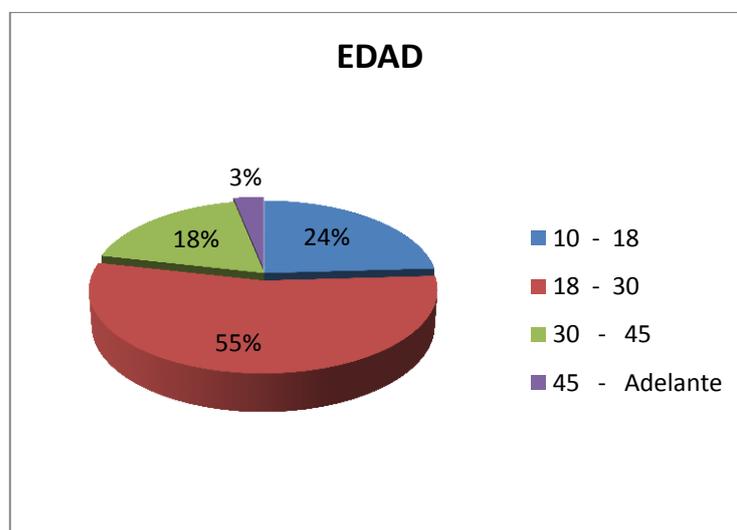
DATOS TÉCNICOS

➤ Edad

Variable	Frecuencia	Porcentaje
10 - 18	92	24,08%
18 - 30	208	54,45%
30 - 45	70	18,32%
45 - Adelante	12	3,14%
TATAL	382	100,00%

Fuente: Propia

Tabla 1.8. Resultados edad de la población de ciudad Ibarra.



Fuente: Propia

Figura 1.9. Edad de la población de ciudad Ibarra.

Como resultado tenemos que hemos encuestado, 14448 personas entre 10 y 18 años, 32670 personas entre 18 y 30 años, 10992 personas entre 30 y 45 años, y 1884 personas mayores a 45 años.

➤ Nivel de Preparación

Variable	Frecuencia	Porcentaje
PRIMARIA	40	10,47%
SECUNDARIA	214	56,02%
SUPERIOR	116	30,37%
CUARTO NIVEL	12	3,14%
TATAL	382	100,00%

Fuente: Propia

Tabla 1.9. Resultados nivel de preparación de la población.

Fuente: Propia

Figura 1.10. Nivel de preparación de la población.

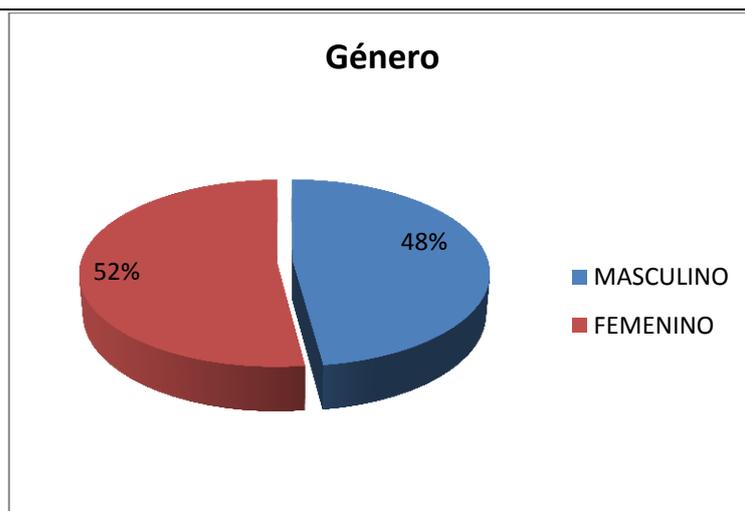
Como resultado tenemos que hemos encuestado a 6282 personas que tienen preparación primaria, 33612 personas que tienen preparación secundaria, 18222 personas que tienen preparación superior, y 1884 que personas tienen preparación de cuarto nivel.

➤ **Género**

Variable	Frecuencia	Porcentaje
MASCULINO	183	47,91%
FEMENINO	199	52,09%
TOTAL	382	100,00%

Fuente: Propia

Tabla 1.10. Resultados género de la población de la ciudad de Ibarra.



Fuente: Propia

Figura 1.11. Género de la población de la ciudad de Ibarra.

Como resultado tenemos que con respecto al género 28746 personas son hombres y 31254 personas son mujeres.

➤ **A qué actividad se dedica?**

Variable	Frecuencia	Porcentaje
OTROS	145	37,88%
COMERCIANTE	64	16,67%
EMPLEADO PÚBLICO	29	7,58%
ESTUDIANTE	116	30,30%
DOCENCIA	29	7,58%
TATAL	382	100,00%

Fuente: Propia

Tabla 1.11. Resultados actividad de la población de la ciudad de Ibarra.



Fuente: Propia

Figura 1.12. Actividad de la población de la ciudad de Ibarra.

Hemos encontrado que con respecto a las actividades más relevantes están primero los estudiantes, segundo los comerciantes, que son las personas que más ocupan un teléfono celular con Bluetooth y otros donde están agrupadas una gran cantidad de actividades.

Análisis oferta demanda

DEMANDA	OFERTA	DEMANDA INSATISFECHA
45540	524	45016

Fuente: Propia

Tabla 1.12. Análisis Oferta Demanda del Estudio de Mercado.

1.9. Conclusiones

A continuación se presenta las conclusiones que se obtuvieron a través de la encuesta, las cuales están basadas en los gráficos y porcentajes.

- 45016 personas de la ciudad de Ibarra, que la cooperativa Atuntaqui puede utilizarlas para el envío de mensajes a través de Bluetooth a sus teléfonos celulares con información de sus productos y servicios.
- Este estudio de mercado justifica el diseño de este sistema para que la cooperativa Atuntaqui pueda utilizarlo en el envío de mensajes publicitarios de sus productos y servicios.
- El costos del envío de mensajes a los usuarios que tengan teléfonos con Bluetooth no tiene ningún valor.

CAPITULO II:

2. HERRAMIENTAS

2.1. Software Libre

Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software: ^[5]

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades. El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tus compañeros.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.

2.2. Bluetooth

Bluetooth es una tecnología de radio de corto alcance, que permite el intercambio de voz y datos entre dispositivos, computadoras portátiles, celulares (móviles), impresoras, cámaras, y otros aparatos electrónicos portátiles mediante un enlace de radiofrecuencia en la banda ISM⁶ de 2,4 GHz. Su máxima velocidad de transmisión de datos es de 1Mbps. El rango de alcance Bluetooth depende de la potencia empleada en la transmisión. La mayor parte de los dispositivos que usan Bluetooth transmiten con una potencia de salida de 0dBm⁷, lo que permite un alcance de unos 10 metros en un ambiente libre de obstáculos.

^[5] **Qué es el Software Libre**, “HispaLinux”, <http://www.hispalinux.es/SoftwareLibre>.

⁶ **ISM**: (Industrial, Scientific and Medical) son bandas reservadas internacionalmente para el uso no comercial de radiofrecuencia electromagnética en el área industrial, científica y médica.

⁷ **dBm**: (Decibel) Unidad de medida utilizada en telecomunicaciones.

Los principales objetivos que se pretenden conseguir con estas normas son:

- Facilitar la comunicación entre equipos móviles y fijos.
- Eliminar cables y conectores entre estos.
- Ofrecemos la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Los dispositivos que con mayor frecuencia utiliza esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal, como PDA⁸, teléfonos móviles, computadoras portátiles, ordenadores personales, impresoras, cámaras digitales, etc.

Las comunicaciones de Bluetooth se realizan por radiofrecuencia de forma que los dispositivo no tienen que estar alineados y pueden incluso estar en habitaciones separadas si la potencia de transmisión lo permite. Estos dispositivos se clasifican como: Clase 1, Clase 2 y Clase 3 en referencia a su potencia de transmisión, siendo totalmente compatibles los dispositivos de una clase con los de otras.

Clase	Potencia máxima permitida(mW)	Potencia máxima permitida(dBm)	Rango (aproximado)
Clase 1	100 mW	20 dBm	100 metros
Clase 2	2.5 mW	4 dBm	10 metros
Clase 3	1 mW	0 dBm	1 metro

Fuente: ^[6]

Tabla 2.1. Clasificación de dispositivos Bluetooth por potencia y alcance.

⁸ **ISM:** (Industrial, Scientific and Medical) son bandas reservadas internacionalmente para el uso no comercial de radiofrecuencia electromagnética en el área industrial, científica y médica.

^[6] **Qué es el Software Libre, "Bluetooth"**, <http://es.wikipedia.org/wiki/Bluetooth>.

Breve Historia

Bluetooth fue creado en los laboratorios Ericsson en 1994 (desde el 2004 se empezaron los estudios de viabilidad) inicialmente para aplicaciones tales como sustituir cables en dispositivos como teléfonos móviles y otros accesorios. Se acordó que cualquier fabricante interesado debería poder acceder libremente a las especificaciones de Bluetooth y así poco a poco se fueron sumando otras compañías. Actualmente participan en el SIG⁹ (Special interest Group) de Bluetooth empresas como ECOM, Ericsson, IBM, Intel, Microsoft, Motorola, Nokia, Toshiba, entre otras más.

En Julio de 1999, el grupo publicó la especificación Bluetooth 1.0, la cual constaba de dos documentos: el núcleo fundamental (core) y el perfil fundamental. El primer documento proporcionaba las especificaciones de diseño, tales como la interfaz de radio, la capa de banda base, el gestor de enlace, el protocolo de descubrimiento de servicios, el nivel de transporte y la interoperabilidad con los diferentes protocolos de comunicaciones; mientras que el perfil fundamental, proporcionaba las directrices para la interoperabilidad de aplicaciones Bluetooth.

A partir de la versión 1.0, que se ratificó en julio de 1999, se han publicado las siguientes versiones:

Versión 1.1:

- Solución erratas de la especificación 1.0.
- Añade el Indicador de calidad de Señal Recibida (RSSI)¹⁰.

Versión 1.2:

⁹ **SIG:** (Special interest Group) es una asociación privada sin ánimos de lucro.

¹⁰ **RSSI:** (Receive Signal Strength indication) se usa para medir el nivel de potencia de las señales recibidas en las redes inalámbricas.

-
- Implementa la técnica de salto en frecuencia, Adaptive Frequency Hopping (AFH)¹¹, para mejorar la resistencia a interferencias.
 - Introduce el tipo de enlace para aplicaciones de audio extended Synchronous Connections (eSCO) que mejora la calidad de voz.
 - Mejoras en el Host Controller Interface (HCI)¹² para una sincronización más rápida de las comunicaciones.

Versión 2.0:

- Nueva versión compatible con la anterior 1.x.
- Incorpora la tecnología Enhanced Data Rate (EDR), que implementa las velocidades de transmisión.
- Reducción del consumo de energía a pesar del incremento de velocidad.

Versión 2.1:

- Simplifica los pasos para crear la conexión entre dispositivos.
- Disminuye el consumo de potencia 5 veces menor.

Versión 3.0:

- Aumenta la velocidad de transferencia.
- La idea es que el nuevo Bluetooth trabaje como WiFi, de tal manera que sea posible lograr mayor velocidad en los smartphones.

Versión 4.0:

- Disminuye drásticamente el consumo de energía
- Permite la incorporación de receptores y transmisores Bluetooth en dispositivos pequeños.
- Se mejora la seguridad con encriptadores AES-128¹³.

ARQUITECTURA BLUETOOTH

¹¹ **AFH:** (Adaptive Frequency hopping) permite reducir interferencias en la banda de 2.4Hz.

¹² **HCI:** (Host Controller Interface) es una interfaz a nivel de registro que permite a un controlador de host para el hardware USB o FireWire comunicarse con un controlador de host en el software.

¹³ **AES-128:** (Advanced Encryption Standard) es una técnica de cifrado de clave simétrica que reemplaza al Estándar de encriptación de Datos.

A la hora de diseñar un sistema de difusión de contenido, es necesario conocer el fundamento de la arquitectura Bluetooth y de los procesos fundamentales que intervienen en el envío de un fichero desde la PC hasta los diferentes dispositivos móviles de los clientes.

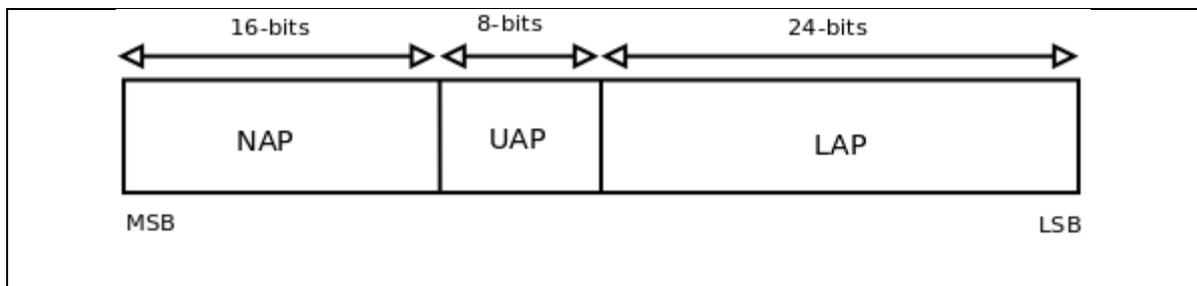
Búsqueda de dispositivos

Todo envío por Bluetooth comienza con una búsqueda inicial de los dispositivos que existen cerca del dispositivo que inicia el envío. Este proceso se llama inquiry y puede ser la operación de mayor duración a la hora de transferir un fichero. Su tiempo medio son 10.24 segundos.

Al igual que en otros protocolos, como TCP/IP, en Bluetooth todos los dispositivos han de tener una dirección que los identifique y distinga de cualquier otro dispositivo. Esta dirección se llama MAC Bluetooth y consta de 6 bloques hexadecimales. Puesto que cada bloque consta de 8 bits, una MAC ocupa $6 \times 8 = 48$ bits. Un ejemplo de MAC sería el siguiente:

11:22:33:44:55:66

Esta dirección mac se divide en 3 partes denominadas NAP (Non significant Address Part), UAP (Upper Address Part) y LAP (Lower Address Parte).



Fuente: Propia

Figura 2.1. Estructura de una dirección Mac

Funcionamiento Inquiry

Pese a que la tecnología Bluetooth lleve varios años en el mercado, hay mucha gente investigando buscando distintas formas de mejorar cada una de sus funciones. Uno de los aspectos fundamentales es el proceso inicial de búsqueda para poder iniciar la comunicación entre dos dispositivos. Este proceso está compuesto de 2 fases:

- **Inquiry:** Se ocupa de encontrar a los demás dispositivos. Su duración suele ser 10.24 segundos aunque podría ser mayor.
- **Paging:** Establece la conexión. Su duración suele ser 7.68 segundos aunque podría ser mayor.

Por otro lado, es necesario comprender que la arquitectura Bluetooth siempre está formada por un maestro y un esclavo, donde el maestro estará realizando la búsqueda teniendo como estado interno el Inquiry state, y el esclavo estará en espera de ser encontrado en el estado inquiry scan state.

Algunas características del modelo maestro/esclavo:

- Un terminal maestro puede comunicarse hasta con 7 esclavos simultáneamente.
- El maestro siempre puede suspender la comunicación con los esclavos.
- Un maestro puede establecer comunicación con un máximo de 256 esclavos.
- Al conjunto maestro/esclavo se le llama piconet.
- Un dispositivo puede ser la vez maestro de un piconet y esclavo de otro piconet.
 - El conjunto resultante se conoce como scatternet.

Piconets

Si un equipo se encuentra dentro del radio de cobertura de otro, estos pueden establecer conexiones entre ellos. Cada dispositivo tiene una dirección única de 48 bits, basada en el estándar IEEE 802.11 para WLAN. En principio solo son necesarias un par de unidades son las mismas características de hardware para establecer un enlace. Dos o más unidades Bluetooth que comparten un mismo canal forman una piconet.



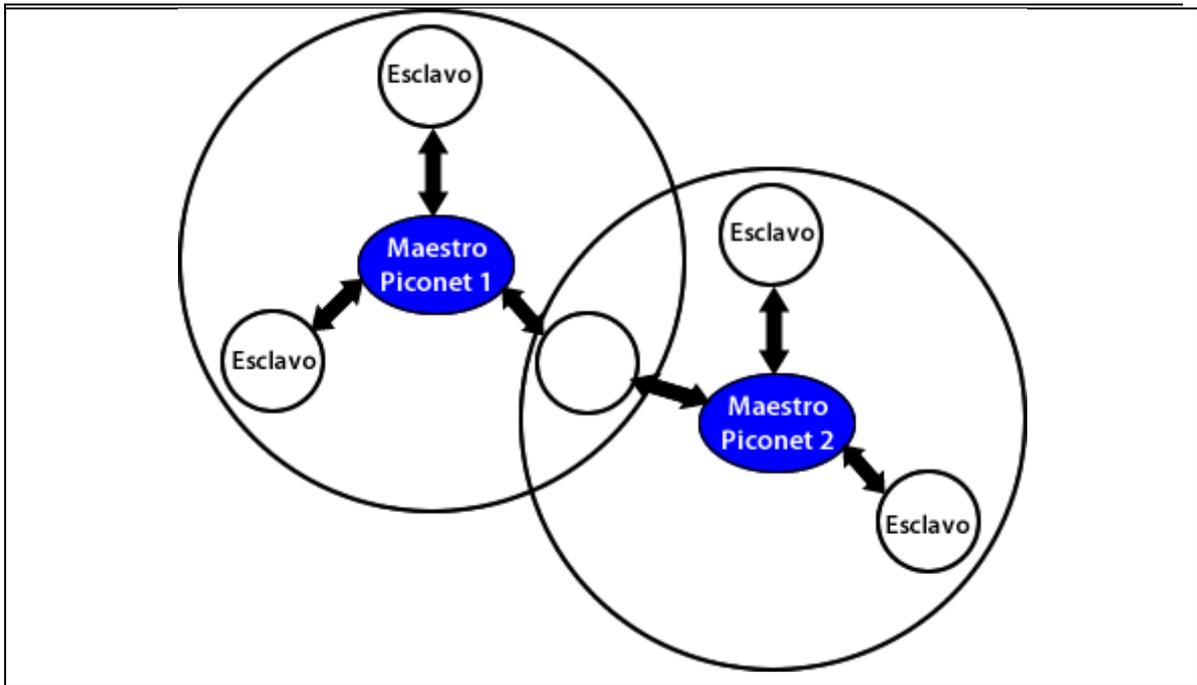


Fuente: Propia

Figura 2.2. Piconet.

Scatternet

Los equipos que comparten un mismo canal sólo pueden utilizar una parte de su capacidad. Aunque los canales tienen un ancho de banda de 1Mbit, cuantos más usuarios se incorporan a la piconet, disminuye la capacidad hasta unos 10kbits más o menos. Para poder solucionar este problema se adoptó una solución de la que nace el concepto de scatternet.



Fuente: Propia

Figura 2.3. Scatternet.

A un grupo de piconets se le llama scatternet. El rendimiento, en conjunto e individualmente de los usuarios de una scatternet es mayor que el que tiene cada usuario cuando participa en un mismo canal de 1Mbit. Además, estadísticamente se obtiene ganancias por multiplexación y rechazo de canales salto. Debido a que individualmente cada piconet tiene un salto de frecuencia diferente, diferentes piconets pueden usar simultáneamente diferentes canales de salto.

Arquitectura de hardware

El hardware que compone el dispositivo Bluetooth está compuesto por dos partes:

- **Un dispositivo de radio**, encargado de modular y transmitir la señal.

-
- **Un controlador digital**, compuesto por una CPU, por un procesador de señales digitales (DSP – Digital Signal Processor) llamado Link Controller (o controlador de enlace) y de las interfaces con el dispositivo anfitrión.

El LC o Link Controller está encargado de hacer el procesamiento de la base y el manejo de los protocolos ARQ¹⁴ y FEC¹⁵ de capa física. Además, se encarga de las funciones de transferencia (tanto asincrónica como sincrónica), codificación de Audio y cifrado de datos.

El CPU del dispositivo se encarga de atender las instrucciones relacionadas con Bluetooth del dispositivo anfitrión, para así simplificar la operación. Para ello, sobre el CPU corre un software denominado Link Manager que tiene las funciones de comunicación con otros dispositivos por medio del protocolo LMP.

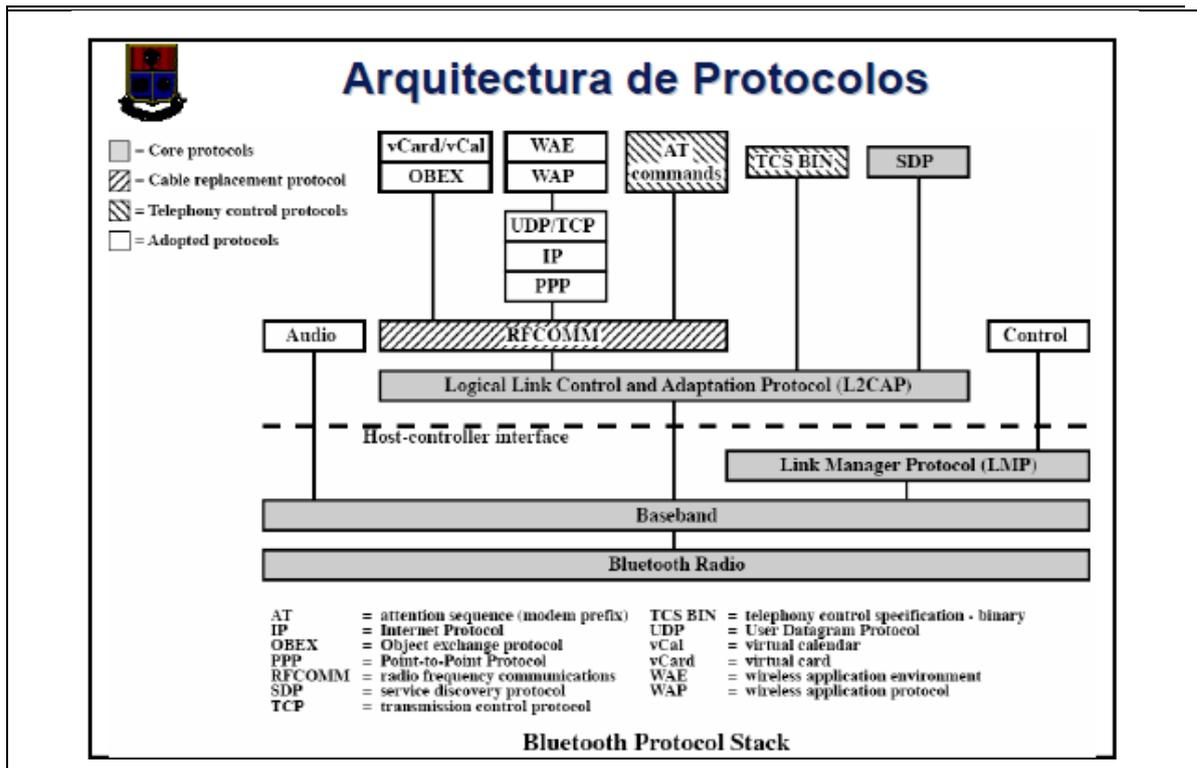
Entre las tareas realizadas por el LC y el Link Manager, destaca las siguientes:

- Envío y Recepción de Datos.
- Empáginamiento y Peticiones.
- Determinación de Conexiones.
- Autenticación.
- Negociación y determinación de tipos de enlace.
- Determinación del tipo de cuerpo de cada paquete.
- Ubicación del dispositivo en modo sniff o hold.

Arquitectura de Protocolos

¹⁴ **ARQ:** (Automatic Repeat-reQuest) son protocolos utilizados para el control de errores en la transferencia de datos, garantizando la integridad de los mismos.

¹⁵ **FEC:** (Forward Error Correction) es el protocolo de corrección de errores usado por el estándar DVB-H, puesto que las interfaces de entrada y salida so IP.



Fuente: [7]

Figura 2.4. Arquitectura de Protocolos.

HCI (Host Controller Interface).- Es parte del stack Bluetooth y utilizando comandos HCI el stack de software del procesador host se comunica con el hardware Bluetooth.

Link Manager Protocol (LMP).- Responsable del establecimiento del enlace entre dispositivos Bluetooth y administración del enlace establecido.

- Sirve para especificar y gestionar el establecimiento y el cierre de la conexión.
- Gestión del consumo de energía.
- Aspectos de seguridad (autenticación y encriptación).
- Gestión de los enlaces sincrónicos y asincrónicos.
- Incluye control y negociación del tamaño de los paquetes de banda base.

La entidad funcional responsable de ejecutar el procesamiento asociado con LMP se denomina Link manager.

[7] **Arquitectura de Protocolos. "2005"**, Escuela Politécnica Nacional, Bluetooth, Iván Bernal ph.D. Quito

Logical Link Control and Adaptation Protocol (L2CAP).- Adapta los protocolos de la capa superior a la capa de banda base.

- Encargada de segmentar y reensamblado de paquetes de gran tamaño.
- Provee servicios sin conexión y orientados a conexión.
- Usualmente L2CAP y las capas superiores se implementan en software.

Service Discovery Protocol (SDP).- Permite a cualquier dispositivo Bluetooth preguntar a otros dispositivos que se encuentra dentro de su alcance:

- Información referente a los dispositivos.
- Descubrimiento de Servicios.
- Características de los servicios (parámetros/atributos).

Esto permite establecer la conexión entre dos o más dispositivos.

RFCOMM.- Presenta un puerto serial virtual, diseñado para que el reemplazo de tecnologías de cable sea tan transparente como sea posible.

TCS - BIN (Telephone Control Specification - Binary).- Protocolo orientado a bit, define la señalización para control de llamadas de voz y datos entre dispositivos Bluetooth.

PPP (Point to Point Protocol).- Para transmitir diagramas IP por un enlace punto – punto.

TCP/UDP/IP.- Protocolos del stack TCP/IP.

OBEX (Object Exchange Protocol).- Protocolo a nivel de sesión, desarrollado para el intercambio de objetos, funcionalidad similar a HTTP pero de forma más simple.

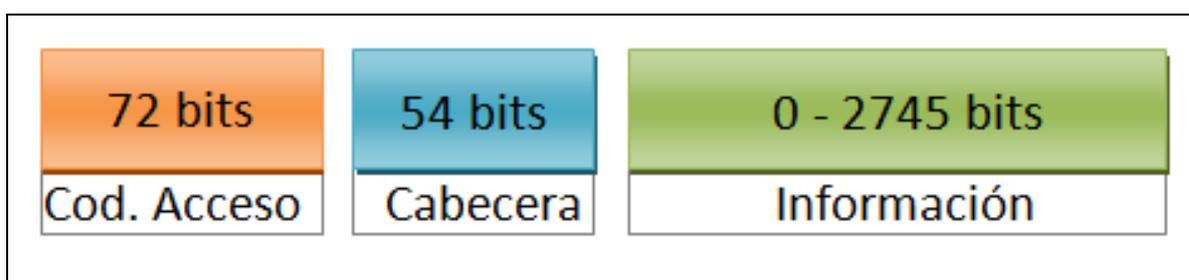
Al igual que HTTP, OBEX se basa en el modelo cliente – servidor y es independiente del mecanismo de transporte.

WAE/WAP (Wireless Application Protocol).- Estándar abierto para proveer a los usuarios móviles de terminales inalámbricos acceso a telefonía y servicios de información.

BaseBand.- Provee una forma de acceso múltiple entre dispositivos localizados en una misma área, pero en diferente piconet.

Datagrama Bluetooth

La información que se intercambia entre dos unidades Bluetooth se realiza mediante un conjunto de slots que forman un paquete de datos. Cada paquete comienza con un código de acceso de 72 bits, que se deriva de la identidad maestra, seguido de un paquete de datos de cabecera de 54 bits. Este contiene importante información de control, como tres bits de acceso de dirección, tipo de paquete, bits de control de flujo, bits para la retransmisión automática de la pregunta, y chequeo de errores de campos de cabecera. La dirección del dispositivo es en forma hexadecimal. Finalmente, el paquete que contiene la información, que puede seguir al de cabecera, tiene una longitud de 0 a 2745 bits.



Fuente: Propia

Figura 2.5. Datagrama Bluetooth.

2.3. API Java Para Bluetooth - BlueCove

BlueCove es una librería en Java que implementa la especificación JSR-82 multiplataforma. Desde la versión 2.1 BlueCove distribuye bajo la licencia de Apache Software.^[8]

La librería soporta los siguientes perfiles bluetooth:

- **SDP.** Perfil de descubrimiento de servicios.
- **RFCOMM.** Emulación del puerto serial.
- **L2CAP.** Enlace lógico y protocolo de adaptación.
- **OBEX.** Protocolo de intercambio de archivos

^[8] **API Bluecove. "2009"**, Universidad de las Palmas de Gran Canaria, Marketing de Proximidad y Herramientas OperSource, RobertoDíaz, Gema Rodríguez.

Entre las principales clases que tenemos en la API BlueCove tenemos:

- **LocalDevice.** Define funciones básicas de Bluetooth sobre el dispositivo local.
- **RemoteDevice.** Define las funciones básicas de Bluetooth sobre el dispositivo remoto.
- **DiscoveryAgent.** Proporciona métodos necesarios para la búsqueda de dispositivos y servicios.
- **DiscoveryListener.** Incluye los diferentes métodos que se ejecutan durante la búsqueda de dispositivos y servicios.
- **UUID.** Define los identificadores de cada uno de los servicios Bluetooth.
- **ClientSession.** Esta interfaz proporciona los métodos para peticiones OBEX, estableciendo las cabeceras de conexión de OBEX.
- **HeaderSet.** Esta interfaz establece las cabeceras de la comunicación a través de OBEX y recibe las repuestas del dispositivo remoto.
- **Operation.** Implementa métodos para la manipulación de operaciones de PUT y GET simple a través de OBEX.^[9]

2.4. Tecnología Java

Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Está desarrollado por la compañía Sun Microsystems con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos un programa en Java podrá funcionar en cualquier ordenador del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux, Apple, etc. Esto lo consigue porque se ha

^[9] API Bluecove. "2009", Universidad de las Palmas de Gran Canaria, Marketing de Proximidad y Herramientas OperSource, RobertoDíaz, Gema Rodríguez.

creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.^[10]

La independencia de plataforma es una de las razones por las que Java es interesante para Internet, ya que muchas personas deben tener acceso con ordenadores distintos. Pero no se queda ahí, Java está desarrollándose incluso para distintos tipos de dispositivos además del ordenador como móviles, agendas y en general para cualquier cosa que se le ocurra a la industria.

2.5. Merapi

Cuando creamos una aplicación Adobe AIR, hay una buena probabilidad de que se han encontrado con problemas con el modelo de seguridad Sandbox¹⁶. El modelo de seguridad Sandbox en AIR esta para evitar que los desarrolladores creen aplicaciones que se puedan extraer información o dañar el equipo del usuario. Estas preocupaciones se justifican y son los que hacen a Adobe AIR en una alternativa segura para la creación de aplicaciones de escritorio. Cuando instalamos una aplicación AIR, siempre se cumple con las indicaciones de las pantallas, y sabemos que el software que está instalando es seguro. El único inconveniente del modelo Sandbox es que este puede ser un limitante a la hora de desarrollar una aplicación, ya que no podemos acceder a unidades externas, líneas de comandos e incluso algunas listas de aplicaciones nativas son bloqueadas por la seguridad del entorno. Como resultado, las aplicaciones suelen ser lanzadas solo con algunos de sus posibles características.^[11]

Con el lanzamiento de AIR 2.0, algunas de los issues de seguridad están resueltas. Ahora con nuestras aplicaciones podemos acceder a los dispositivos de almacenamiento externo y podemos abrir archivos en las aplicaciones nativas, pero todavía hay algunas limitaciones. Aquí es donde Merapi entra en juego.

Merapi es una tecnología Open Source que utilizamos como puente entre las aplicaciones AIR basadas en Flex y programas de escritos Java. El puente Merapi permite la comunicación entre las dos tecnologías a través del uso de los mensajes. Los mensajes pueden contener datos complejos, los protocolos de AMF de serialización de datos se

^[10] **Tecnología Java.** "Java", <http://www.desarrolloweb.com/articulos/497.php>

¹⁶ **Seguridad Sandbox:** Es un mecanismo para ejecutar programas con seguridad y de manera separada.

^[11] **Definición de Merapi.** "Merapi", <http://java.dzone.com/articles/merapi-bridge-between-air-and>

utilizan para la transferencia de datos. La introducción de Java en la ecuación nos permite crear soluciones de vanguardia de software, con Java somos capaces de interactuar con muchos aspectos del sistema de usuario. Mediante el uso de Java, uno es capaz de acceder no solo al código subyacente y las características del sistema operativo del usuario, sino también a numerosas bibliotecas que Java tiene para ofrecer. Como muchos dispositivos externos (como escanear códigos de barra, sensores de temperatura, unidades de GPS, etc.), a menudo viene con API basado en Java, Merapi permite al desarrollador interactuar con los Apis y crear aplicaciones hechas a medida para adaptarse a situaciones específicas que AIR por sí solo no sería capaz de acomodar. ^[12]

A menudo los usuarios tienen preocupación acerca de eludir el modelo de seguridad Sandbox de AIR, después de todo, la seguridad se puso en marcha por una razón. No queremos ver en el mercado inundado de aplicaciones inseguras que podrían ser utilizados por hackers para acceder a la máquina de los usuarios. Sin embargo, esto no va a suceder, sobre todo porque a fin de utilizar Merapi, tanto la aplicación AIR y la aplicación Java deben estar ejecutándose al mismo tiempo. La aplicación Java debe instalarse por separado de la aplicación AIR, y la aplicación de AIR en sí mismo no puede instalar los archivos de Java. Si no instala los archivos de Java y ejecuta la aplicación Java antes de iniciar la aplicación de AIR, la comunicación no puede tener lugar.

2.6. Frameworks RIA

Adobe Flex

Adobe Flex (hasta 2005 **Macromedia Flex**) es un término que agrupa una serie de tecnologías publicadas desde Marzo del 2004 por Macromedia para dar soporte al despliegue y desarrollo de Aplicaciones Enriquecidas de Internet, basadas en su plataforma propietaria Flash. ^[13]

^[12] **Definición de Merapi.** "Merapi", <http://java.dzone.com/articles/merapi-bridge-between-air-and>

^[13] **Definición Adobe Flex.** "Adobe Flex", http://help.adobe.com/en_US/Flex/4.0/UsingSDK/index.html.

Flex fue inicialmente liberado como una aplicación de la J2EE¹⁷ o biblioteca de etiquetas JSP¹⁸ que compilaba el lenguaje de marcas Flex (MXML) y ejecutaba mediante ActionScript aplicaciones Flash (archivos SWF binarios).

El objetivo de Flex es permitirnos a los desarrolladores construir aplicaciones RIAs (Aplicaciones de Internet Ricas), rápidas y fácilmente. En un modelo multi-capas, las aplicaciones Flex son el nivel de presentación.

Flex pone en relieve el desarrollo de Interfaces gráficas de usuario usando un lenguaje XML llamado MXML. En Flex tenemos varios componentes y características que aportan funcionalidades tales como Servicio Web, objetos remotos, arrastrar y soltar, columnas ordenables, gráficas, efectos de animación y otras interacciones simples. El cliente solo carga la aplicación una vez, mejorando así el flujo de datos frente a aplicaciones basadas en HTML (PHP, ASO, JSO, CFMX), las cuales requieren de ejecutar plantillas en el servidor para cada acción. El lenguaje y la estructura de archivos de Flex buscan el desacoplamiento de la lógica y el diseño.

El servidor de Flex también actúa como un Gateway¹⁹ permitiendo al cliente comunicarse con servicios web XML y objetos remotos (tales como Coldfusion CFCs, clases Java y cualquiera que soporte el formato de mensajes de acciones).

Adobe AIR

Adobe AIR es una tecnología que nos permite crear aplicaciones de escritorio (de propósito general) a partir de tecnologías de desarrollo de páginas web, como pueden ser HTML, Ajax o Flash. Con Adobe AIR podemos aprovechar nuestros conocimientos a la hora de crear páginas web para hacer aplicaciones multimedia para el escritorio. Es decir, a partir de un desarrollo de una aplicación web, crear una aplicación general con los mismos contenidos o utilidades del sitio web.^[14]

¹⁷ **J2EE:** (Java 2 Enterprise Edition) Define un estándar para el desarrollo de aplicaciones empresariales multicapas diseñada por Sun Microsystems.

¹⁸ **JSP:** es una tecnología y framework para aplicaciones java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.

¹⁹ **Gateway:** Es una puerta de enlace, un nodo en una red informática que sirve de punto de acceso a otra red.

^[14] **Definición Adobe AIR. "desarrolloweb"**, <http://www.desarrolloweb.com/articulos/que-es-adobe-air.html>

Adobe AIR se distribuye gratuitamente para cualquier persona, tanto los usuarios que deseen instalar y usar programas creados con AIR, como para nosotros, los desarrolladores que creamos aplicaciones de escritorio a partir de sus proyectos web. Por tanto, no es necesario disponer de nada especial que haya que pagar para poder beneficiarse de las capacidades de Adobe AIR. Según el equipo de Adobe, están seguros que esta situación de gratuidad permanecerá por siempre. Eso sí, usando las herramientas de Adobe, como Flash, Dreamweaver o Flex Builder, se simplifica bastante la generación de programas AIR.

Una de las ventajas, adicionales a las que ya he apuntado, sobre AIR es que las aplicaciones generadas son multiplataforma. Es decir, los programas de escritorio de Adobe AIR se podrán utilizar en cualquier ordenador, independientemente del sistema operativo que tenga. Esto se consigue porque cuando descargamos una aplicación con Adobe Air, se comprueba si está instalado el entorno AIR en nuestro ordenador. Si no estaba instalado se descarga el entorno de ejecución (ocupa actualmente unos 10 megas) necesario para que las aplicaciones funcionen en nuestro sistema operativo concreto. Una vez ya el entorno se ha instalado en nuestro sistema no se tiene que volver a descargar ni instalar de nuevo. Con ello, Adobe AIR consigue acercarse al paradigma, "programa una vez y ejecuta donde quieras", a la vez que se hace universal y al alcance de todos los usuarios y plataformas

2.7. SQLite

SQLite es un sistema de gestión de base de datos relacional compatible con ACID, contenida en una pequeña librería de aproximadamente 650Kb, programado en el lenguaje C, de dominio público, totalmente libre creado por D. Richard Hipp.^[15]

A diferencia de los sistemas de gestión de bases de datos cliente-servidor, el motor de SQLite no es un proceso independiente con el que el programa principal se comunica. En lugar de eso, la biblioteca SQLite se enlaza con el programa pasando a ser parte integral del mismo. El programa utiliza la funcionalidad de SQLite a través de llamadas simples a subrutinas y funciones. Esto reduce la latencia en el acceso a la base de datos, debido a que las llamadas a funciones son más eficientes que la comunicación entre procesos. El conjunto de la base de datos (definiciones, tablas, índices y los propios datos), son

^[15] **Definición SQLite.** "SQLite", <http://www.sqlite.org>.

guardados como un solo fichero estándar en la maquina u ordenador. Este diseño simple se logra bloqueando todos los ficheros de base de datos al principio de cada transacción. SQLite permite bases de datos de hasta 2 Terabytes de tamaño, y también permite la inclusión de campos tipo BLOB.

Algunas razones para escoger SQLite como herramienta de desarrollo.

- **Tamaño:** SQLite tiene una pequeña memoria y una única biblioteca es necesaria para acceder a bases de datos, lo que lo hace ideal para aplicaciones de base de datos incorporadas.
- **Rendimiento de base de datos:** SQLite realiza operaciones de manera eficiente y es más rápido que MySQL y PostgreSQL.
- **Portabilidad:** Se ejecuta en muchas plataformas y sus bases de datos pueden ser fácilmente portadas sin ninguna configuración o administración.
- **Estabilidad:** SQLite es compatible con ACID, reunión de cuatro criterios de Atomicidad, Consistencia, Aislamiento y Durabilidad.
- **SQL:** Implementa un gran subconjunto de la ANSI – 92 SQL estándar, incluyendo sub-consultas, generación de usuarios, vistas y trigger.
- **Interfaces:** Cuenta con diferentes interfaces del API, las cuales permiten trabajar con C++, PHP, Perl, Python, Ruby, Tcl, groovy, etc.
- **Costo:** SQLite es de dominio público, y por lo tanto, es libre de utilizar para cualquier propósito sin costo y se puede redistribuir libremente.

Características

- **No posee configuración.-** De la forma en que fue creado y diseñado SQLite, NO necesita ser instalado. NO necesita, reiniciar o apagar un servidor, e incluso configurarlo. Esta cualidad permite que no haya un administrador de base de datos para crear las tablas, vistas, asignar permisos. O bien la adopción de medidas de recuperación de servidor por cada caída del sistema.
- **Portabilidad.-** SQLite se ejecuta en diferentes sistemas operativos, como Windows, Linux, BSD, Mac OS X, Solaris, HPUX, AIX o estar embebido en muchos otros como QNX, VxWorks, Symbian, Palm OS, Windows CE. Se puede

notar que muchos de ellos trabajan a 16, 32 y 64 Bits²⁰. La portabilidad no está dada en sí por el software, sino por la base de datos condensada en un solo fichero, que puede estar situado en cualquier directorio, trayendo como ventaja que la base de datos puede ser fácilmente copiada a algún dispositivo USB o ser enviada vía correo electrónico.

- **Registros de longitud variable.-** Generalmente los motores asignan una cantidad fija de espacio en disco para cada fila en la mayoría de los campos de una determinada tabla. Por ejemplo, tomemos un campo de tipo VARCHAR (255), esto significa que el motor le asignará 255 bytes de espacio fijo en disco, independientemente de la cantidad de información que se almacene en ese campo. En cambio, SQLite aplica su tecnología y realizará todo lo contrario, utilizando para ello la cantidad de espacio en disco necesario para almacenar la información real del campo.

El uso de registros de longitud variable por SQLite, tiene una serie de ventajas, entre ellas el resultado de un pequeño archivo de base de datos y optimización de la velocidad de la misma, puesto que hay menos información desperdiciada que leer y recorrer.

Limitaciones

Así como encontramos algunas ventajas y características realmente asombrosas, también cuenta con algunas limitaciones:

- **Limitaciones en Where:** esta limitación está dada por el soporte para clausuras anidadas.
- **Falta de Clave Foránea:** se hace caso omiso de las claves foráneas, esto quiere decir, cuando se realice la creación de la tabla desde el modo consola, está permitiendo el uso de la clausura, aunque no realizara el chequeo de la misma.
- **Falta de documentación en español:** si bien ya contamos con una comunidad latino americana de SQLite, sería importante encontrar mucha más documentación,

²⁰ **Bits:** Es un dígito de sistema de numeración binaria.

libros, review, etc. como muchos otros motores de bases de datos cuentan hoy en día.

Por último, conocimos un motor de base de datos realmente muy interesante y que es para tenerlo muy en cuenta, de hecho grandes empresas como Adobe, Firefox, Google, McAfee, Toshiba, Sun Microsystem, Symbian, Microsoft, etc. hacen uso de SQLite para el desarrollo de mucho de sus productos, demostrando de esta manera la confianza y el gran rendimiento de la misma.

2.8. Metodología

Las metodologías y estándares utilizados en nuestro desarrollo de software nos proporcionan las guías para poder conocer todo el camino a recorrer desde antes de empezar la implementación, con lo cual nos aseguramos la calidad de nuestro sistema final, así como también el cumplimiento en la entrega del mismo en un tiempo estipulado.

Para la elaboración del presente proyecto de tesis, selecciona la metodología RUP²¹ (*Rational Unified Process*), ya que presenta un proceso de desarrollo iterativo, se maneja el uso de requerimientos, propone una arquitectura basada en componentes, trata el control de cambio de software y la verificación de la calidad del software.

Rational Unified Process (RUP) es un proceso de Ingeniería de Software planteado por Kruchten (2000) cuyo objetivo es producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos.^[17]

RUP²² toma en cuenta las mejores prácticas en el modelo de desarrollo de software en particular las siguientes:

- Desarrollo de software en forma iterativa (repite una acción).
- Manejo de requerimientos.
- Utiliza arquitectura basada en componentes.
- Modela el software visualmente (Modela con Unified Modeling Language, UML)
- Verifica la calidad del software.

²¹ RUP: Rational Unifies Process

^[16] Definición RUP. "RUP", http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational.

²² RUP: Rational Unifies Process

- Controla los cambios.

CAPITULO III: FASE DE INCIO

3. PLAN DE DESARROLLO DE SOFTWARE

Este Plan de Desarrollo del Software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto: Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto ha sido basado en una metodología de Unificación de Procesos con el fin de implantar un esquema inicial de ésta metodología para futuros desarrollos.

El enfoque desarrollo propuesto constituye una configuración del proceso Unificación de Procesos de acuerdo a las características del proyecto, seleccionando las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de la unificación de Procesos.

3.1. Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se describe el enfoque de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

- El jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.
- Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

3.2. Alcance

El Plan de Desarrollo del Software describe el plan global usado para el desarrollo del Sistema Informático utilizando Bluetooth para la promoción de publicidad de los productos y servicios de la Cooperativa Atuntaqui.

Durante el proceso de desarrollo en el artefacto “Visión” se definen las características del producto a desarrollar.

Para el Plan de Desarrollo del Software, me he basado en la definición de requisitos por medio del stakeholder²³ del Área de Marketing de la Cooperativa Atuntaqui, para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”, el cual se utilizará para refinar este documento. Posteriormente, el avance del proyecto y el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

3.3. Vista general del proyecto

La información que a continuación se incluye ha sido extraída de las diferentes reuniones que se han celebrado con el stakeholder de la Cooperativa Atuntaqui desde el inicio del proyecto.

A partir de los procedimientos ya establecidos en el área de Marketing de la Cooperativa Atuntaqui, y como parte del plan de automatización establecido por el Departamento de Informática, se determina la creación del Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth que permita mejorar y automatizar la gestión de las actividades relacionadas con la publicidad de la institución.

El proyecto debe proporcionar una propuesta para el desarrollo de un sistema de marketing de proximidad utilizando la tecnología inalámbrica Bluetooth para la Cooperativa Atuntaqui. Los sub-módulos indicados son los siguientes.

➤ **Archivos multimedia.**

- Cargar de Archivos multimedia a la aplicación.
- Visualización de los archivos cargados en la aplicación.

➤ **Eventos.**

²³ **Stakeholder:** Es un término, para referirnos a quienes pueden afectar o son afectados por las actividades de una empresa.

- Creación de nuevos eventos.
- Comprobar el ingreso correcto de los nuevos eventos.
- Visualización de los archivos multimedia que van a ser parte del nuevo evento.

➤ **Eventos Diarios**

- Asignar los eventos creados a un determinado día, esta va ha ser parte de las campañas.
- Asignación en un rango de fechas de los eventos creados.
- Visualización de los eventos asignado a los días.

➤ **Reportes.**

- Visualización de los datos en los reportes.
- Visualización las gráficas en los reportes.
- Exportar los datos de la tabla de los reportes a EXCEL.

➤ **Procesos Automáticos.**

- Buscar teléfonos celulares con el dispositivo Bluetooth encendido.
- Envió de eventos (campañas) a teléfonos celulares en un rango determinado.

➤ **Suposiciones Y Restricciones.**

Las suposiciones y restricciones respecto del Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth, y se deriva directamente de las entrevistas con los stakeholder del área de Marketing de la Cooperativa Atuntaqui son:

- La versión beta del sistema estará lista el tercer semestre del 2011.

- El sistema será diseñado sobre la plataforma Adobe AIR y Java cumpliendo los estándares de calidad vigente para desarrollo de software. Este se conseguirá cumpliendo con el estándar PMI para dirección de proyectos, metodología RUP para el proceso de ingeniería de software y Adobe AIR para la construcción de la aplicación.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

➤ **Entregables del proyecto**

A continuación se indica y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de UP²⁴ desde la perspectiva de artefactos, y que propaga para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de UP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos.

➤ **Plan de Desarrollo de Software.**

En el presente documento.

➤ **Visión**

²⁴ UP.- Unificación de Procesos

Este documento define la visión del proyecto desde la perspectiva de los usuarios, especificando las necesidades y características del proyecto. Constituye una base de acuerdo en cuanto a los requerimientos del sistema.

➤ **Glosario**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

➤ **Especificaciones de Casos de Uso**

Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

Prototipos de Interfaces de Usuario

Se trata de prototipos que permiten al usuario hacerse una idea de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema. Estos prototipos se realizarán como: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados. Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

➤ **Modelo de Análisis y Diseño**

Este modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación) hacia una de diseño (incluyendo una orientación hacia el entorno de implementación), de acuerdo al avance del proyecto.

➤ **Modelo de Datos**

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelado relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).

➤ **Modelo de Implementación**

Este modelo es una colección de componentes. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y demás ficheros necesarios para la implantación y despliegue del sistema. (Este modelo es sólo una versión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento).

➤ **Modelo de Despliegue**

Este modelo muestra el despliegue de la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

➤ **Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración. Cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizar la prueba, y

dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

➤ **Lista de Riesgos**

Este documento incluye una lista de los riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

➤ **Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Manual de Usuario y Manual Técnico.

➤ **Producto**

Los archivos del sistema BlueMarking están empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva release²⁵ al final de cada iteración.

3.4. Organización del Proyecto

➤ **Participantes en el Proyecto**

De momento no se incluye el personal que designará Responsable del Proyecto, Comité de Control y Seguimiento, otros participantes que se estimen convenientes para proporcionar los requisitos y validar el sistema.

El resto del personal del proyecto considerando las fases de Inicio, Elaboración y dos iteraciones de la fase de Construcción, estará formado por los siguientes puestos de trabajo y personal asociado:

²⁵ **Release.**- Nueva versión de una aplicación informática.

- **Jefe de Proyecto.** Con una experiencia en metodologías de desarrollo, herramientas CASE y notaciones, en particular la notación UML y el proceso de desarrollo RUP.

- **Analista de Sistemas.** El perfil establecido es: Ingeniero en Informática con conocimientos de UML, uno de ellos al menos con experiencia en sistemas afines a la línea del proyecto

- **Analistas – Programadores.** Con conocimientos en el entorno de desarrollo del proyecto, con el fin de que los prototipos puedan ser lo más cercanos posibles al producto final. Este trabajo ha sido encomendado a Manuel Jesús Cepeda Arteaga.

- **Ingeniero de Software.** El perfil establecido es: Ingeniero en Informática que participará realizando labores de gestión de requisitos, gestión de configuración, documentación y diseño de datos. Encargada de las pruebas funcionales del sistema, realizará la labor de Tester.

- **Interfaces Externas**

Se define los participantes del proyecto que proporcionarán los requisitos del sistema, y entre ellos quiénes serán los encargados de evaluar los artefactos de acuerdo a cada subsistema y según el plan establecido. El equipo de desarrollo interactuará activamente con los participantes para especificación y validación de los artefactos generados.

- **Roles y Responsabilidades**

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.

Puesto	Responsabilidad
Jefe de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.
Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

Fuente: Propia

Tabla 3.1. Roles y Responsabilidades

3.5. Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

➤ Plan de las Fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar).

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	6 semanas
Fase de Elaboración	2	8 semanas
Fase de Construcción	2	14 semanas
Fase de Transición	2	6 semanas

Fuente: Propia

Tabla 3.2. Plan de Fases

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de Inicio	En esta fase desarrollará los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. La aceptación del cliente / usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta

	<p>fase, todos los casos de uso correspondientes a requisitos que serán implementados en el primer release de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. En nuestro caso particular, por no incluirse las fases siguientes, la revisión y entrega de todos los artefactos hasta este punto de desarrollo también se incluye como hito. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos. Ambas iteraciones tendrán una duración de ocho semanas.</p>
Fase de Construcción	<p>Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 módulos, cada uno produciendo un release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 3.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada.</p>
Fase de Transición	<p>En esta fase se prepararán y asegura una implantación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la</p>

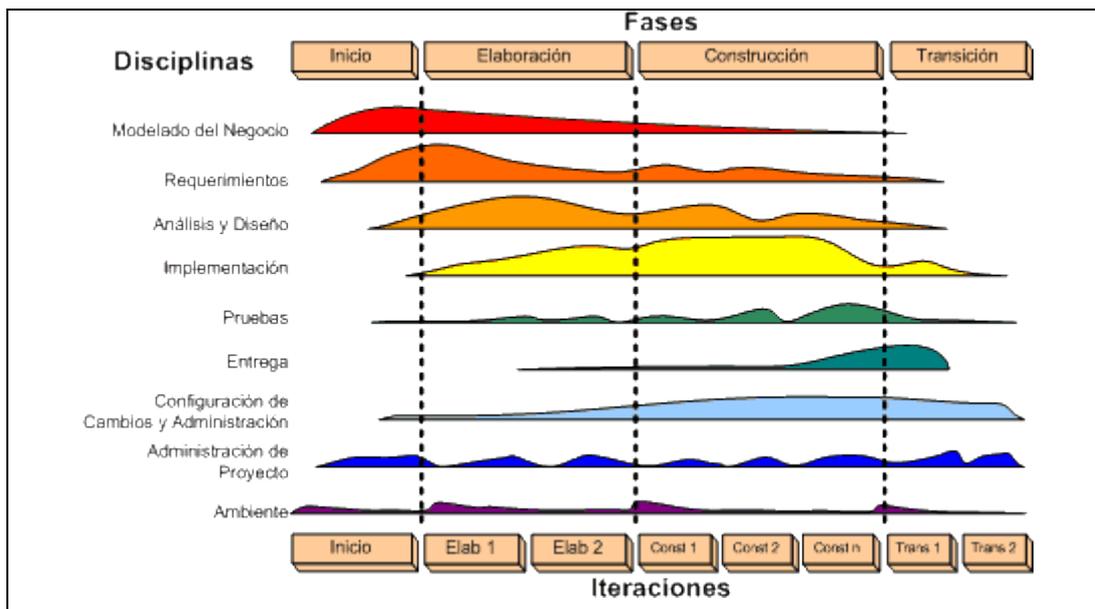
	finalización del entrenamiento de los usuarios.
--	---

Fuente: Propia

Tabla 3.3. Plan de Fases: Hitos

➤ **Calendario del Proyecto**

A continuación se presenta un calendario de las principales tareas del proyecto incluyendo sólo las fases de Inicio y Elaboración. Como se ha comentado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase e iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (workflow²⁶) en un momento determinado del desarrollo.



Fuente: [17]

Figura 3.1. Descripción del Proceso Unificado

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

²⁶ **Workflow:** Flujo de trabajo.

[17] **RUP Etapa diseño. "2011"**, <http://consolida-it.com/marco.htm>.

Disciplinas / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 4	Semana 5
Requisitos		
Glosario	Semana 2	Semana 5
Visión	Semana 2	Semana 4
Modelo de Casos de Uso	Semana 3	siguiente fase
Especificación de Casos de Uso	Semana 5	siguiente fase
Especificaciones Adicionales	Semana 5	siguiente fase
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 6	siguiente fase
Modelo de Datos	Semana 7	siguiente fase
Implementación		
Prototipos de Interfaces de Usuario	Semana 8	Semana 9
Modelo de Implementación	Semana 10	siguiente fase
Pruebas		
Casos de Pruebas Funcionales	Semana 11	siguiente fase
Despliegue		
Modelo de Despliegue	Semana 11	siguiente fase
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 1.0 y planes de las Iteraciones	Semana 1	Semana 2
Ambiente	Durante todo el proyecto	

Fuente: Propia

Tabla 3.4. Roles y Responsabilidades

Disciplinas / Artefactos generados o modificados durante la Fase de Elaboración	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Semana 4	Aprobado
Requisitos		
Glosario	Semana 2	Aprobado
Visión	Semana 2	Aprobado
Modelo de Casos de Uso	Semana 3	Semana 5
Especificación de Casos de Uso	Semana 5	Semana 5
Especificaciones Adicionales	Semana 5	Semana 5
Análisis / Diseño		
Modelo de Análisis / Diseño	Semana 6	Revisar en cada iteración
Modelo de Datos	Semana 7	Revisar en cada iteración
Implementación		
Prototipos de Interfaces de Usuario	Semana 8	Revisar en cada iteración
Modelo de Implementación	Semana 10	Revisar en cada iteración
Pruebas		
Casos de Pruebas Funcionales	Semana 11	Revisar en cada iteración
Despliegue		
Modelo de Despliegue	Semana 11	Revisar en cada iteración

Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan de Desarrollo del Software en su versión 0.9 y planes de las Iteraciones	Semana 1	Revisar en cada iteración
Ambiente	Durante todo el proyecto	

Fuente: Propia

Tabla 3.5. Roles y Responsabilidades

➤ Seguimiento y Control del Proyecto

▪ Gestión de Requisitos

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá una serie de atributos tales como importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante una Solicitud de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

▪ Control de Plazos

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto y por el Comité de Seguimiento y Control.

▪ Control de Calidad

Los defectos detectados en las revisiones y formalizados también en una Solicitud de Cambio tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias. Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist²⁷ incluidas en RUP.

- **Gestión de Riesgos**

A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos²⁸ o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

- **Gestión de Configuración**

Se realizará una gestión de configuración para llevar un registro de los artefactos generados y sus versiones.

También se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto. Al final de cada iteración se establecerá un registro del estado de cada artefacto, estableciendo una versión, el cual podrá ser modificado sólo por una Solicitud de Cambio aprobada.

²⁷ **Checklist:** Lista de verificación.

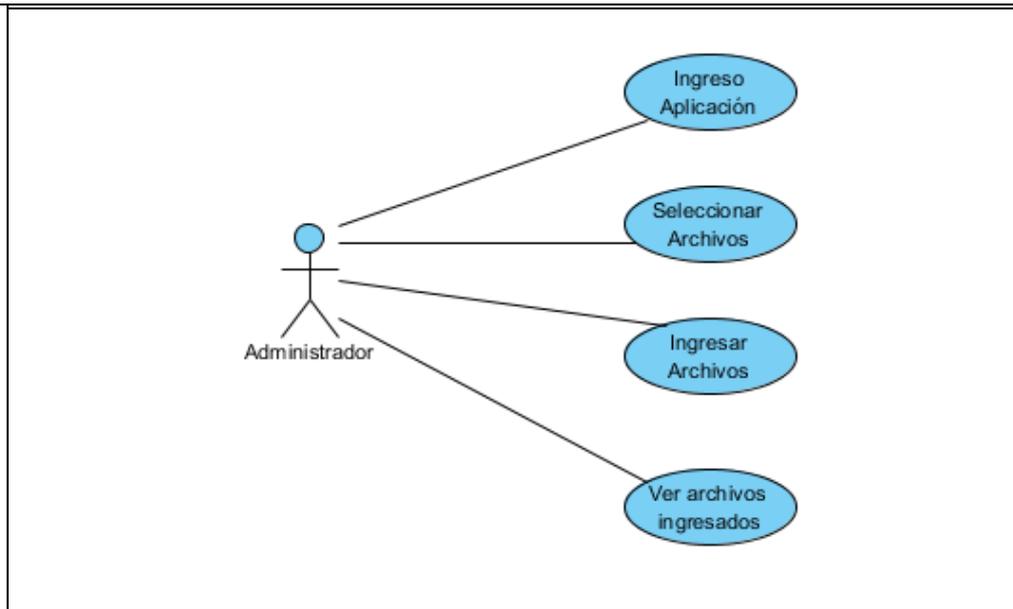
²⁸ **Mitigarlos:** Modelar, aplacar o suavizar la dureza de algo.

CAPITULO IV: FASE D ELABORACIÓN

4. ESPECIFICACION DE LOS CASOS DE USO

Esta sección presenta el modelo de casos de uso del presente sistema, en la cual se muestra las iteraciones y los casos de uso. Este Modelo de casos de Uso es un modelo de las funciones esperadas del sistema y su ambiente, y sirve como un contrato entre el cliente y los desarrolladores. Este modelo es utilizado como una entrada fundamental para las actividades de análisis, diseño y prueba. El rol más importante de un Modelo de Casos de Uso es identificar el comportamiento deseado del sistema con clientes y usuarios finales. En consecuencia, el modelo debe ser fácil de entender.

4.1. Especificación Caso de Uso: Carga de Archivos



Fuente: Propia

Figura 4.1. UC. Carga de Archivos.

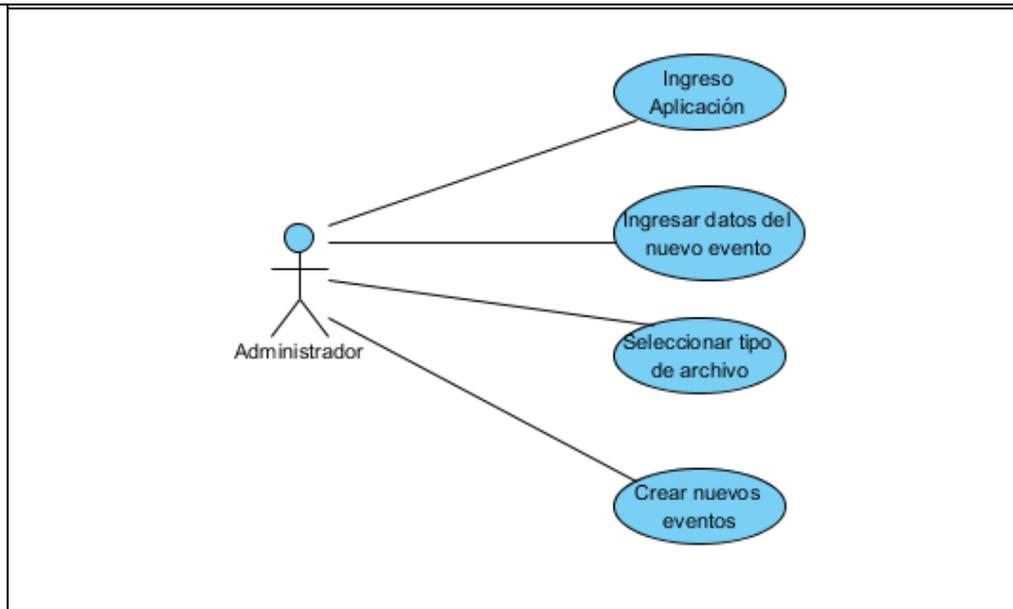
Caso de Uso	Carga de Archivos
Código	CU-001
Descripción	En este módulo se realiza la carga de los archivos que se utilizan en la creación de los Eventos, estos archivos pueden ser Videos, Imágenes, Audio y Texto. Aquí debemos tomar en cuenta que al cargar un video debemos cargar los dos formatos el 3gp y el flv que nos sirven para el envío y para la visualización en el sistema respectivamente.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la

	aplicación AIR.
Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Abrimos la aplicación BlueMarking.	
2. Seleccionamos la pestaña Archivos y Agregar.	
3. Damos clic en el botón Agregar.	
4. Navegamos hasta la ubicación de los archivos.	
5. Seleccionamos los archivos, Aceptar	
6. Damos clic en el botón Subir Archivos.	
7. Visualizamos los archivos cargados en el sistema.	

Fuente: Propia

Tabla 4.1. UC. Carga de Archivos.

4.2. Especificación Caso de Uso: Creación de Eventos (campañas)



Fuente: Propia

Figura 4.2. UC. Creación de Eventos.

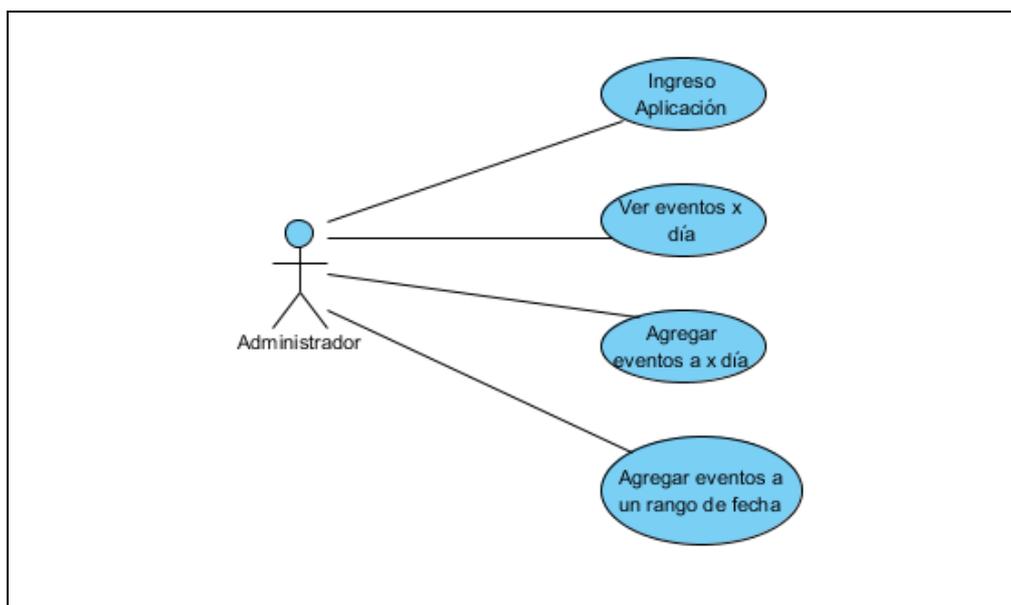
Caso de Uso	Creación de Eventos
Código	CU-002
Descripción	En este módulo creamos nuevos eventos que utilizamos para la asignación de eventos a determinado día, al crear nuevos eventos debemos ingresar el nombre, la descripción y seleccionar el tipo de archivo, este puede ser Video, Imagen, Sonido y Texto. Si ingresamos bien los datos se crea el nuevo evento exitosamente.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la aplicación AIR.
Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Abrimos la aplicación BlueMarking.	

- | |
|---|
| 2. Seleccionamos la pestaña Evento y Nuevo. |
| 3. Ingresamos Datos del nuevo evento. |
| 4. Seleccionamos el tipo de archivo (Imagen, Video, Audio o Texto). |
| 5. Damos clic en el botón Nuevo Evento. |
| 6. Mensaje de “Ingreso Nuevo Evento”. |

Fuente: Propia

Tabla 4.2. UC. Creación de eventos.

4.3. Especificación Caso de Uso: Asignar Eventos Diarios.



Fuente: Propia

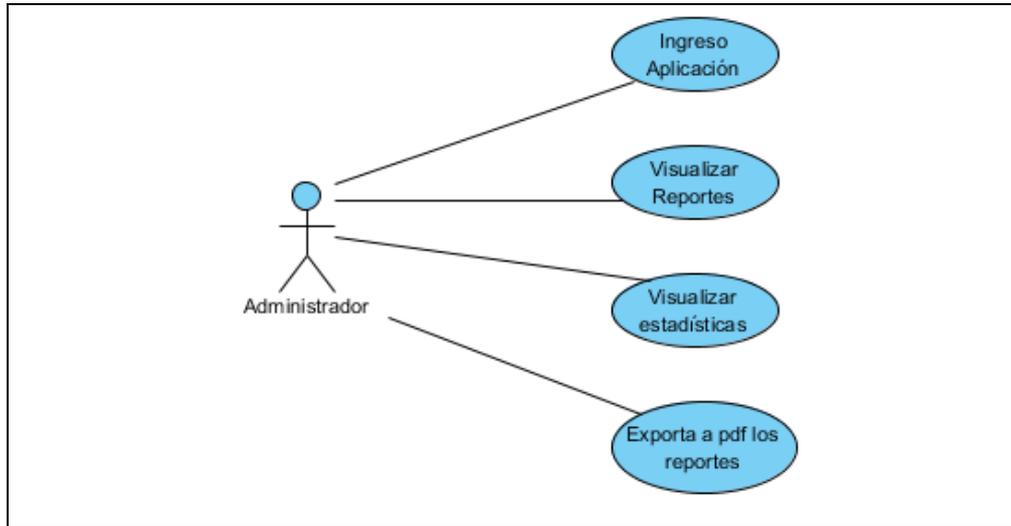
Figura 4.3. UC. Asignar Eventos.

Caso de Uso	Asignar Eventos
Código	CU-003
Descripción	En este módulo asignamos los eventos creados a determinado día del calendario, este evento es el que se va a transferir a un dispositivo móvil. Aquí también podemos ver los eventos que tiene un día. Y podemos crear un evento diario a un rango de fechas.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la aplicación AIR.
Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Abrimos la aplicación BlueMarking.	
2. Seleccionamos la pestaña Ver y Eventos.	
3. Seleccionar Flujo Alterno 1.	
4. Seleccionar Flujo Alterno 2.	
Flujo Alternativo 1 (evento diario)	
1. Damos clic en un día del mes.	
2. Damos clic en Agregar Eventos.	
3. Seleccionamos el Evento.	
4. Ingresamos el nombre del nuevo evento diario.	
5. Damos clic en el botón OK.	
Flujo Alternativo 1 (evento masivo)	
1. Damos clic en el botón Agregar Eventos.	
2. Seleccionamos un rango de Fechas.	
3. Seleccionamos el Evento.	
4. Ingresamos el nombre del nuevo evento diario.	
5. Damos clic en el botón OK.	

Fuente: Propia

Tabla 4.3. UC. Asignar Eventos.

4.4. Especificación Caso de Uso: Reportes y Estadísticas.



Fuente: Propia

Figura 4.4. UC. Reportes y Estadísticas.

Caso de Uso	Carga de Archivos
Código	CU-003
Descripción	Tenemos los reportes de los archivos, evento creado y eventos enviados, que nos sirven para poder tomar las mejores decisiones a futuro, estos resultados se pueden exportar a EXCEL.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la aplicación AIR.
Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Abrimos la aplicación BlueMarking.	
2. Seleccionamos la pestaña Reportes y Reportes y Estadísticas.	
3. Seleccionamos las diferentes pestañas (Cada una es un reporte o estadística).	

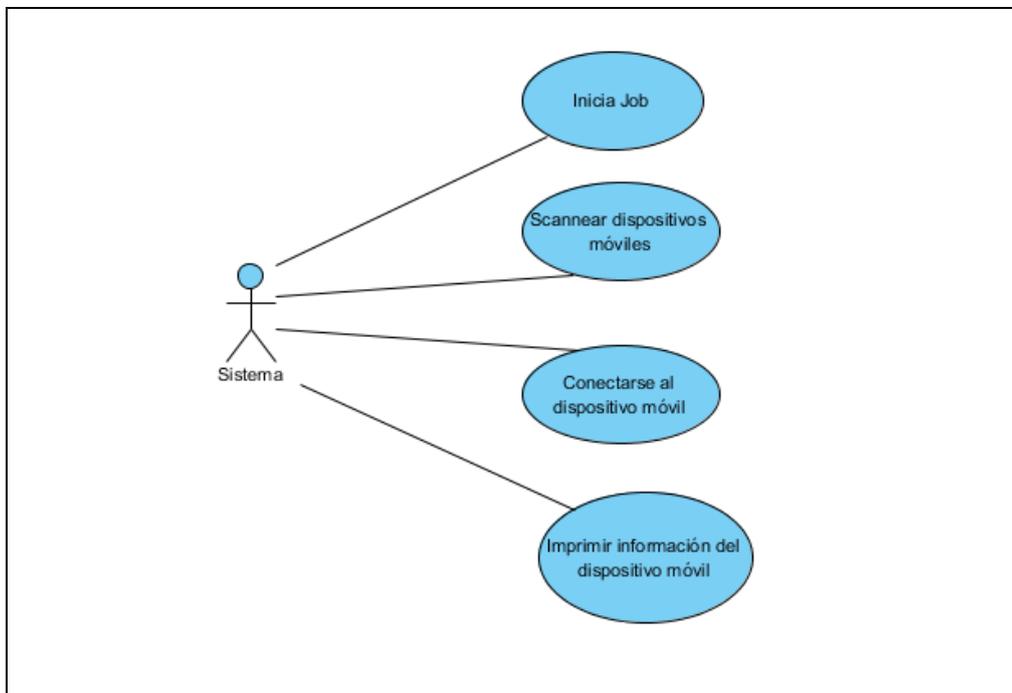
4. Seleccionar Flujo Alterno 1.
5. Seleccionar Flujo Alterno 2.
6. Seleccionar Flujo Alterno 3.
7. Seleccionar Flujo Alterno 4.
8. Seleccionar Flujo Alterno 5.
9. Seleccionar Flujo Alterno 6.
Flujo Alternativo 1
1. Seleccionamos Tipo de Archivos.
2. Visualizamos las estadísticas de los tipos de archivos.
3. Exportamos a Excel los datos.
Flujo Alternativo 2
1. Seleccionamos Total eventos enviados.
2. Visualizamos las estadísticas de eventos enviados y no enviados.
3. Exportamos a Excel los datos.
Flujo Alternativo 3
1. Seleccionamos Eventos por mes.
2. Seleccionamos un rango de fecha.
3. Damos clic en el botón Eventos Mensuales.
4. Visualizamos las estadísticas de los eventos enviados por mes
5. Exportamos a Excel los datos.
Flujo Alternativo 4
1. Seleccionamos Eventos por día.
2. Seleccionamos un rango de fecha.
3. Damos clic en el botón Eventos Por Día.

4. Visualizamos las estadísticas de los eventos enviados por día.
5. Exportamos a Excel los datos.
Flujo Alternativo 5
1. Seleccionamos Nro. Eventos enviados
2. Visualizamos las estadísticas de los eventos enviados agrupados por tipo de archivo.
3. Exportamos a Excel los datos.
Flujo Alternativo 6
1. Seleccionamos Eventos enviados
2. Visualizamos las estadísticas de los eventos enviados.
3. Exportamos a Excel los datos.

Fuente: Propia

Tabla 4.4. UC. Reportes y Estadísticas.

4.5. Especificación Caso de Uso: Buscar teléfonos celulares.



Fuente: Propia

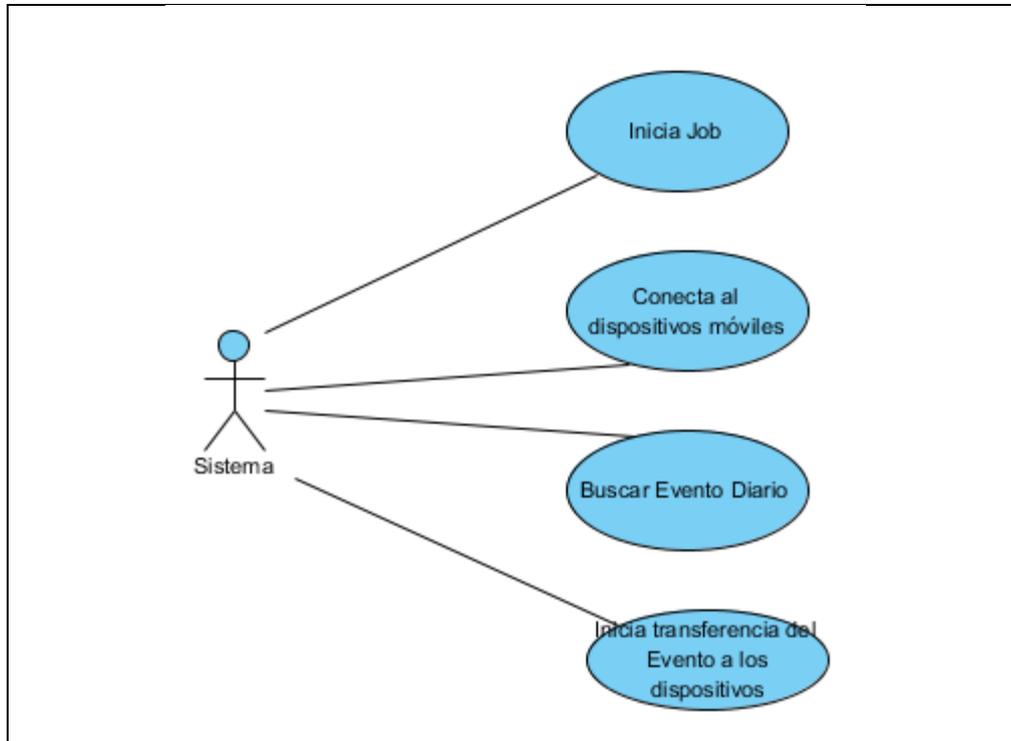
Figura 4.5. UC. Buscar teléfonos celulares.

Caso de Uso	Asignar Eventos
Código	CU-003
Descripción	En este módulo es el encargado de buscar teléfonos celulares con Bluetooth encendido, en un determinado rango, que puede ser entre 10m y 100m de diámetro.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la aplicación AIR.
Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Inicia Job (Proceso Automático)	
2. Comienza a scannear teléfonos celulares con Bluetooth encendido.	
3. Conectarse al dispositivo móvil.	
4. Imprimir información de los dispositivos encontrados.	

Fuente: Propia

Tabla 4.5. UC. Asignar Eventos.

4.6. Especificación Caso de Uso: Transferencia de Eventos.



Fuente: Propia

Figura 4.6. UC. Trasferencia de Eventos.

Caso de Uso	Asignar Eventos
Código	CU-003
Descripción	En este módulo es el encargado de transferir los evento de un determinado día a los teléfono celulares de los posible clientes y de almacenar en la base de datos la UUID de los dispositivos encontrados.
Autor(es)	Usuario - Personal de la Cooperativa Atuntaqui, encargados del sistema.
Precondiciones	Debe estar levantada la aplicación Java e instalada la aplicación AIR.

Post condición	Ninguna
Flujo del Evento	
Flujo Principal	
1. Inicia Job (Proceso Automático)	
2. Conectarse al dispositivo móvil.	
3. Buscar Evento Diario.	
4. Enviar evento al dispositivo móvil.	
5. Guardar la información del dispositivo.	

Fuente: Propia

Tabla 4.6. UC. Traslación de Eventos.

CAPITULO V: FASE DE CONSTRUCCIÓN

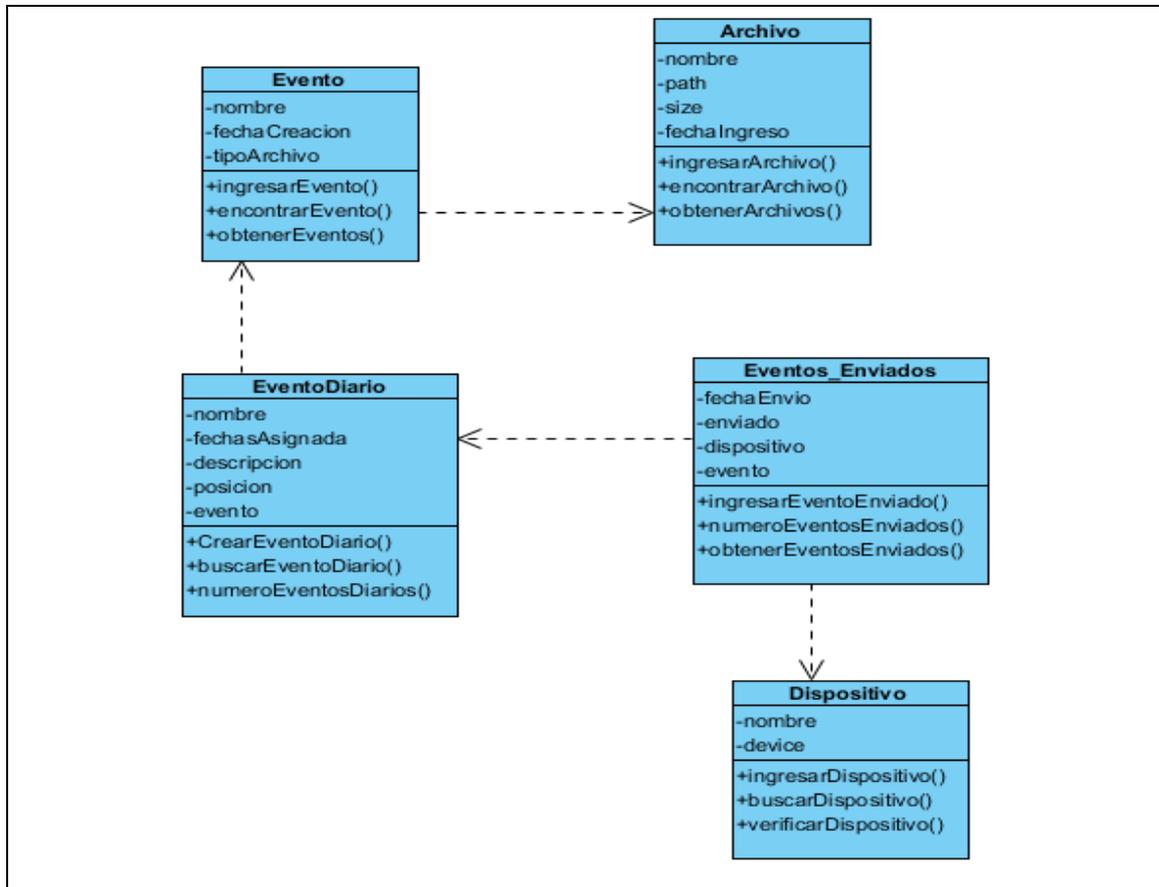
5. VISTA LÓGICA

5.1. Diagrama de Clases

Debido a que más y más clases son añadidas a un modelo, una representación textual de las clases no es suficiente. En un sistema Orientado a Objetos, su visión lógica se construye, fundamentalmente, definiendo las clases. Los objetos identificados en el dominio de aplicación son agrupados en clases, cada una de las cuales representa una categoría de objetos que tienen iguales propiedades (atributos) y comportamientos (operaciones, métodos o servicios). La clase como componente principal del sistema, y las relaciones entre ésta y otras clases, hacen posible la estructuración lógica de un sistema Orientado a Objetos. Los Diagramas de Clase son creados para proveer una visualización de algunas o todas las clases en el modelo. En UML (Unified Modeling Language), los diagramas que permiten visualizar la descomposición del sistema en clases mostrando su estructura genérica, se denominan Diagramas de Clase.

El presente sistema está diseñado para enviar promociones y ofertas a un teléfono celular utilizando Bluetooth. Cada envío de una promoción u oferta lo denominamos Evento (campana), a cada evento va a estar asociado un tipo de archivo (Imagen, Video, Audio o Texto). Los eventos creados va a ser asignados a los diferentes días del año (eventosDiarios) y estos eventosDiarios van a ser enviados por el presente sistema a los teléfonos celulares que se encuentre en un rango que pueda ser detectado por el Bluetooth conectado a la PC donde está instalada la aplicación.

A continuación presentamos el diagrama de clase de nuestra aplicación.



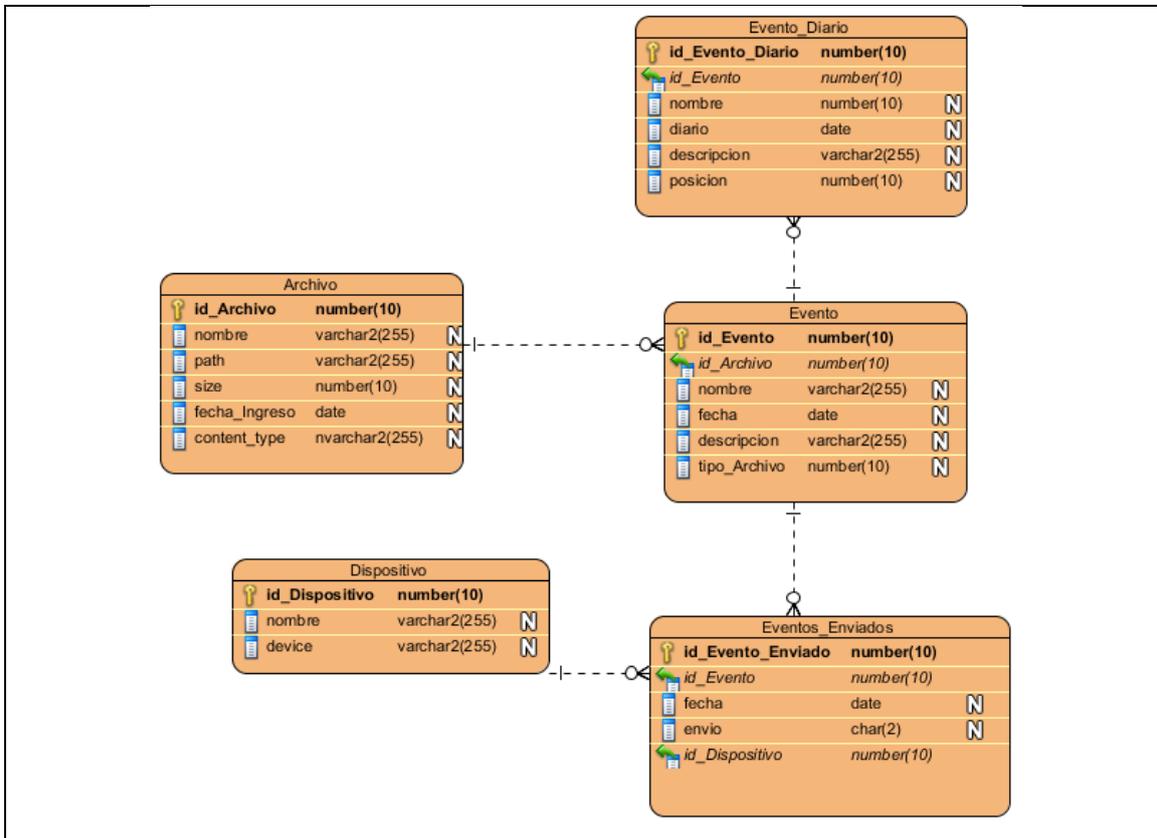
Fuente: Propia

Figura 5.1. Diagrama de Clases.

5.2. Diagrama Entidad Relación

La base de datos es el lugar donde la información quedara almacenada tras la interacción entre el usuario y el sistema. Esta sin duda es una pieza clave de la aplicación puesto que sin un gestor de base de datos, no se podría dejar constancia de los posibles cambios producidos.

En el diagrama entidad-relación con sus tablas y sus relaciones a implementar en el Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth es el siguiente.



Fuente: Propia

Figura 5.2. Diagrama Entidad Relación del sistema BlueMarking.

5.3. Arquitectura del Sistema

La solución propuesta para la realización del Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth, consiste en una arquitectura de escritorio separada en tres capas: capa de presentación, capa de negocio y capa de datos.

El hecho de dividir el sistema en capas permite tener un sistema bien estructurado, donde cada capa es independiente del resto. Con ello consigue un sistema estandarizada, reutilizable y con un bajo acoplamiento.

➤ **Capa de Presentación**

La capa de presentación es la que conocemos como la interfaz de usuario, nos ayuda a interactuar con la aplicación, permite visualizar la información ingresada por el usuario para evitar errores, así como los resultados de la aplicación. Esta capa debe tener la característica de ser amigable (entendible y fácil de usar) para el usuario. Esta capa está desarrollada en Adobe AIR.

➤ **Capa de Negocio**

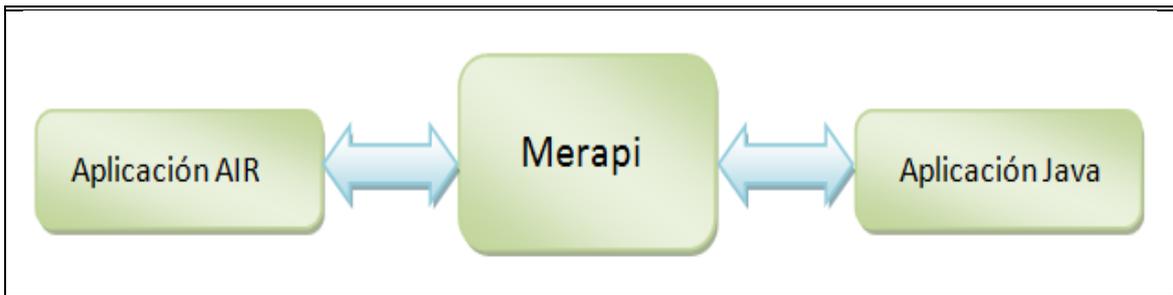
La capa de negocio también conocida como capa de dominio, es la que mantiene toda la lógica y las reglas del negocio. Se encarga de validar datos, realizar las operaciones y cálculos con los datos y gestionar las interacciones con la capa de datos y la capa de presentación. Esta capa está desarrollada en Java.

➤ **Capa de Datos**

La capa de datos es la que se encarga de la persistencia y recuperación de los datos desde la base de datos relacional.

Para la comunicación entre la capa de presentación y la capa de negocio utilizamos Merapi, que es un puente que permite la comunicación a través del uso de los mensajes entre aplicaciones Adobe AIR y programas escritos en Java.

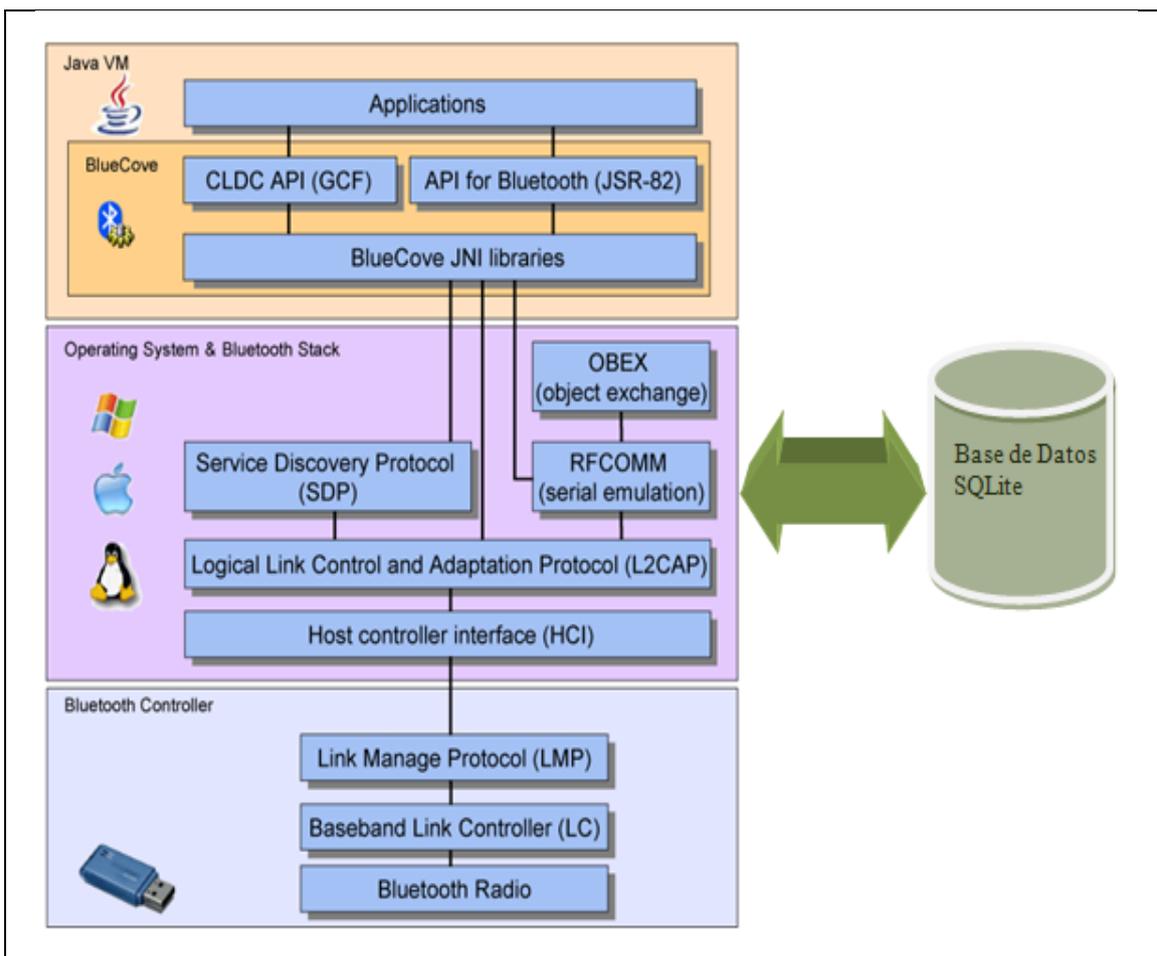
La figura nos indica los componentes y arquitectura del presente sistema.



Fuente: Propia

Figura 5.3. Diagrama de Arquitectura.

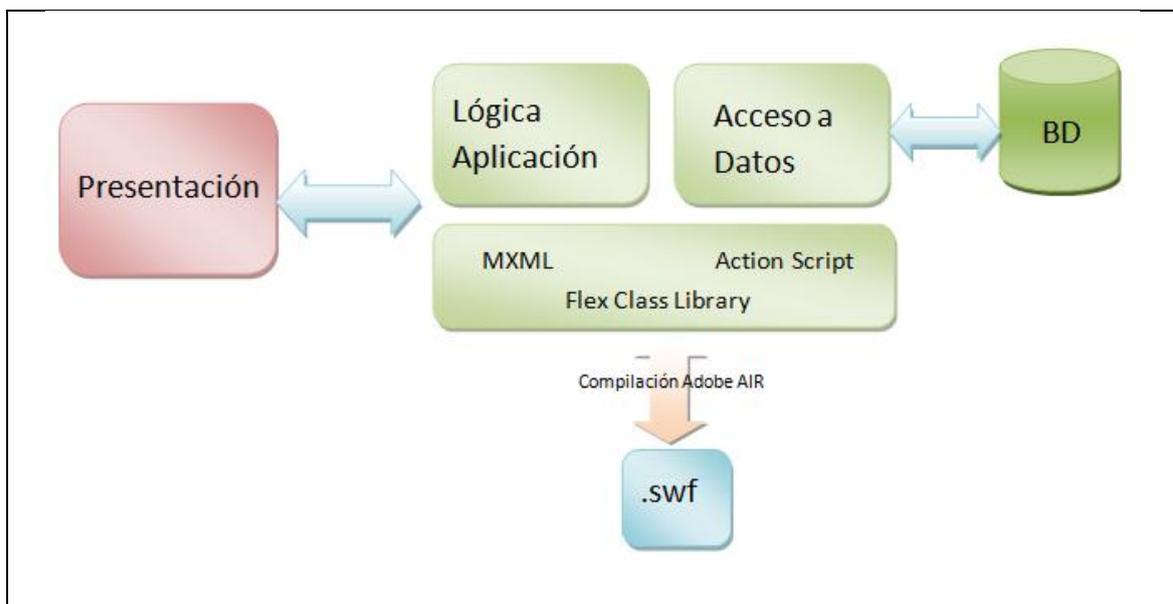
➤ Arquitectura del módulo Java



Fuente: Propia

Figura 5.4. Diagrama de Arquitectura Java

➤ **Arquitectura del módulo de Adobe AIR**



Fuente: Propia

Figura 5.5. Diagrama de Arquitectura Adobe AIR

CAPITULO VI: FASE DE IMPLEMENTACION

6. ESPECIFICACIONES DE CASOS DE PRUEBA

6.1. Especificación de Caso de Prueba: Carga de Archivos

- **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Carga de Archivo”. El objetivo será verificar que un usuario pueda visualizar los archivos del disco duro y pueda agregarlos a la aplicación. El entorno del cual partiremos para realizar la prueba será el formulario de agregar archivos en el módulo de Archivos.

- **Comprobar si podemos agregar archivos a la aplicación**

Ingresamos a la aplicación en la ventana de Archivo y Agregar, en la parte inferior damos clic en Agregar para podemos navegar en el disco de la PC y poder ver los archivos, la aplicación permite seleccionar uno o más archivos, Además nos validad que un archivo no se seleccione más de una vez.

- **Condiciones de ejecución**

Las condiciones de ejecución del caso de prueba son que se pueden visualizar los archivos y se puedan seleccionarlos y agregarlos a la aplicación.

- **Entrada**

- Ingresamos en la pestaña Archivo.
- Seleccionamos Agregar.
- Seleccionamos los archivos que deseamos cargar a la aplicación.
- Pulsamos el botón Abrir.
- Pulsamos el botón Subir Archivos.

- **Resultado esperado**

El sistema nos muestra una pantalla donde podemos ver los archivos cargados en la aplicación, en los que se visualiza los archivos antes cargados.

- **Evaluación de la Prueba**

Prueba superada con éxito.

6.2. Especificación de Caso de Prueba: Crear de Eventos

- **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Creación de Eventos”. El objetivo de esta prueba es verificar que un usuario pueda crear nuevos eventos en la aplicación. El entorno del cual partiremos para realizar la prueba será el formulario de Nuevo en el módulo de Evento.

- **Comprobar si podemos Crear Nuevos Eventos**

Ingresamos a la aplicación en la ventana de Evento y clic Nuevo y podemos ver el formulario de ingreso para un nuevo evento, ingresamos toda la información básica, nombre, descripción, y seleccionamos el tipo de Archivo.

- **Condiciones de ejecución**

Las condiciones de ejecución del caso de prueba son que el usuario pueda crear nuevos eventos, ingresando los datos correctamente.

- **Entrada**

- Seleccionar la pestaña Evento
- Damos clic en Nuevo
- Ingresamos el Nombre del Evento
- Ingresamos la Descripción
- Seleccionamos el Tipo de Archivo
- Pulsamos el botón Nuevo Evento

- **Resultado esperado**

El sistema nos muestra un mensaje “Ingreso Nuevo Evento”.

- **Evaluación de la Prueba**

Prueba superada con éxito.

6.3. Especificación de Caso de Prueba: Asignar Eventos Diarios

- **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Asignar Eventos Diarios”. El objetivo de esta prueba es verificar que un usuario pueda asignar nuevos eventos a un determinado día. El entorno del cual partiremos para realizar la prueba será el formulario de Eventos en el módulo de Agenda.

- **Comprobar si podemos la Asignación de Eventos**

Ingresamos a la aplicación en la ventana de Agenda y clic en Eventos y podemos ver la

agenda que representa a un mes con los respectivos días, esta nos ayuda a la asignación de eventos a los días.

- **Condiciones de ejecución**

Las condiciones de ejecución del caso de prueba son que el usuario pueda asignar nuevos eventos a un determinado día y en un rango de fechas correctamente.

- **Entrada**

- Seleccionar la pestaña Ver
- Damos clic en Eventos
- Damos clic en día del mes.
- Nos despliega un popUp.
- Pulsamos el botón Agregar Eventos.
- Seleccionamos el evento que vamos agregar al día.
- Ingresamos le nombre del evento diario
- Pulsamos el botón OK
- Vemos como se incrementa el número de evento para ese día

- **Resultado esperado**

El sistema asigne correctamente los eventos al día seleccionando.

- **Evaluación de la Prueba**

Prueba superada con éxito.

6.4. Especificación de Caso de Prueba: Trasferencia de Eventos

- **Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso “Trasferencia de Eventos”. El objetivo de esta prueba es verificar que la aplicación pueda transferir evento a los teléfonos celular con Bluetooth encendido encontrados en un rango determinado. El entorno del cual partiremos para realizar la prueba será el inicio automático de un proceso que realiza para cumplir este requerimiento.

- **Comprobar si podemos Crear Nuevos Eventos**

Inicia el Job o proceso automáticamente, escanea en un rango determinado si existen teléfonos celulares con el Bluetooth encendido y transferir los eventos.

- **Condiciones de ejecución**

Las condiciones de ejecución del caso de prueba son que la aplicación encuentre teléfonos celulares en un rango determinado y pueda transferir eventos correctamente.

- **Entrada**

- Inicia proceso automático
- Escanear en un rango determinado teléfonos celulares con el Bluetooth encendido.
- Sincronizarse con los teléfonos celulares encontrados
- Buscar el Evento que va ha ser transferido al teléfono celular
- Transferir el evento
- Guardar el UUID del teléfono celular

- **Resultado esperado**

El sistema nos muestra un mensaje “Ingreso Exitoso al dispositivo XXXXXXXXXX”.

- **Evaluación de la Prueba**

Prueba superada con éxito.

CAPITULO VII:

7. LISTA DE RIESGOS

#	Descripción del Riesgo	Impacto	Probabilidad de Ocurrencia	Estrategia de mitigación del riesgo
1	El reléase R1 del BlueMarking podría no salir para el mes en que se planifica entregar la aplicación.	7	30%	Incrementar esfuerzo.
2	La rango de trasferencia del Evento es limitado por el dispositivo Bluetooth	6	30%	Instalar una versión actual del dispositivo Bluetooth y que sea de clase 1.
4	Modificar los requerimientos puestos en marcha	5	20%	Realizar actas de reunión de manera que el usuario revise y verifique sus requerimientos y apruebe dicha acta con firma.

6	Expectativas irreales	8	30%	<p>Delimitar adecuadamente el alcance del proyecto.</p> <p>Planificar adecuadamente los tiempos de desarrollo.</p> <p>No crear falsas expectativas en los desarrolladores del proyecto, para evitar frustración en el caso de que no se logran las metas previstas.</p>
---	-----------------------	---	-----	---

Fuente: Propia

Tabla 6.1. Lista de Riesgos

CAPITULO VIII:

8. CONCLUSIONES Y RECOMENDACIONES

8.1. Conclusiones

- El objetivo principal de este proyecto de Tesis, fue desarrollar un sistema de transferencia de publicidad a través de Bluetooth a teléfonos celulares en un rango determinado por el dispositivo Bluetooth, que puede variar en un radio entre 10 y 100 metros. La simulación de la aplicación funciona en su totalidad y permite encontrar teléfonos celulares en radio y una vez encontrado comienza la transferencia de la publicidad, esta publicidad se envía por medio de un Bluetooth conectado a la PC a los teléfonos celulares encontrados. Esta aplicación puede enviar Imágenes, Videos, Audio y Texto como parte de la publicidad y propaganda de servicios y productos.
- Este sistema puede ser implementado en cualquier organización que desee tener un sistema informático de Marketing, con el cual puede realizar tareas como son: buscar teléfonos celulares, y enviar mensajes con la publicidad que desee. Esto permite utilizar los teléfonos celulares de los posibles clientes como receptores de publicidad reduciendo el costo.
- Además el presenta sistema puede ser implementado en Hoteles, Supermercados, Aeropuertos, etc. En cualquier lugar donde el transito del público sea masivo. Nosotros

la implementación del presente sistema BlueMarketing la hemos efectuado en Cooperativa Atuntaqui y de esta manera esperamos un incremento de clientes.

- Para desarrollar el presente sistema hemos utilizado software libre, ya que provee de herramientas útiles, adaptables a nuestro entorno y sin ningún costo. Con estas herramientas creamos un módulo con Adobe AIR para la presentación y un módulo con Java para el manejo del dispositivo Bluetooth.
- Se recogió información sobre la opinión del público de la ciudad de Ibarra para conocer el impacto del presente sistema de marketing por Bluetooth mediante el uso de encuestas, que arrojaron como resultado el gran nivel de interés y aceptación del presente sistema en la ciudad.
- Dichos resultados permitieron conocer a gran rango el nivel de penetración de la tecnología Bluetooth así como otras consideraciones importantes que son datos valiosos para la factibilidad de implementar el sistema BlueMarketing en la ciudad de Ibarra.
- Como resultado del estudio de mercado realizado en la ciudad de Ibarra, se concluyó que un gran porcentaje de la población cuenta con un teléfono celular con tecnología Bluetooth, lo que significa que es un potencial mercado para el envío de mensajes publicitarios.
- El estudio realizado en la ciudad de Ibarra justifica el diseño del sistema BlueMarketing para la Cooperativa Atuntaqui pueda utilizarlo en el envío de mensajes publicitarios de productos y servicios de la Cooperativa.
- En conjunto, todo nuestro análisis acerca de esta nueva aplicación de tecnología Bluetooth permite concluir que es favorable la investigación y desarrollo de aplicaciones en el sector de Telecomunicaciones para acercarnos más al público e ingresar en nuevos ámbitos de aplicaciones antes no desarrolladas.
- Al utilizar Adobe AIR para la elaboración del presente proyecto de tesis, me di cuenta que es una herramienta excelente para el diseño de aplicaciones RIAs, poseen una serie de características que la hacen sólidas frente al manejo de aplicaciones de escritorio, lo que a mi criterio hace a las aplicaciones con una interfaz de usuario muy agradable.

- Bluetooth es una excelente tecnología para la transmisión de datos en rango corto y a un costo mínimo, a diferentes teléfonos celulares que tengan Bluetooth y además este encendido.
- En este proyecto de tesis se ha implementado satisfactoriamente el manejo de la tecnología inalámbrica Bluetooth con la ayuda del API BlueCove para Java, permitiendo su instalación y utilización en los diferentes sistemas operativos.
- Se optó por utilizar SQLite como gestor de base de datos, ya que es un producto que ofrece características necesarias para desarrollar el sistema, y además se encuentra embebida en el sistema por lo que no necesita ser administrada.
- El manejo de la metodología RUP para el desarrollo de software implantado en la universidad ha sido de gran ayuda, ya que permite tener una idea clara en todas las fases de un proyecto, su estructura obliga al desarrollador a documentar todo el proceso desde el inicio, elaboración, pruebas y puesta a producción, contando con un registro detallado de todos los cambios realizados en el transcurso hasta la finalización del mismo.
- La elaboración de distintos diagramas y artefactos siguiendo la metodología RUP provee una fácil ejecución del proceso de elaboración del presente sistema, ya que estos diagramas describen cómo está estructurado el sistema desde diferentes perspectivas.
- El desarrollo de prototipos ayuda a visualizar de mejor manera los requerimientos del usuario ya que nos permite realizar pruebas prematuras del sistema con los usuarios reales.
- El sistema ofrece las funcionalidades necesarias para convertir este proyecto en un producto comercial que puede ser utilizado en cualquier lugar donde el tránsito de la población sea muy concurrente.

8.2. Recomendaciones

Al realizar la implementación del Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth, se debe tener en consideración las siguientes recomendaciones que ayudará al buen funcionamiento y desempeño del Sistema:

- Se recomienda instalar la aplicación BlueMarking en un lugar donde el tránsito de la población se masivo, para una mayor difusión de servicios y productos.
- Para cumplir los objetivos planteados, es fundamental la participación activa de los usuarios.
- Aprovechar las ventajas de AIR, de Adobe, para la creación de aplicaciones de escritorio elegantes para el usuario final.
- Fomentar un programa de comunicación social del proyecto. Se debe de informar a la población los detalles del presente sistema.

- Es recomendable realizar un estudio de mercado antes de empezar a desarrollar un sistema, para entender el comportamiento del mercado y la demanda, dentro de este estudio podemos medir el nivel de demanda y aceptación del presente sistema.
- Para obtener un máximo control de variables conlleva un desarrollo de aplicaciones y poder mantener una ordenada implementación de éstas, es importante seguir metodologías y estándares que nos llevan a estar en competitividad en todo momento.
- La metodología RUP cuenta con un enfoque disciplinado en la asignación de tareas y responsables dentro de una organización del desarrollo, con la cual se puede mantener una fácil administración de este proceso.
- Al implementar un Desarrollo Rápido de Aplicaciones de un Sistema particular, es importante la utilización de patrones, los cuales ya tienen una funcionalidad general y han sido predefinidos, y así contar con una base consistente y previamente elaborada para la implementación del Software.
- Coordinar con el personal técnico encargado del mantenimiento del sistema BlueMarking para un buen funcionamiento futuro del sistema.

9. GLOSARIO

- **Estándares:** Que sirve como tipo, modelo, norma, patrón o referencia.

- **RIA:** Aplicaciones informáticas con alta experiencia de usuario

- **RUP:** Rational Unified Process.

- **UP:** Unificación de Procesos

- **Release** Nueva versión de una aplicación informática.

- **Checklist** Lista de verificación.

- **Workflow** Flujo de trabajo.

- **Stakeholder** Es un término, para referirnos a quienes pueden afectar o son afectados por las actividades de una empresa.

- **Bits** Es un dígito de sistema de numeración binaria.

- **J2EE** (Java 2 Enterprise Edition) define un estándar para el desarrollo de aplicaciones empresariales multicapas diseñada por Sun Microsystems.

- **JSP** Es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.

- **Gateway** Es una puerta de enlace, un nodo en una red informática que sirve de punto de acceso a otra red.

- **Seguridad Sandbox** Es un mecanismo para ejecutar programas con seguridad y de manera separada.

- **FEC:** : (Forward Error Correction) es el protocolo de corrección de errores usado por el estándar DVB-H, puesto que las interfaces de entrada y salida son IP.

- **ARQ:** (Automatic Repeat-request) son protocolos utilizados para el control de errores en la transferencia de datos, garantizando la integridad de los mismos.

- **AES-128:** (Advanced Encryption Standard) es una técnica de cifrado de clave simétrica que reemplaza al Estándar de encriptación de Datos.

- **ISM:** (Industrial, Scientific and Medical) son bandas reservadas internacionalmente para el uso no comercial de radiofrecuencia electromagnética en el área industrial, científica y médica.

- **SIG:** (Special interest Group) es una asociación privada sin ánimos de lucro.

- **RSSI:** (Receive Signal Strength indication) se usa para medir el nivel de potencia de las señales recibidas en las redes inalámbricas.

- **AFH:** (Adaptive Frequency hopping) permite reducir interferencias en la banda de 2.4Hz.

- **HCI:** (Host Controller Interface) es una interfaz a nivel de registro que permite a un controlador de host para el hardware USB o FireWire comunicarse con un controlador de host en el software.

- **ISM:** (Industrial, Scientific and Medical) son bandas reservadas internacionalmente para el uso no comercial de radiofrecuencia electromagnética en el área industrial, científica y médica..

- **API:** (Application Programming Interface) es un conjunto de librerías de código JAVA compilado que se incluyen en la Java Runtime Environment.

- **Bluetooth:** Es una especificación para redes inalámbricas, permite la transmisión de voz y datos entre distintos dispositivos mediante una radiofrecuencia segura (2,4 GHz).

- **homogéneos:** Es aquello que pertenece o que está relacionado a un mismo género.

- **globalización:** Es una interdependencia económica creciente entre los países, provocada por el aumento del volumen y la variedad de las transacciones de bienes y servicios.

- **competitividad:** Capacidad de generar la mayor satisfacción de los consumidores al menor precio.

10. REFERENCIAS

Bibliografía

Adobe Systems Incorporated, Using ADOBE FLEX 4, 345 Park Avenue, San Jose, California 95110, USA.

Escuela Politécnica Nacional, Comunicaciones Inalámbricas Bluetooth, Iván Bernal, ph.D. Quito 2005, Ecuador.

Universidad de las Palmas de Gran Canaria, Marketing de proximidad y Herramientas OpenSource, Roberto Díaz Ortega, Gema Socorro Rodríguez, 2009

Sitios de Internet

- ^[1] **Estudio de Mercado, “Emprendedores”,** <http://www.blog-emprendedor.info/que-es-el-estudio-de-mercado>.

- ^[2] **Mercado, “Proyectos de inversión”,** <http://www.mailxmail.com/curso-proyectos-inversion/estudio-mercado>
- ^[3] **Objetivo del Estudio de Mercado, “Proyectos de inversión”,** <http://www.mailxmail.com/curso-proyectos-inversion/estudio-mercado>
- ^[4] **Segmentación del Mercado, “Marketing”,** <http://www.monografias.com/trabajos57/segmentacion-mercado/segmentacion-mercado.shtml>
- ^[5] **Qué es el Software Libre, “HispaLinux”,** <http://www.hispalinux.es/SoftwareLibre>
- ^[6] **Qué es el Software Libre, “Bluetooth”,** <http://es.wikipedia.org/wiki/Bluetooth>
- ^[10] **Tecnología Java. ”Java”,** <http://www.desarrolloweb.com/articulos/497.php>
- ^[11] **Definición de Merapi. ”Merapi”,** <http://java.dzone.com/articles/merapi-bridge-between-air-and>
- ^[13] **Definición Adobe Flex. ”Adobe Flex”,** http://help.adobe.com/en_US/Flex/4.0/UsingSDK/index.html
- ^[14] **Definición Adobe AIR. ”desarrolloweb”,** <http://www.desarrolloweb.com/articulos/que-es-adobe-air.html>
- ^[15] **Definición SQLite. ”SQLite”,** <http://www.sqlite.org>
- ^[16] **Definición RUP. ”RUP”,** http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational
- ^[17] **RUP Etapa diseño. ”2011”,** <http://consolida-it.com/marco.htm>

ANEXOS

ENCUESTA
DICCIONARIO DE DATOS
GUIA DE PROGRAMACION
PROTOTIPO DE INTERFAZ DE USUARIO
MANUAL DE USUARIO
MANUAL DE INSTALACIÓN

ANEXO A: Encuesta

A La presente investigación tiene como finalidad determinar la utilización del servicio Bluetooth para el envío de mensajes con información de promociones y ofertas de la Cooperativa Atuntaqui en la ciudad de Ibarra.

Utiliza usted un teléfono celular con tecnología Bluetooth?

Si No

Le gustaría recibir mensajes con información de nuestros productos, a través de Bluetooth?

Si No

Qué tipo de mensaje con información de nuestros productos, prefiere recibir?

- Video
- Imagen
- Audio

- () Texto
() Otros _____

Qué tipo de campaña promocional le gustaría recibir?

- Crédito de vivienda
 Crédito de consumo
 Microcrédito
 Crédito comercial

Recibe mensajes de ofertas en su teléfono a través de Bluetooth?

- () Si () No

En este momento los mensajes por Bluetooth que recibe de que empresa provienen.

- a) _____
b) _____
c) Ninguna

En el día cuantos mensajes de ofertas ha recibido a través de Bluetooth?

- () Ninguno
() 1
() 2
() 3 - 5
() 6 - 10
() Otro _____

DATOS TÉCNICOS

➤ *Edad*

10 – 18	18 – 30	30 – 45	45 – Adelante
---------	---------	---------	---------------

➤ *Nivel de Preparación*

Primario	Secundaria	Superior	Cuarto Nivel
----------	------------	----------	--------------

➤ *Género*

Masculino	Femenino
-----------	----------

➤ *A qué actividad se dedica*

ANEXO B: Diccionario de Datos

B.1. Nombre de las tablas del Sistema

- TAB_ARCHIVO
- TAB_DISPOSITIVO
- TAB_EVENTO
- TAB_EVENTO_DIARIO
- TAB_EVENTO_ENVIADO

B.2. Explicación de campos de cada tabla del Sistema

- **TAB_ARCHIVO**

NOMBRE_COLUMNA	# COLUM	CLAVE PRIMARIA	NUL L	TIPO DATO	VALOR POR DEFECTO	DESCRIPCION
ID_ARCHIVO	1	1	N	INTEGER		Campo numérico secuencial
NOMBRE	2		N	TEXT		Nombre del Archivo.
PATH	3		N	TEXT		Ubicación de Archivo
CONTENT_TYPE	4		N	TEXT		Extensión del Archivo
SIZE	5		N	TEXT		Tamaño del Archivo
FECHA_INGRESO	6		N	DATETIME		Fecha de Ingreso del Archivo

Fuente: Propia

Tabla B.2.1. Tabla de Archivos

Descripción

Esta tabla guarda el listado de todos los archivos que sirve para describir a manera genérica los tipos de Archivos que podemos utilizar en la Creación de Eventos.

- TAB_DISPOSITIVO

NOMBRE_COLUMNA	# COLUM	CLAVE PRIMARIA	NUL L	TIPO DATO	VALOR POR DEFECTO	DESCRIPCION
ID_DISPOSITIVO	1	1	N	INTEGER		Campo numérico secuencial
NOMBRE	2		N	TEXT		Nombre del dispositivo.
DEVICE	3		N	TEXT		UUID del dispositivo

Fuente: Propia

Tabla B.2.2. Tabla de Dispositivos

Descripción

Esta tabla guarda el listado de todos los teléfonos celulares encontrados por el dispositivo Bluetooth que sirve para tener la información UUID del Bluetooth de los dispositivos y de esta manera poder transferir los mensajes.

- TAB_EVENTO

NOMBRE_COLUMNA	# COLUM	CLAVE PRIMARIA	NUL L	TIPO DATO	VALOR POR DEFECTO	DESCRIPCION
ID_EVENTO	1	1	N	INTEGER		Campo numérico secuencial
ID_ARCHIVO	2		N	INTEGER		Clave foránea de la tabla ARCHIVO.
NOMBRE	3		N	TEXT		Nombre del evento
FECHA	4		N	DATETIME		Fecha de Creación del Evento
DESCRIPCION	5		N	TEXT		Descripción del evento
TIPO_ARCHIVO	6		N	TEXT		Tipo de Archivo

Fuente: Propia

Tabla B.2.3. Tabla de Eventos

Descripción

Esta tabla guarda el listado de todos los Eventos creados, estos eventos nos sirven para asignarlos a determinados días.

- TAB_EVENTO_DIARIO

NOMBRE_COLUMNA	# COLUM	CLAVE PRIMARIA	NUL L	TIPO DATO	VALOR POR DEFECTO	DESCRIPCION
ID_EVENTO_DIARIO	1	1	N	INTEGER		Campo numérico secuencial
ID_EVENTO	2		N	INTEGER		Clave foránea de la tabla EVENTO.
NOMBRE	3		N	TEXT		Nombre del Evento Diario
DIARIO	4		N	DATETIME		Fecha de creación.

DESCRIPCION	5		N	TEXT		Descripción del Evento Diario
POSICION	6		N	INTEGER		Posición del Evento Diario

Fuente: Propia

Tabla B.2.4. Tabla de Eventos Diarios

Descripción

Esta tabla guarda el listado de todos los eventos diarios que sirve para el envío de las campañas diarias a los teléfonos celulares.

- TAB_EVENTO_ENVIADOS

NOMBRE_COLUMNA	# COLUM	CLAVE PRIMARIA	NUL L	TIPO DATO	VALOR POR DEFECTO	DESCRIPCION
ID_EVENTO_ENVIADO	1	1	N	INTEGER		Campo numérico secuencial
ID_EVENTO	2		N	INTEGER		Clave foránea de la tabla EVENTO.
DISPOSITIVO	3		N	TEXT		UUID del dispositivo
FECHA	4		N	DATETIME		Fecha de Envío

Fuente: Propia

Tabla B.2.5. Tabla de Eventos Enviados

Descripción

Esta tabla guarda el listado de todos los eventos enviados a los teléfonos celulares y nos ayuda para la generación de los reportes y estadísticas de eventos enviados y eventos no enviados.

ANEXO C: Guía de Programación

C.1. Estándares de Programación

Un aspecto muy importante al emprender un proyecto informático es la estandarización de normas y políticas que permitan la mejor comprensión de los documentos, código de programación, implementación de la base de datos y demás recursos inmersos, entre las personas relacionadas en el desarrollo.

Se ha comprobado que las personas encargadas del mantenimiento de la aplicación pasan la mitad del tiempo tratando de interpretar bloques de código implementado por terceras erróneamente escritas, es por esta razón que este documento pretende dar los lineamientos necesarios que permitan un mejor entendimiento de la codificación implementada en el diseño y desarrollo del Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth.

- **Propósito**

El presente documento tiene como finalidad dar a conocer a los interesados los estándares de programación que regirán el desarrollo y mantenimiento de la aplicación que se desea implementar, el mismo que servirá de base para las aplicaciones futuras.

- **Descripción**

El presente documento presenta al interesado las reglas y políticas que permita estandarizar, normar el desarrollo del Proyecto “Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth” utilizando la plataforma Adobe AIR como servidor para ejecutar aplicaciones AIR y como IDE de programación Eclipse Galileo.

Para una mejor comprensión del presente documento, la persona interesada deberá tener los conocimientos necesarios sobre las siguientes tecnologías:

- Bases de Datos
- Conocimientos sobre Adobe AIR
- Lenguaje de programación Java

C.2. Estandarización del Lenguaje de Programación

Las políticas y normas que rigen la codificación de los procesos en el lenguaje de programación elegido es la siguiente:

- **Nombres de objetos**

Para la declaración de variables de objetos como formularios, cuadros de texto, checkbox y otros se seguirá la siguiente nomenclatura:

Los nombres de los objetos empezarán con el prefijo de 3 caracteres que corresponde al prefijo relacionado con el tipo de objeto, seguido del nombre que se quiere asignar a la variable iniciando la primera letra en mayúscula. Por ejemplo: btnGuardar

En el caso de que el nombre de la variable conste de 2 o más palabras el nombre del objeto empezará con un prefijo de 3 caracteres correspondiente al prefijo del objeto, seguido de la primer palabra con la primer letra en mayúscula y seguido de la segunda palabra con la primer letra en mayúscula.

Por ejemplo: btnEventosEnviados

- **Identificadores**

Un identificador es un nombre que identifica a una variable, a un método o función miembro, a una clase. Todos los lenguajes tienen ciertas reglas para componer los identificadores:

- Todos los identificadores han de comenzar con una letra, el carácter subrayado (`_`) o el carácter dólar (`$`).
- Puede incluir, pero no comenzar por un número
- No puede incluir el carácter espacio en blanco
- Distingue entre letras mayúsculas y minúsculas
- No se pueden utilizar las palabras reservadas como identificadores

Además de estas restricciones, hay ciertas convenciones que hacen que el programa sea más legible, pero que no afectan a la ejecución del programa. La primera y fundamental es la de encontrar un nombre que sea significativo, de modo que el programa sea lo más legible posible. El tiempo que se pretende ahorrar eligiendo nombres cortos y poco significativos se pierde con creces cuando se revisa el programa después de cierto tiempo.

Tipo de identificador	Convención	Ejemplo
nombre de una clase	Comienza por letra	String, Evento, Archivo

	mayúscula	
nombre de función	comienza con letra minúscula	buscarDispositivos, createPopUp, setArchivo
nombre de variable	comienza por letra minúscula	dispositivo, nombreArchivo
nombre de constante	En letras mayúsculas	PI, PATH_IMAGENES

Fuente: Propia

Tabla C.1. Tabla de tipos de identificadores

- Comentarios

Un comentario es un texto adicional que se añade al código para explicar su funcionalidad, bien a otras personas que lean el programa, o al propio autor como recordatorio. Los comentarios son una parte importante de la documentación de un programa. Los comentarios son ignorados por el compilador, por lo que no incrementan el tamaño del archivo ejecutable; se pueden por tanto, añadir libremente al código para que pueda entenderse mejor.

La programación orientada a objetos facilita mucho la lectura del código, por lo que lo que no se precisa hacer tanto uso de los comentarios como en los lenguajes estructurados. En Java existen tres tipos de comentarios

- Comentarios en una sola línea
- Comentarios de varias líneas
- Comentarios de documentación

Como podemos observar un comentario en varias líneas es un bloque de texto situado entre el símbolo de comienzo del bloque /*, y otro de terminación del mismo */. Teniendo en cuenta este hecho, los programadores diseñan comentarios como el siguiente:

```
/*-----|
| (C) Manuel Cepeda      |
| fecha: Marzo 2011      |
| programa: PrimeroApp.java |
|-----*/
```

Los comentarios de documentación es un bloque de texto situado entre el símbolo de comienzo del bloque `/**`, y otro de terminación del mismo `*/`. El programa `javadoc` utiliza estos comentarios para generar la documentación del código.

```
/** Este es el primer programa de una
serie dedicada a explicar los fundamentos del lenguaje Java */
```

Habitualmente, usaremos comentarios en una sola línea `//`, ya que no tiene el inconveniente de aprendernos los símbolos de comienzo y terminación del bloque, u olvidarnos de poner este último, dando lugar a un error en el momento de la compilación.

- Sentencias

Una sentencia es una orden que se le da al programa para realizar una tarea específica, esta puede ser: mostrar un mensaje en la pantalla, declarar una variable (para reservar espacio en memoria), inicializarla, llamar a una función, etc. Las sentencias acaban con `;` Este carácter separa una sentencia de la siguiente. Normalmente, las sentencias se ponen unas debajo de otras, aunque sentencias cortas pueden colocarse en una misma línea. He aquí algunos ejemplos de sentencias

```
int i=1;
import java.awt.*;
System.out.println("El primer programa");
rect.mover(10, 20);
```

En el lenguaje Java, los caracteres espacio en blanco se pueden emplear libremente. Como podremos ver en los sucesivos ejemplos, es muy importante para la legibilidad de un programa la colocación de unas líneas debajo de otras empleando tabuladores. El editor del IDE nos ayudará plenamente en esta tarea sin apenas percibirlo.

- Bloques de código

Un bloque de código es un grupo de sentencias que se comportan como una unidad. Un bloque de código está limitado por las llaves de apertura `{` y cierre `}`. Como ejemplos de bloques de código tenemos la definición de una clase, la definición de una función miembro, una sentencia iterativa **for**, los bloques **try ... catch**, para el tratamiento de las excepciones, etc.

- Variables

Una variable es un nombre que se asocia con una porción de la memoria del ordenador, en la que se guarda el valor asignado a dicha variable. Hay varios tipos de variables que requieren distintas cantidades de memoria para guardar datos.

Todas las variables han de declararse antes de usarlas, la declaración consiste en una sentencia en la que figura el tipo de dato y el nombre que asignamos a la variable. Una vez declarada se le podrá asignar valores. Manejaremos tres tipos de variables:

- de instancia
- de clase
- locales

Las variables de instancia o miembros dato como veremos más adelante, se usan para guardar los atributos de un objeto particular.

Las variables de clase o miembros dato estáticos son similares a las variables de instancia, con la excepción de que los valores que guardan son los mismos para todos los objetos de una determinada clase.

Las variables locales se utilizan dentro de las funciones miembro o métodos.

Delante del nombre de cada variable se ha de especificar el tipo de variable que hemos destacado en letra negrita. Las variables son uno de los elementos básicos de un programa, y se deben

- Declarar
- Inicializar
- Usar

Tipo	Descripción
Boolean	Tiene dos valores true o false .
Char	Caracteres Unicode de 16 bits. Los caracteres alfa-numéricos son los mismos que los ASCII con el bit alto puesto a 0. El intervalo de valores va desde 0 hasta 65535 (valores de 16-bits sin signo).
Byte	Tamaño 8 bits. El intervalo de valores va desde -2^7 hasta $2^7 - 1$ (-128 a 127)

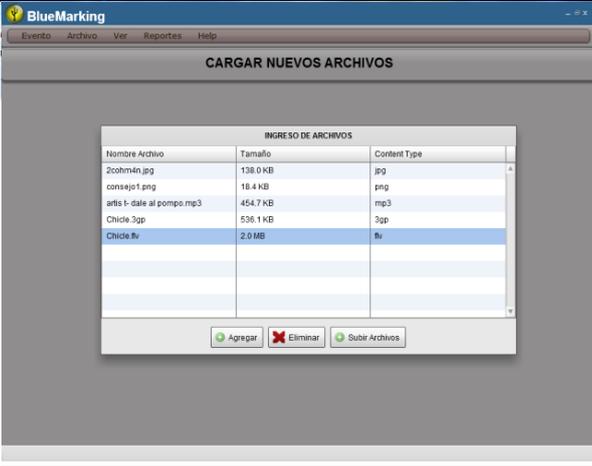
Short	Tamaño 16 bits. El intervalo de valores va desde -2^{15} hasta $2^{15}-1$ (-32768 a 32767)
Int	Tamaño 32 bits. El intervalo de valores va desde -2^{31} hasta $2^{31}-1$ (-2147483648 a 2147483647)
Long	Tamaño 64 bits. El intervalo de valores va desde -2^{63} hasta $2^{63}-1$ (-9223372036854775808 a 9223372036854775807)
Float	Tamaño 32 bits. Números en coma flotante de simple precisión. Estándar IEEE 754-1985 (de 1.40239846e-45f a 3.40282347e+38f)
Double	Tamaño 64 bits. Números en coma flotante de doble precisión. Estándar IEEE 754-1985. (de 4.94065645841246544e-324d a 1.7976931348623157e+308d.)

Fuente: Propia

Tabla C.2. Tabla de tipos de datos primitivos

ANEXO D: Prototipo de Interfaz de usuario

Toda obra de diseño conlleva la elección de los elementos básicos que la van a formar. Una composición gráfica está destinada a representar un medio de comunicación entre personas y ordenadores, este es el caso de una interfaz de usuario.

Prototipo de Pantalla 1:	Pantalla de Ingreso de Archivos
Descripción	<p>En esta pantalla nosotros vamos a ingresar los diferentes tipos de archivos, al seleccionar los archivos nos valida que no ingresemos archivos repetidos.</p> <ul style="list-style-type: none"> • Agregar.- Abre una ventana donde seleccionamos los archivos que vamos a ingresar. • Eliminar.- Eliminar los archivos seleccionados de la tabla. • Subir Archivos.- Carga los archivos de la tabla al sistema.
Caso de Uso	Caso de Uso: CU-001
	

Fuente: Propia

Tabla D.1. Prototipo interfaz, Ingreso de Archivos.

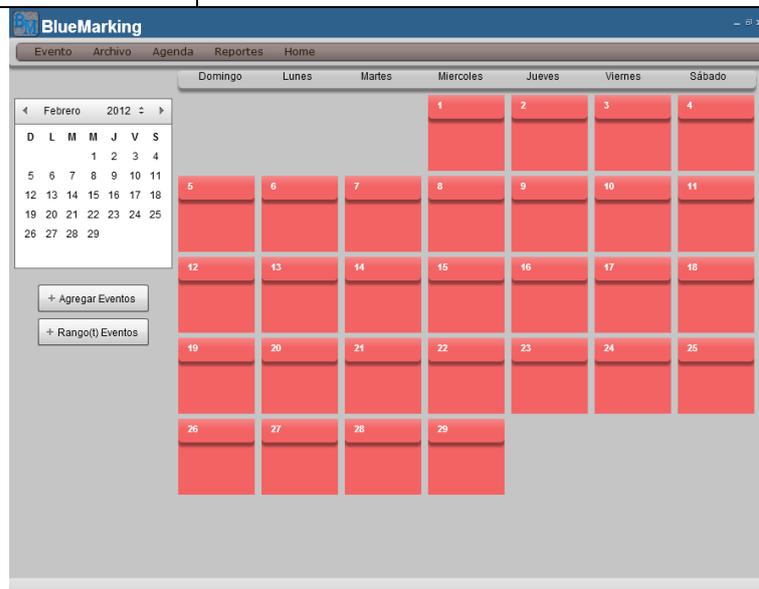
Prototipo de Pantalla 2:	Pantalla Crear Nuevo Evento
<p>Descripción</p>	<p>Esta pantalla tiene la opción de ingresar un nuevo evento (campaña), estos eventos vamos a utilizar para asignar a los diferentes días del año.</p> <p>El seleccionar el tipo de Archivo podemos visualizar los archivos antes de seleccionarlos.</p> <p>Si todos los datos son correctos nos aparece un mensaje de ingreso exitoso.</p> <ul style="list-style-type: none"> • Nombre Evento.- El nombre del nuevo evento. • Descripción.- La descripción del nuevo evento • Tipo Archivos.- Seleccionamos el archivo del nuevo evento.
<p>Caso de Uso</p>	<p>Caso de Uso: CU-002</p>



Fuente: Propia

Tabla D.2. Prototipo interfaz, Crear Nuevo Evento.

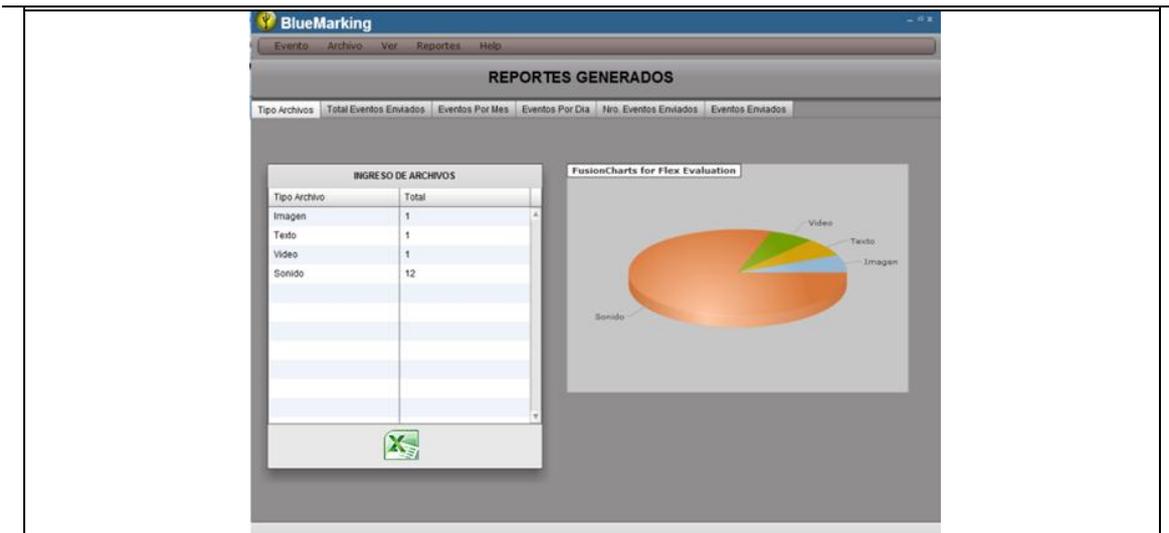
Prototipo de Pantalla 3:	Pantalla Asignar Eventos Diarios
<p>Descripción</p>	<p>Esta pantalla tiene la opción de ingresar un nuevo evento diario (campaña), a un X días del año.</p> <p>Seleccionamos el día donde vamos asignar el nuevo evento diario, Aquí tenemos la posibilidad de visualizar los evento ya creados.</p> <p>Si todos los datos son correctos se incrementa el número de eventos en el día seleccionado.</p> <ul style="list-style-type: none"> • Clic en un día.- ingresamos nuevos eventos diarios y visualizar los eventos creados. • Agregar eventos.- Permite crear eventos diarios en un rango de fechas.
<p>Caso de Uso</p>	<p>Caso de Uso: CU-003</p>



Fuente: Propia

Tabla D.3. Prototipo interfaz, Asignar Eventos Diarios.

Prototipo de Pantalla 4:	Pantalla Reportes y Estadísticas
Descripción	<p>Esta pantalla tiene todos los reportes necesarios para estar al tanto de los que sucede con el sistema y poder tener las estadísticas actuales.</p> <p>Todos los reportes se exportan a un Excel.</p> <ul style="list-style-type: none"> • Tipo Archivos.- Reportes de los tipos de archivos. • Total Eventos Enviados.- Reportes de eventos enviados y no enviados. • Eventos Por Mes.- Reportes de los eventos enviados por mes. • Eventos Por Día.- Reportes de los eventos enviados por día. • Nro. Eventos Enviados.- Reportes de los eventos enviados agrupados por tipo de archivo. • Eventos Enviados.- Reportes de los eventos enviados.
Caso de Uso	Caso de Uso: CU-004



Fuente: Propia

Tabla D.4. Prototipo interfaz, Reportes y Estadísticas.

ANEXO E: Manual de Usuario

Introducción

Olvídese de folletos, carteles, folletos. Pon tu mensaje en la palma de su mano a los clientes. Su empresa puede ser de los primeros en adoptar esta tecnología, y unirse a gigantes como Land Rover, Pepsi y Nokia que han llevado a cabo campañas exitosas. Envía imágenes, audio, clips de películas directamente a tu cliente potencial.

Publicidad por Bluetooth es una instalación inalámbrica local de distribución dentro de un área de proximidad, donde la información puede ser enviada y recibida por los clientes potenciales de forma gratuita.

Que es BlueMarking?

BlueMarking es una solución para enviar contenido utilizando campañas (eventos) de Marketing de proximidad Bluetooth. Con este software se promociona su negocio o empresa, enviando todo tipo de información. Es muy sencillo de trabajar con este software, necesita tener los archivos: BlueMarking.air y BlueMarking.jar, Instalarlos y comenzar a utilizarlo como una gran herramienta para transferir publicidad y propaganda a los posibles clientes que tengan un teléfono celular con el Bluetooth encendido, y sin ningún costo de envío.

Que es Marketing de Proximidad Bluetooth?

"El Marketing de Proximidad consiste en distribuir información de interés o contenido publicitario a través de mensajes, ya sean multimedia, imágenes, texto y/o música a cualquier dispositivo móvil, equipado con Bluetooth, dicha transmisión de información será recibida por los usuarios dentro de un espacio determinado que va de 10 a 100 metros aproximadamente."^[19]

Como trabaja BlueMarketing

BlueMarketing, una nueva y emocionante manera de promocionar sus servicios y productos por medio de la tecnología inalámbrica Bluetooth. Los estudios muestran que un porcentaje muy elevado de posibles clientes ahora están llevando teléfonos celulares que son compatibles con Bluetooth (hasta 80%). BlueMarketing se basa en este hecho para trabajar como un empleado dedicado a su negocio. Así mismo, mantendrá la supervisión de su área de negocio a la espera de un posible nuevo cliente y cuando este cliente se acerca, se le ofrecen para recibir una promoción de imagen, video, audio y texto.

En otras palabras, si usted tiene un teléfono celular en la proximidad del rango que se encuentra emitiendo una campaña de Marketing, será capaz de recibir los contenidos, mensajes o anuncios. La clave es tener su teléfono celular con la conexión Bluetooth encendida y en modo "Detectable". Esto permite que todos los anuncios y ofertas lleguen a su teléfono, preguntándole si desea recibir el contenido.

Esta forma de publicidad ha estado en uso desde hace ya algún tiempo en Estados Unidos y Europa. Hay docenas de compañías que ya están en pleno apogeo, promoviendo la comercialización de sus productos y servicios a través del Marketing por Bluetooth.

[19] **Definición Marketing de Proximidad Bluetooth. "Marketing de Proximidad",**
<http://www.mobimonto.com/publicidad-bluetooth-marketing-proximidad.php>

El alcance y la accesibilidad de los contenidos de Marketing por Bluetooth dependen de su equipo, muchos no requieren de “emparejamiento” para su recepción. El rango medio de alcance para las campañas de Marketing es de un diámetro de 100 metros, aunque con los avances en la tecnología, esta distancia es cada vez mayor.

Características BlueMarketing

BlueMarketing tiene muchas características que no se puede encontrar en ningún otro producto en el mercado, entre las más importantes tenemos.

1. Esta desarrollado con una interfaz de usuario muy amigable y fácil de manejar.
2. Se puede utilizar para anunciar los servicios y productos de su negocio 24 / 7 durante todo el año de forma gratuita.
3. BlueMarketing pueden publicar en un rango de hasta 100 metros de diámetro. (Dependiendo de su hardware de Bluetooth).
4. No requiere ningún hardware especial.
5. Funciona bajo Microsoft Windows, Linux, etc. Esto significa que cualquier equipo puede ser utilizado para publicar sus anuncios.
6. El sistema se puede minimizar lo que significa que usted puede utilizar el mismo equipo para otros fines.
7. No es necesario ningún software adicional en los teléfonos celulares.
8. Puede enviar diferentes tipos de anuncios que incluyen:
 - Texto
 - Imágenes fijas
 - Imágenes animadas
 - Archivos de audio
 - Clips de vídeo

Requisitos mínimos:

BlueMarketing está diseñado para trabajar con Microsoft Windows, Linux, etc.

Hay dos pasos sencillos para empezar a trabajar con BlueMarketing:

1. Instalar el hardware adecuado:

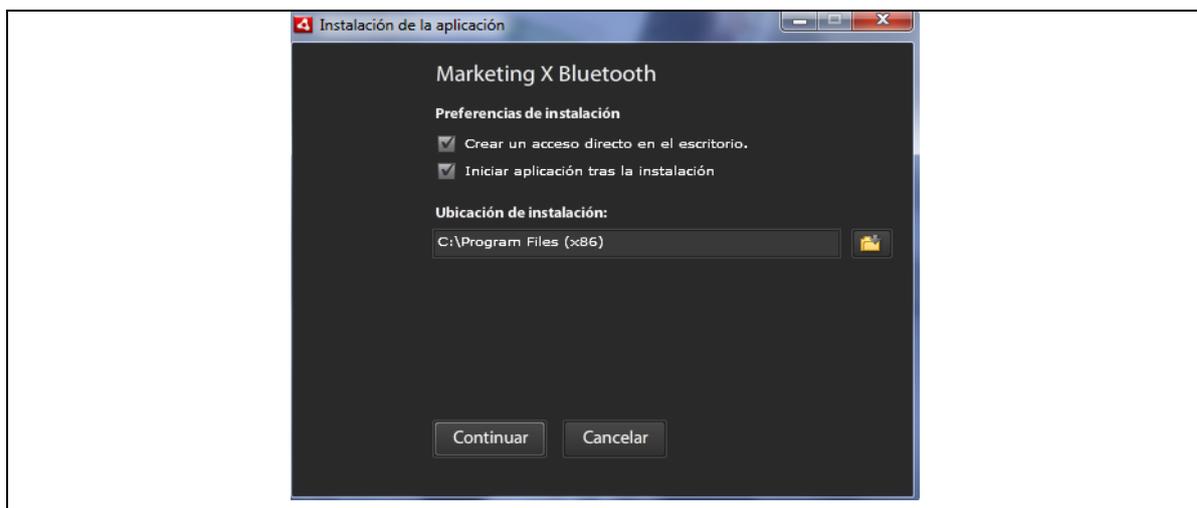
Para que su computadora pueda enviar solicitudes de Bluetooth y de datos, tiene que ser compatible con Bluetooth. Algunos ordenadores y portátiles modernos vienen con tecnología Bluetooth integrada y lista para funcionar. Si usted no tiene un ordenador con Bluetooth, puede corregir fácilmente. Todo lo que necesitas hacer es conseguir un adaptador Bluetooth que normalmente cuesta alrededor de \$ 10. Ahora,

2. Instalación de BlueMarking

Debe Tener los archivos BlueMarking.air y BlueMarking.jar

E.1. Instalación de BlueMarking - AIR

Doble clic en el archivo BlueMarking.air. Damos clic en instalar y seleccionamos la ubicación de la instalación.



Fuente: Propia

Figura E.1. BlueMarking, ubicación de instalación.

A continuación damos clic en Continuar, y se comienza a instalar nuestra aplicación BlueMarking.air. Al completar la instalación se abre la aplicación. La siguiente figura muestra la aplicación instalada.

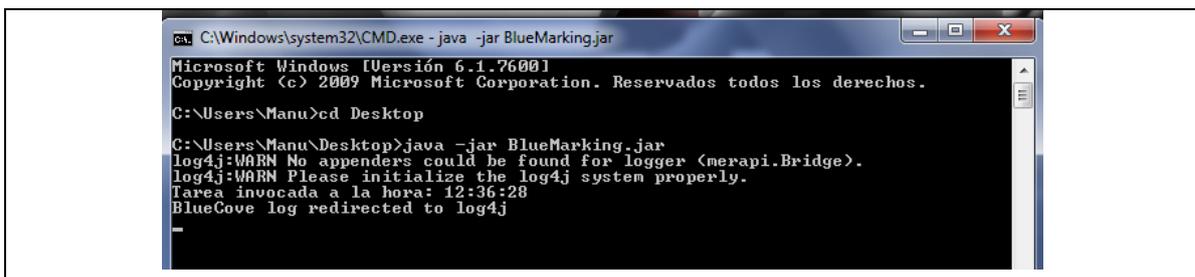


Fuente: Propia

Figura E.2. BlueMarking, aplicación instalada.

E.2. Instalación de BlueMarking - Java

Para la ejecución de BlueMarking.jar debemos abrir una consola de comando y navegar hasta la ubicación del mismo, y ejecutamos el siguiente comando: `java -jar BlueMarking.jar`. La siguiente figura muestra la ejecución de la aplicación .jar



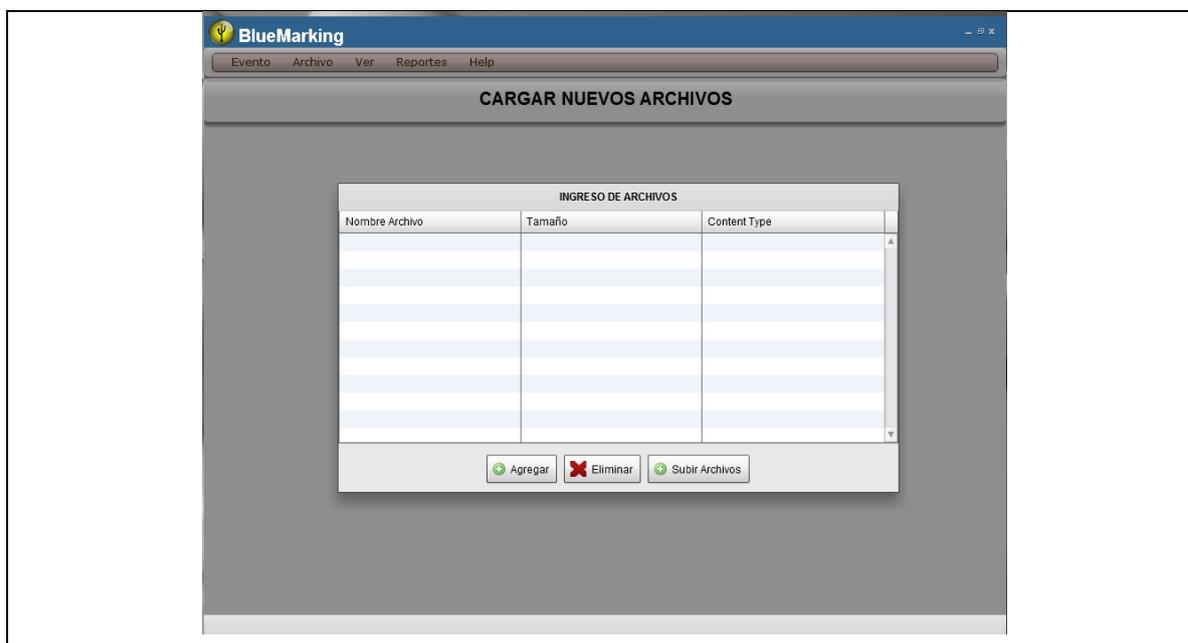
Fuente: Propia

Figura E.3. BlueMarking, ejecutar el archivo BlueMarking.jar.

Una vez instaladas las dos aplicaciones estamos listos para comenzar a interactuar con nuestra aplicación.

Cargar de Archivos

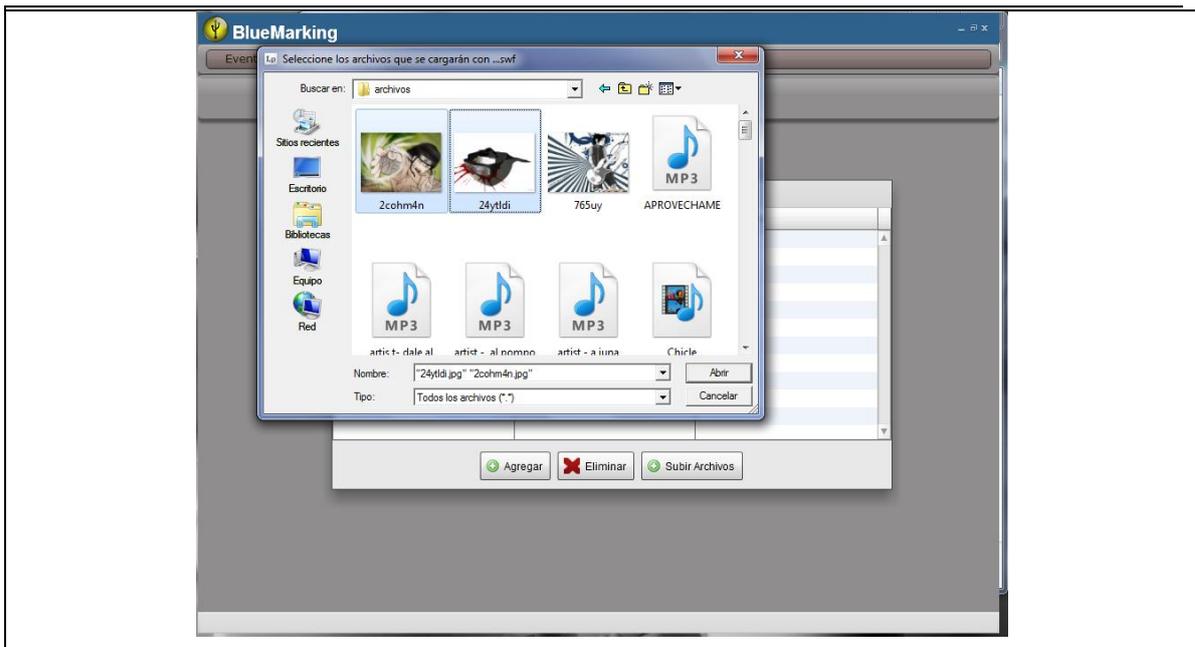
Para realizar la carga de archivos al sistemas dar clic en Archivo / Agregar, y damos clic en Agregar. La siguiente figura muestra la pantalla de Carga de Archivos.



Fuente: Propia

Figura E.4. BlueMarking, pantalla carga de Archivos.

A continuación, seleccionamos el o los archivos que vamos a cargar, y damos clic agregar y podemos ver los archivos que vamos a cargar, luego damos clic en subir Archivos. La siguiente figura muestra la selección de archivos.



Fuente: Propia

Figura E.5. BlueMarking, selección de Archivos.

NOTA. Cuando cargamos archivos de video tenemos que asegurarnos de cargar los archivos .3gp y .flv.

Crear un nuevo evento

Para crear un nuevo evento dar clic en Evento / Nuevo. Ingresamos los siguientes datos:

- Nombre del Evento.- Para poder identificar los eventos
- Descripción del evento.- Breve descripción del evento creado.
- Tipo de Archivo.- Tipo de Archivo que se va a enviar, este puede ser de Imagen, Video, Sonido y Texto.

Al dar clic en el tipo de Archivo, muestra todos los archivos que cargamos anteriormente, agrupando por tipo. La siguiente figura muestra la selección de archivos.



Fuente: Propia

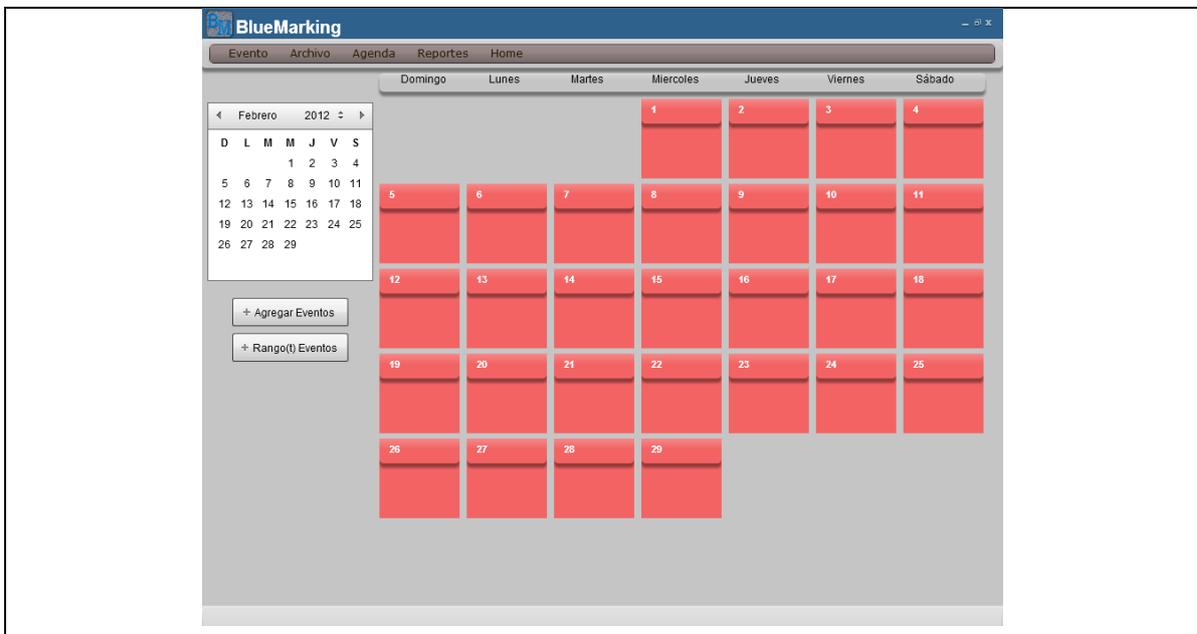
Figura E.6. BlueMarking, tipos de Archivos.

Al completar con el ingreso de los datos dar clic en el botón **Nuevo Evento**. Si el ingreso no tuvo ningún problema nos aparece el siguiente mensaje **“Ingreso Nuevo Evento”**.

Agregar y ver eventos diarios

Para agregar y ver los eventos diario hemos utilizado una agenda dinámica la cual nos dibuja el mes actual.

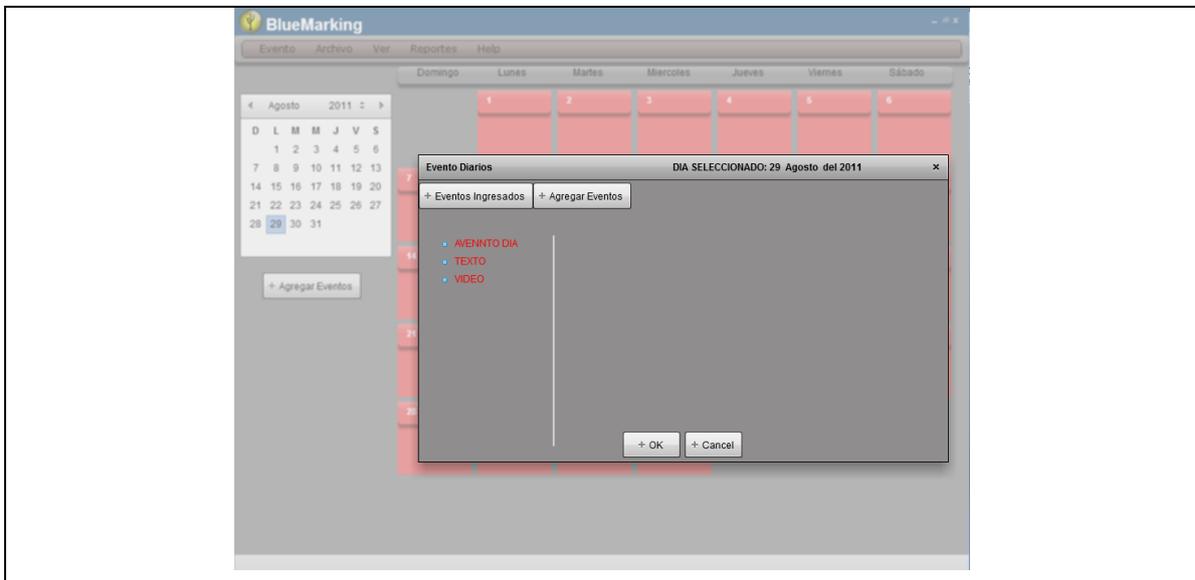
Cada cuadro representa un día del mes actual. Y en la parte izquierda tenemos un calendario para navegar en los diferentes meses del año, también tenemos el botón **“Agregar Evento”**. La siguiente figura muestra la agenda dinámica.



Fuente: Propia

Figura E.7. BlueMarking, Agenda dinámica.

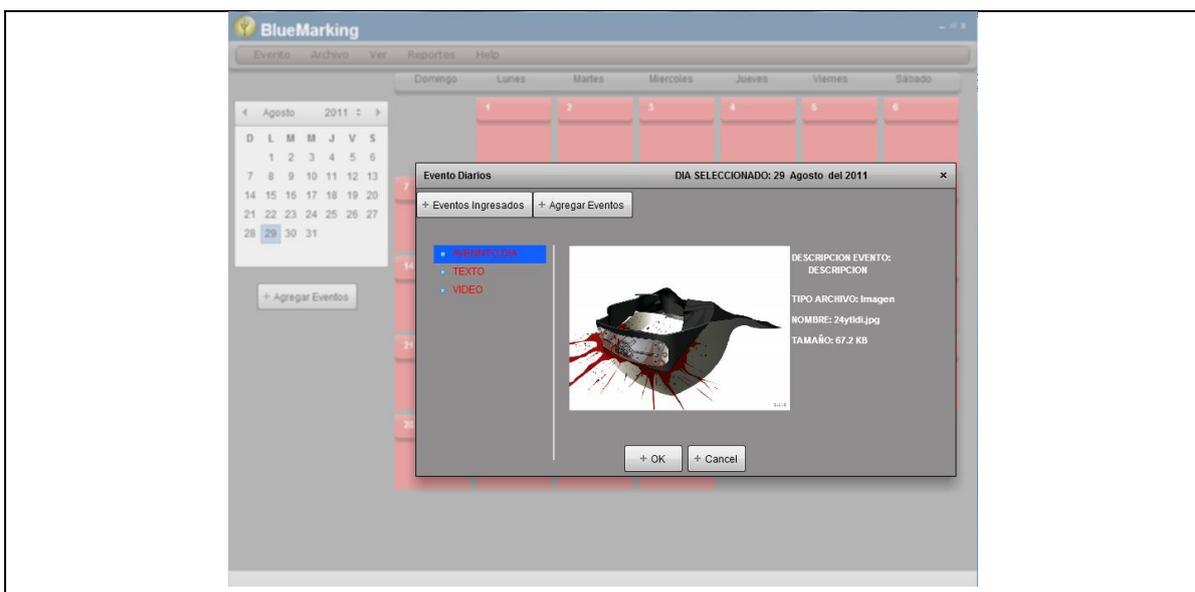
En esta agenda dinámica, se dibujan los días del mes actual o del mes seleccionado, cada día tiene un evento clic, al dar clic en un día nos muestra la siguiente pantalla donde podemos visualizar los eventos que tiene el día y la posibilidad de agregar nuevos eventos.



Fuente: Propia

Figura E.8. BlueMarking, popUp para cada día.

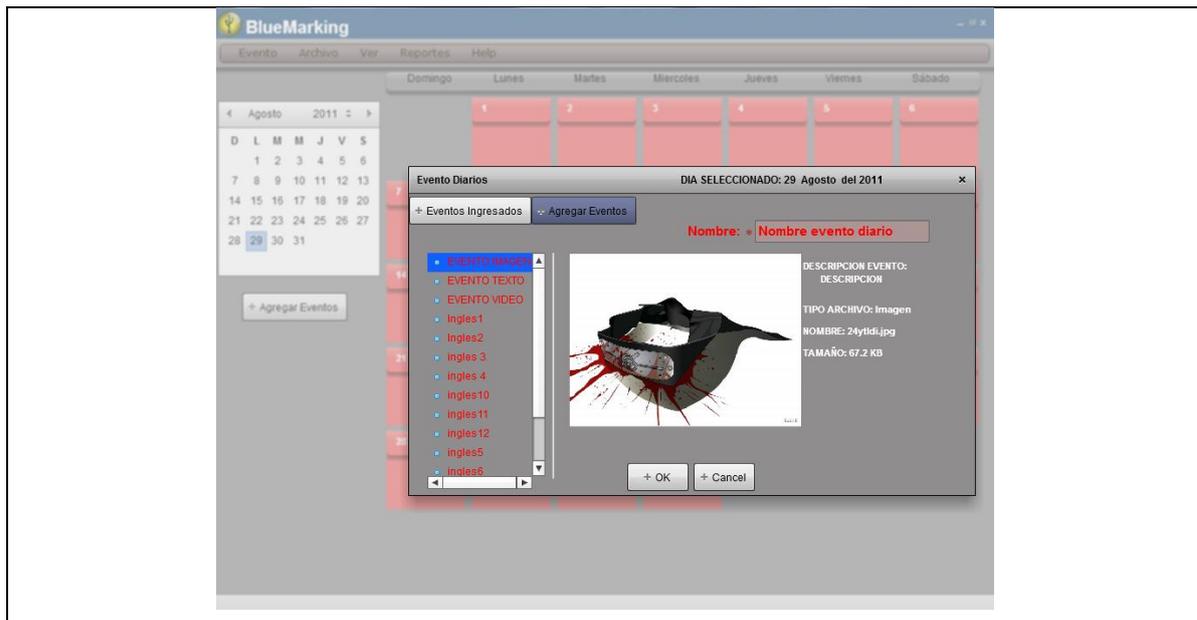
Al dar clic en el botón **Eventos Ingresados**, en la parte izquierda podemos ver todos los eventos o campañas agregados para el día seleccionado, al dar clic en uno de ellos muestra el tipo de evento con toda la descripción del evento. La siguiente figura muestra los eventos ingresados para un día.



Fuente: Propia

Figura E.9. BlueMarking, eventos ingresados para un día.

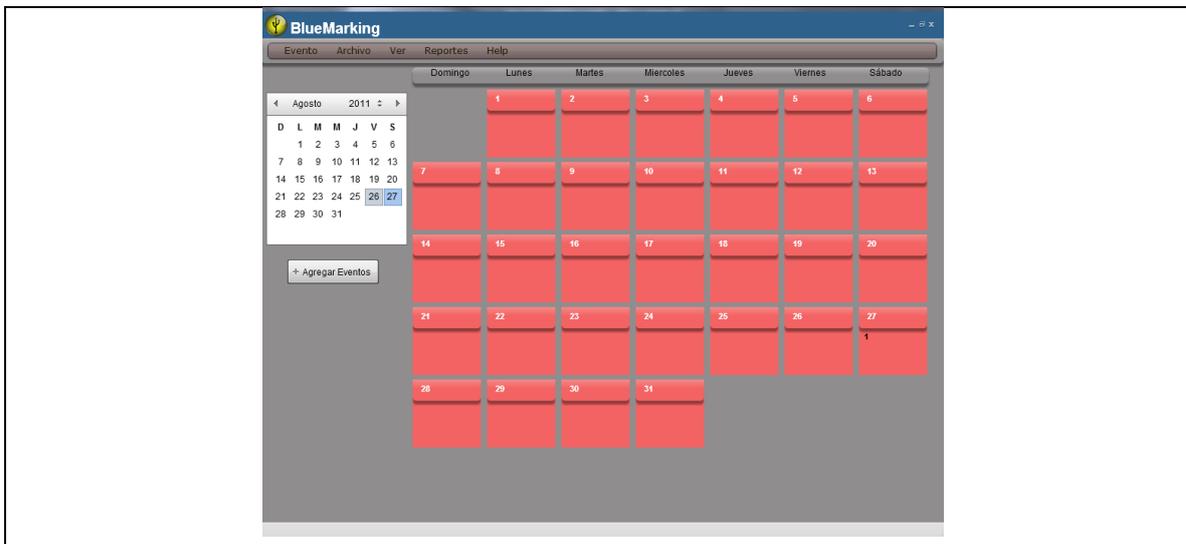
Al dar clic en el botón **Agregar Eventos**, en la parte izquierda podemos ver todos los eventos creados y son los que podemos agregar al día, al dar clic en uno de ellos muestra el tipo de evento con toda la descripción del evento y con un campo de texto para ingresar el nombre del nuevo evento diario. La siguiente figura muestra la asignación de eventos a un día.



Fuente: Propia

Figura E.10. BlueMarking, asignar eventos para un día.

Al dar clic en OK se guarda el nuevo evento diaria para el día seleccionamos. Y por cada evento creado se incrementa el número que aparece en los días de la agenda en caso de que tenga 1 o más evento diario. La siguiente figura muestra el incremento de evento a un día.

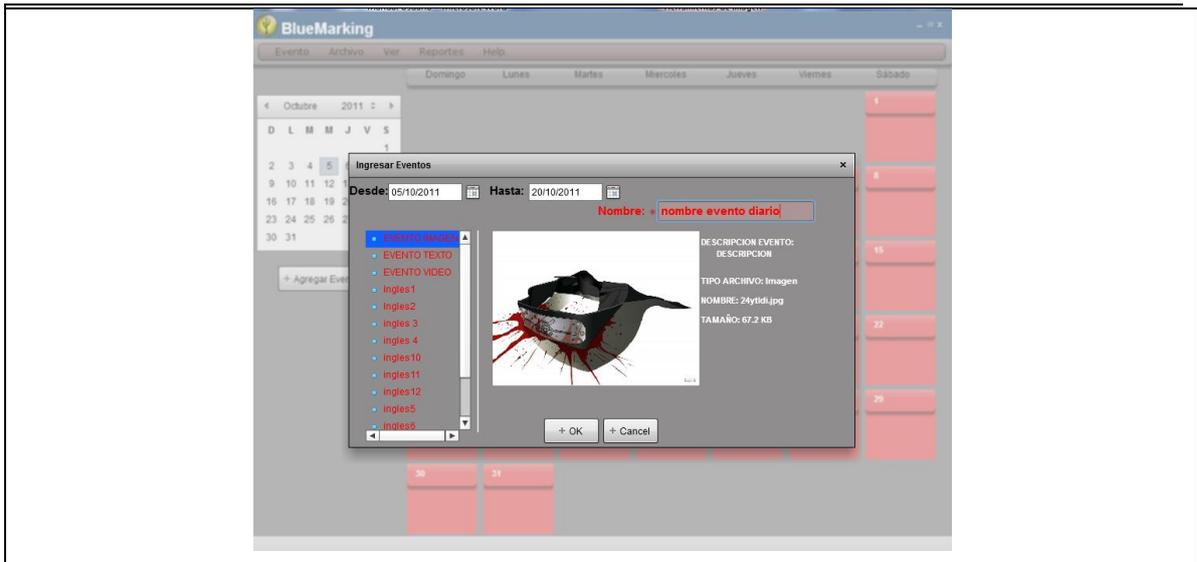


Fuente: Propia

Figura E.11. BlueMarking, incremento de evento a un día.

Al dar clic en el botón **Agregar Eventos**, agrega nuevos eventos diarios pero en un rango de fechas para que se nos haga más fácil crear eventos diarios.

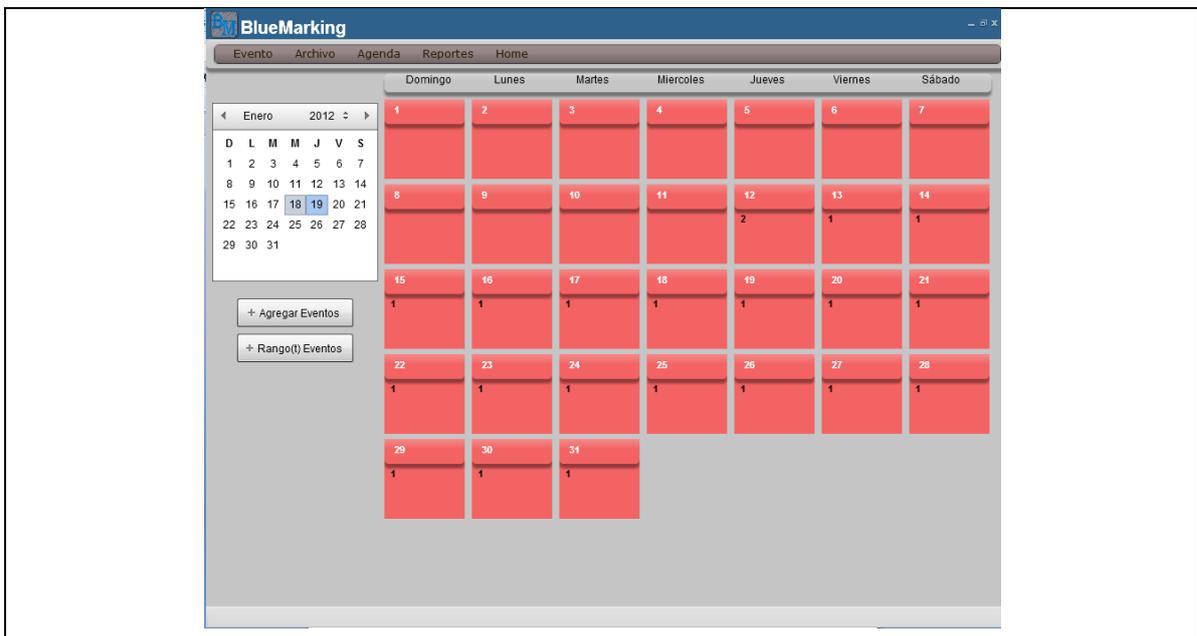
En la parte superior ingresamos el rango de fechas, es decir, la fecha desde y hasta, donde se va a crear el nuevo evento diario, todos los días entre este rango se van agregar el evento, en la parte izquierda están los eventos agregados al dar clic en uno de estos eventos se despliega el tipo de evento y la información de este evento y con un campo de texto para ingresar el nombre del nuevo evento diario. La siguiente figura muestra la asignar de un Evento en un rango de fechas.



Fuente: Propia

Figura E.12. BlueMarking, asignar eventos en un rango de fechas.

Al dar clic en OK se guarda el nuevo evento diario para el rango de días seleccionados. Y por cada evento creado se incrementa el número que aparece en los días de la agenda, en caso de que tenga 1 o más evento diario. Como se muestra en la siguiente pantalla.



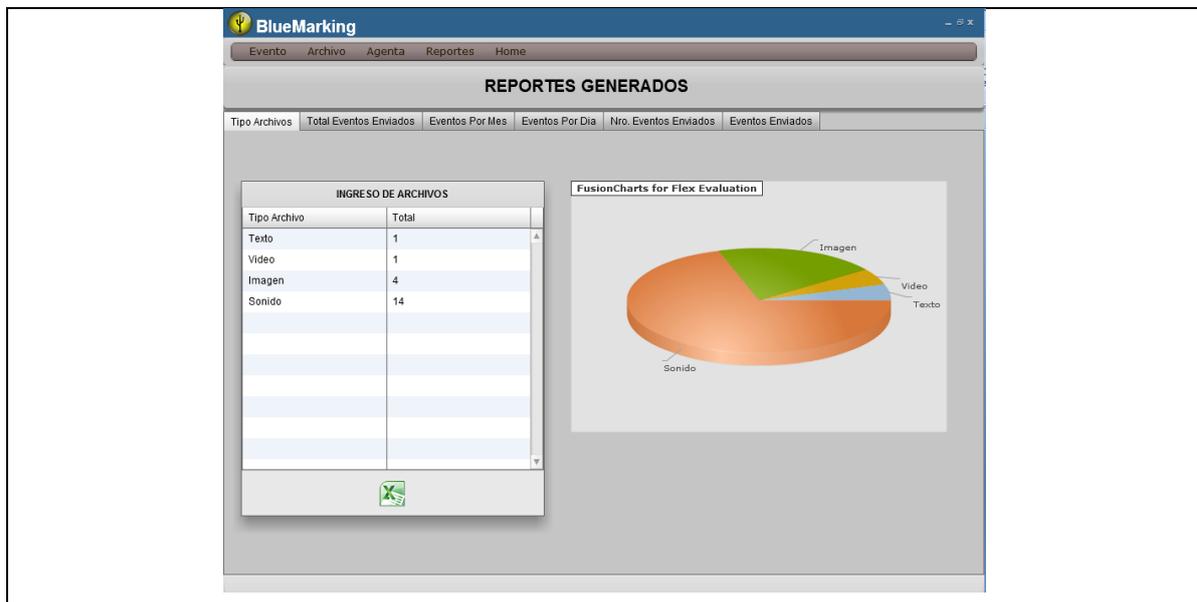
Fuente: Propia

Figura E.13. BlueMarking, incremento de evento a un rango de fechas.

Reportes

Para poder ver todos los reportes, dar clic en Reporte / Reportes Eventos. Aquí por defecto esta seleccionado el **Tipo Archivos** el cual despliega la información con el total de todos

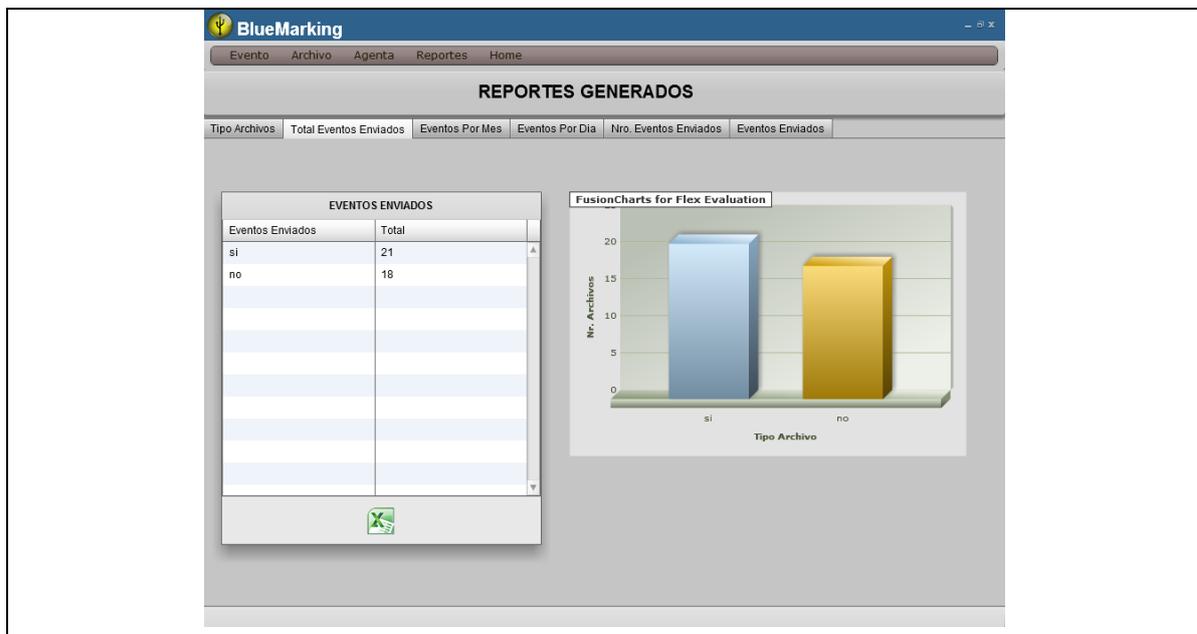
los tipos de archivos que se han creado en los eventos. La siguiente figura muestra el reporte por tipo de Archivos.



Fuente: Propia

Figura E.14. BlueMarking, reporte por tipo de Archivos.

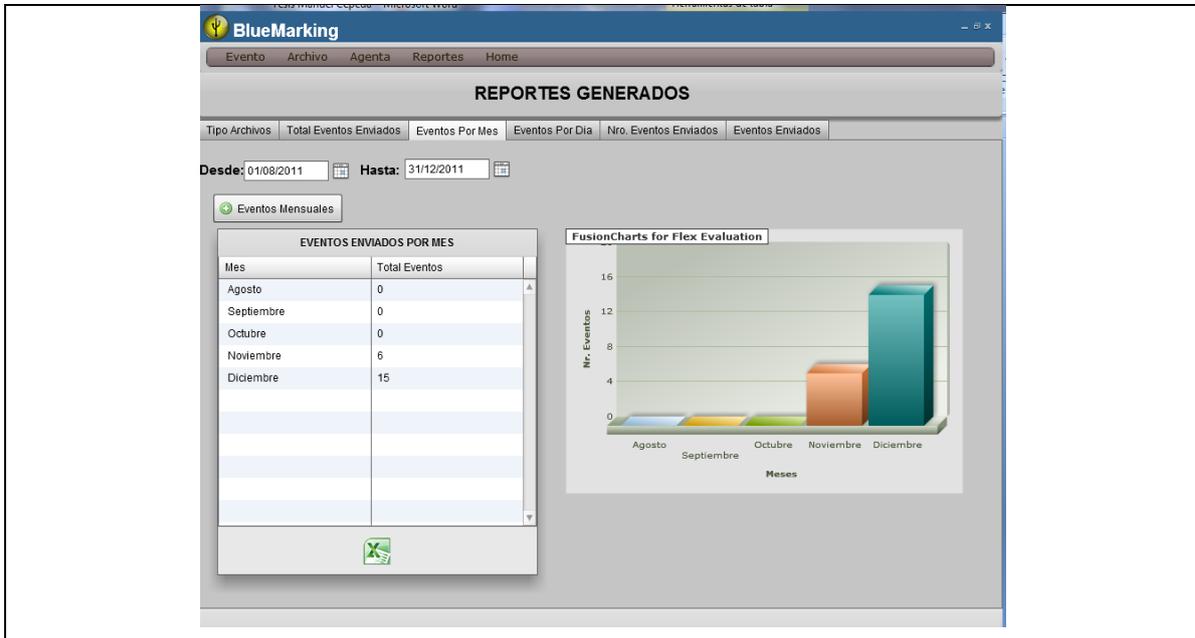
Al dar clic en la pestaña **Total Eventos Enviados** despliega el total de eventos enviados y el total de eventos no enviados. La siguiente figura muestra el reporte de los Eventos enviados los que no se enviaron.



Fuente: Propia

Figura E.15. BlueMarking, reporte de eventos enviados y no enviados.

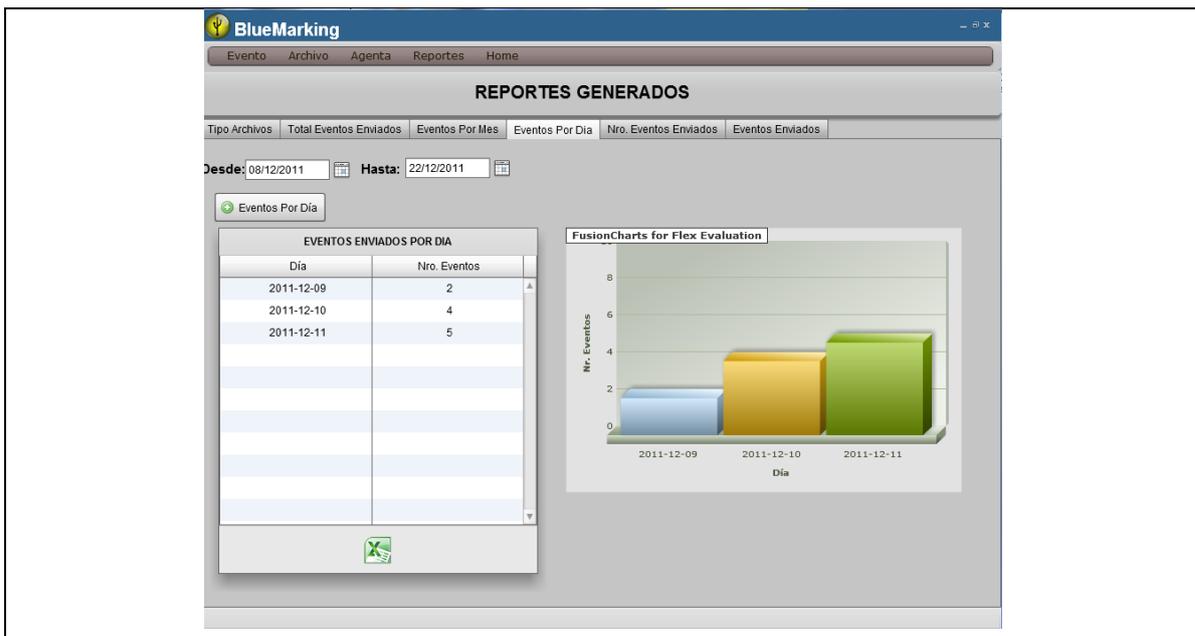
Al dar clic en la pestaña **Eventos Por Mes** debemos ingresar un rango de fechas y damos clic en el botón **Eventos Mensuales**, despliega el total de eventos enviados por mes. La siguiente figura muestra el reporte de eventos enviados por mes.



Fuente: Propia

Figura E.16. BlueMarking, reporte de eventos enviados por mes.

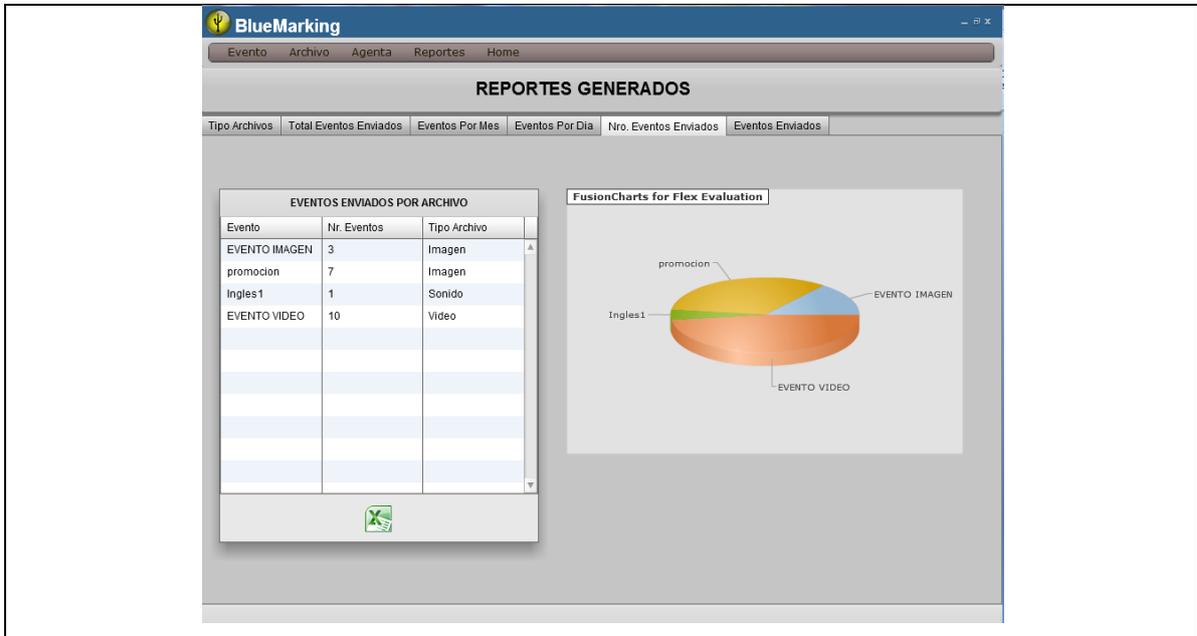
Al dar clic en la pestaña **Eventos Por Días** debemos ingresar un rango de fechas de un determinado mes y damos clic en el botón **Eventos Por Día**, despliega el total de eventos enviados por día. La siguiente figura muestra el reporte de eventos enviados por día.



Fuente: Propia

Figura E.17. BlueMarking, reporte de eventos enviados por día.

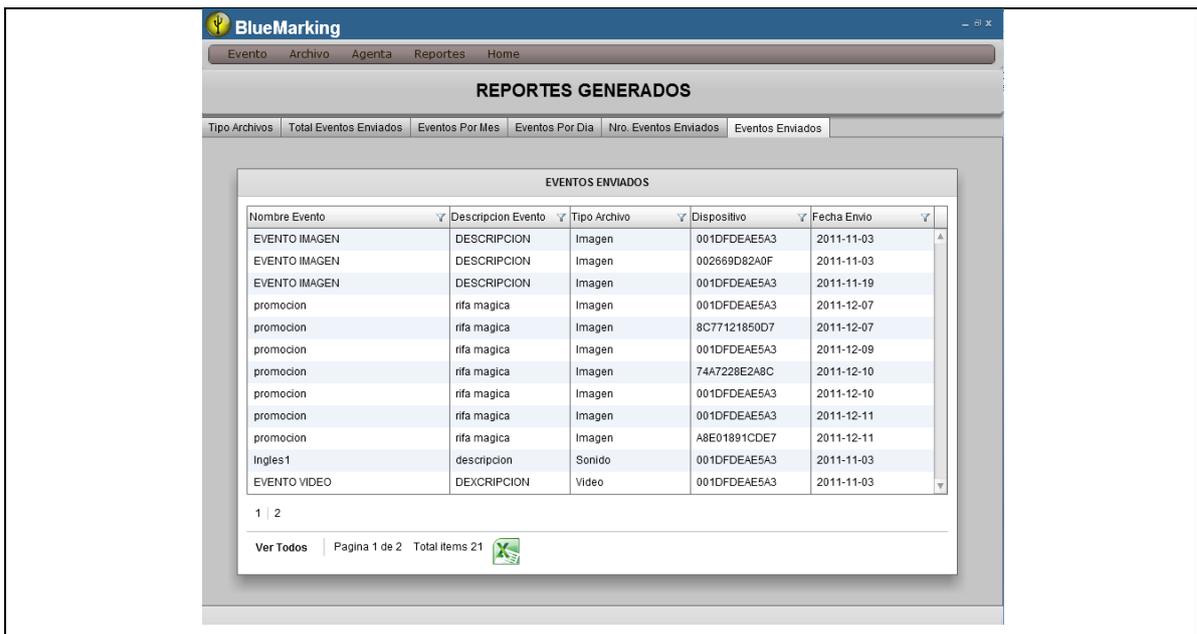
Al dar clic en la pestaña **Nro. Eventos Enviados** despliega el total de eventos enviados agrupados por el tipo de Archivo. La siguiente figura muestra el reporte el número de eventos enviados agrupados por el tipo de Archivo.



Fuente: Propia

Figura E.18. BlueMarking, reporte de eventos enviados agrupados por tipo de Archivo.

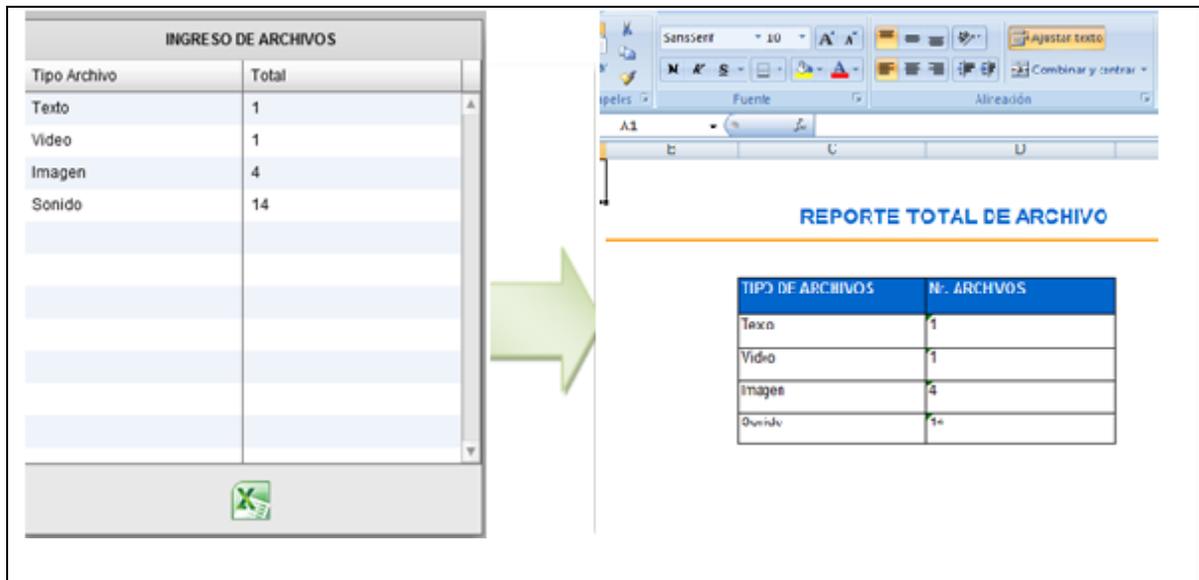
Al dar clic en la pestaña **Eventos Enviados** despliega el total de eventos enviados. La siguiente figura muestra el reporte total de los eventos enviados.



Fuente: Propia

Figura E.19. BlueMarking, reporte total de eventos enviados.

NOTA. Todas las pestañas despliegan los datos en una tabla y cada una tiene en la parte inferior un botón Exportar, el cual nos exporta los datos a un archivo EXCEL. La siguiente figura muestra la exportación de los datos de la tabla a EXCEL.



Fuente: Propia

Figura E.20. BlueMarking, exportar los datos de la tabla a Excel.

ANEXO F: Manuel Técnico

Introducción

La finalidad de este manual técnico es la de proporcionar al lector la lógica con la que se ha desarrollado el presente Sistema Informático para la promoción y publicidad de los productos y servicios de la Cooperativa Atuntaqui utilizando Bluetooth, la cual se sabe que es propia de cada programador, por lo que se considera necesario ser documentada.

Aclarando que este manual no pretende ser un curso de aprendizaje de cada una de las herramientas empleadas para el desarrollo del sistema BlueMarking, sino documentar su aplicación en el desarrollo del programa. Para un mayor detalla acerca de cada una de las herramientas utilizadas, y su forma de operación y aplicación, se recomienda consultas los manuales respectivos de cada una de las herramientas empleadas en el desarrollo de la aplicación.

Objetivo

- Proporcionar una guía para el lector, del desarrollo de la aplicación.

Instalación Adobe AIR

Para la ejecución e instalación de nuestra aplicación necesitamos tener instalado Adobe AIR. Para instalar o actualizar el motor de ejecución, el usuario debe tener privilegios administrativos para el equipo.

Instalación del motor de ejecución en un ordenador con Windows

1. Descargue el archivo de instalación del motor de ejecución en <http://get.adobe.com/es/air>.

2. Haga doble clic en el archivo de instalación del motor de ejecución, damos clic en Ejecutar, luego aceptamos los términos y condiciones. La siguiente figura muestra la pantalla de los términos del producto.



Fuente: Propia

Figura F.1. Términos y condiciones de Adobe AIR.

3. A continuación comienza a instalarse Adobe AIR, luego damos clic en finalizar y la instalación se ha completado.

Instalación del motor de ejecución en un ordenador con Mac

1. Descargue el archivo de instalación del motor de ejecución en <http://get.adobe.com/es/air>.
2. Haga doble clic en el archivo de instalación del motor de ejecución.
3. Siga las indicaciones que aparecen en la ventana de instalación para llevar a cabo la instalación.
4. Si el instalador presenta una ventana de autenticación, escriba el nombre de usuario y la contraseña que utiliza para Mac OS.

Instalación del motor de ejecución en un ordenador con Linux

Nota: de momento, AIR 2.7 y posterior no se admite en Linux. Las aplicaciones de AIR desarrolladas para Linux deben seguir utilizando el SDK de AIR 2.6.

Uso del instalador binario:

1. Localice el archivo binario de instalación en la siguiente dirección http://kb2.adobe.com/cps/853/cpsid_85304.html y realice la descarga.
2. Establezca los permisos de archivo para que se pueda ejecutar la aplicación de instalación: Desde una línea de comandos, se pueden establecer los permisos de archivo con:

```
chmod +x AdobeAIRInstaller.bin
```

Algunas versiones de Linux permiten establecer permisos de archivo en el cuadro de diálogo de propiedades que se abre mediante un menú contextual.

3. Ejecute el instalador desde la línea de comandos o haciendo doble clic en el archivo de instalación.
4. Siga las indicaciones que aparecen en la ventana de instalación para llevar a cabo la instalación.

INSTALACION DE ADOBE FLASH BUILDER 4 Plugin

Como IDE de programación hemos utilizado Adobe Flash Builder con plugin de Eclipse Galileo en un ambiente Windows, en mi caso lo hice en Windows 7.

Descargamos FlashBuilder_4_Plugin_LS10.exe de la página oficial de Adobe, le damos doble clic, si estas en windows vista o superior, hay que dar permisos. Se prepara para la instalación y carga todos los paquetes necesarios.

A continuación, seleccionamos el idioma, nosotros escogimos English para el lenguaje del instalador y le damos clic en OK.



Fuente: Propia

Figura F.2. Selección del Idioma de Adobe Flash Builder.

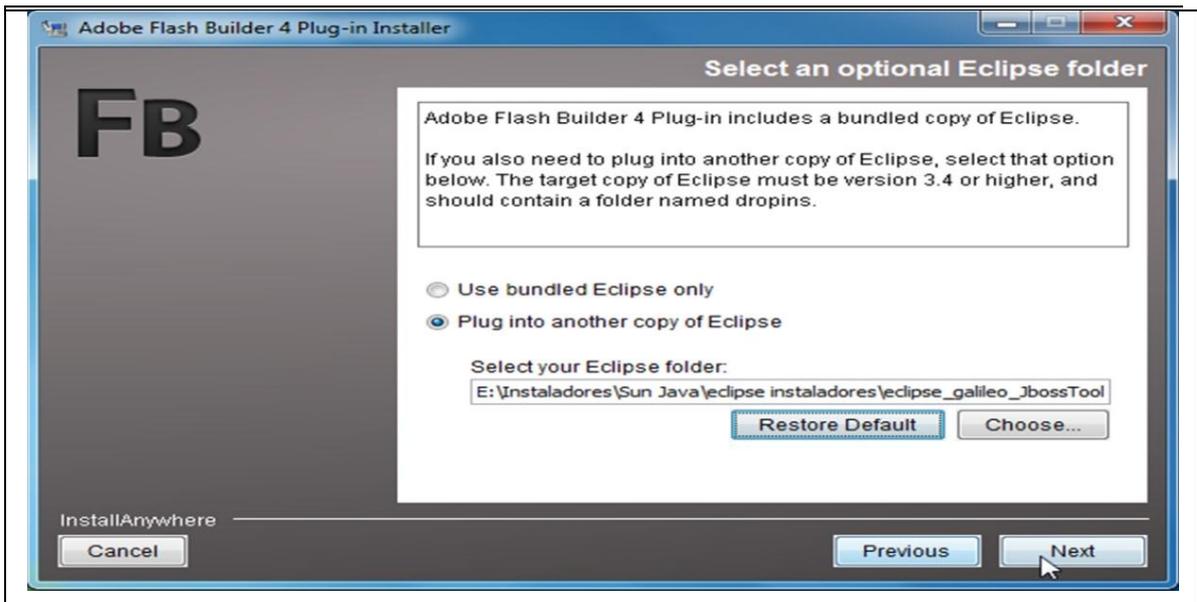
Luego damos clic en Next, y nos aparece la pantalla de términos y condiciones de la licencia de Adobe.



Fuente: Propia

Figura F.3. Términos y condiciones de Adobe Flash Builder.

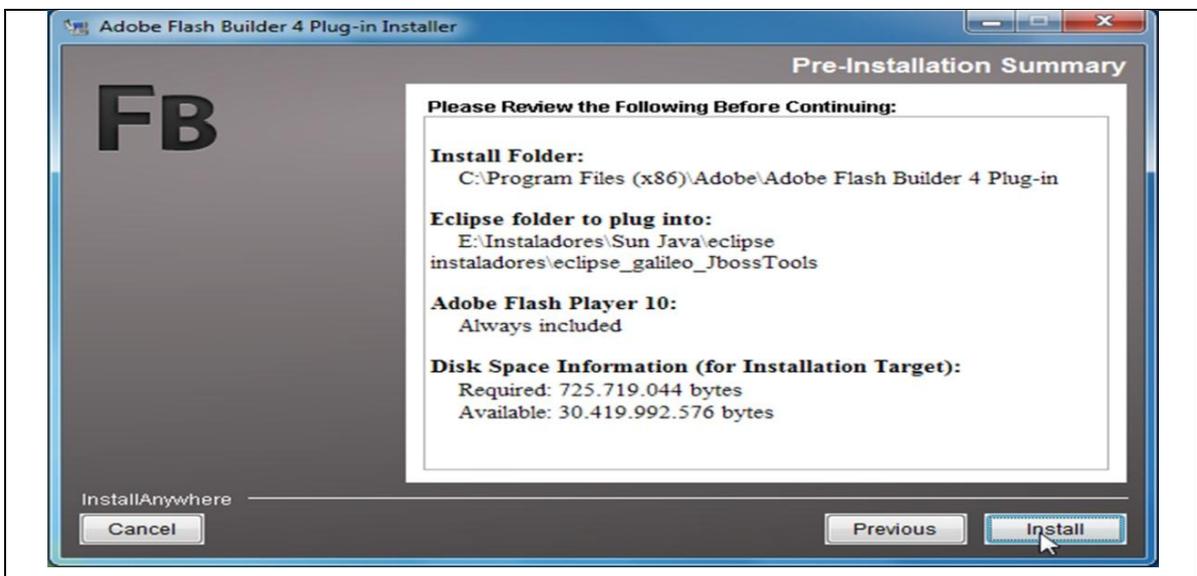
A continuación seleccionamos la ubicación donde vamos a instalar el plugin, y damos clic en Next. En la siguiente pantalla seleccionamos la opción Plugin into another copy of Eclipse, y seleccionamos la ubicación de la carpeta raíz de Eclipse.



Fuente: Propia

Figura F.4. Seleccionamos la ubicación de Eclipse.

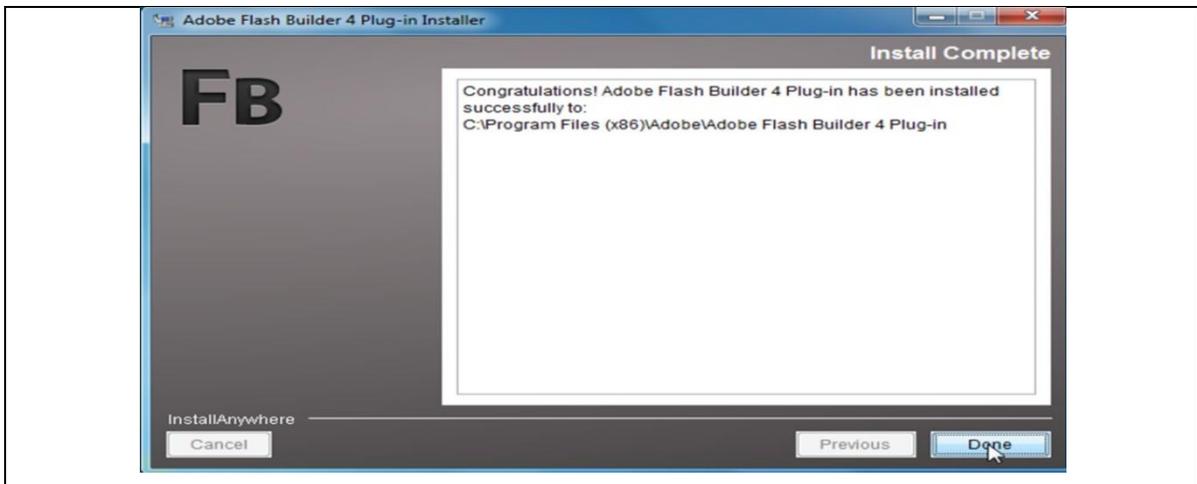
Nos muestra un resumen de las opciones y chunches por instalar, a continuación le damos clic en INSTALL:



Fuente: Propia

Figura F.5. Resumen de las opciones por instalar.

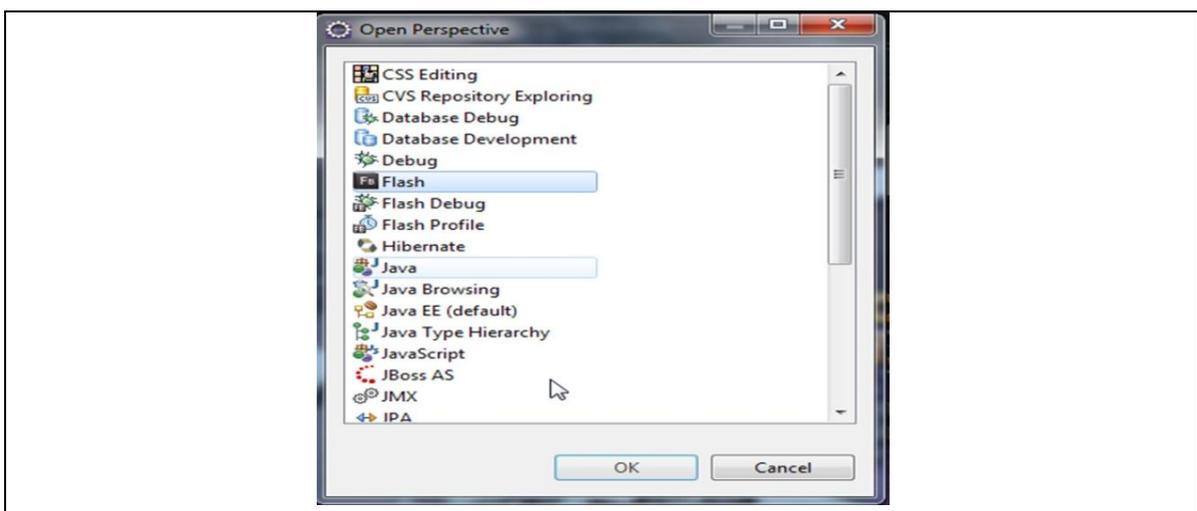
Empieza a correr el “progress bar” con la información de lo que va instalando, esperamos a que se instalen todos los paquetes necesarios. Al terminar la instalación nos aparece un apantalla, que la instalación fue completada.



Fuente: Propia

Figura F.6. Resumen de la instalación.

Nos ubicamos en la carpeta donde esta el eclipse y damos doble clic en el icono de eclipse.exe, se nos abre el IDE de Eclipse. Para obtener la perspectiva de Adobe damos clic en Windows / Open Perspective / Other y seleccionamos Flash.



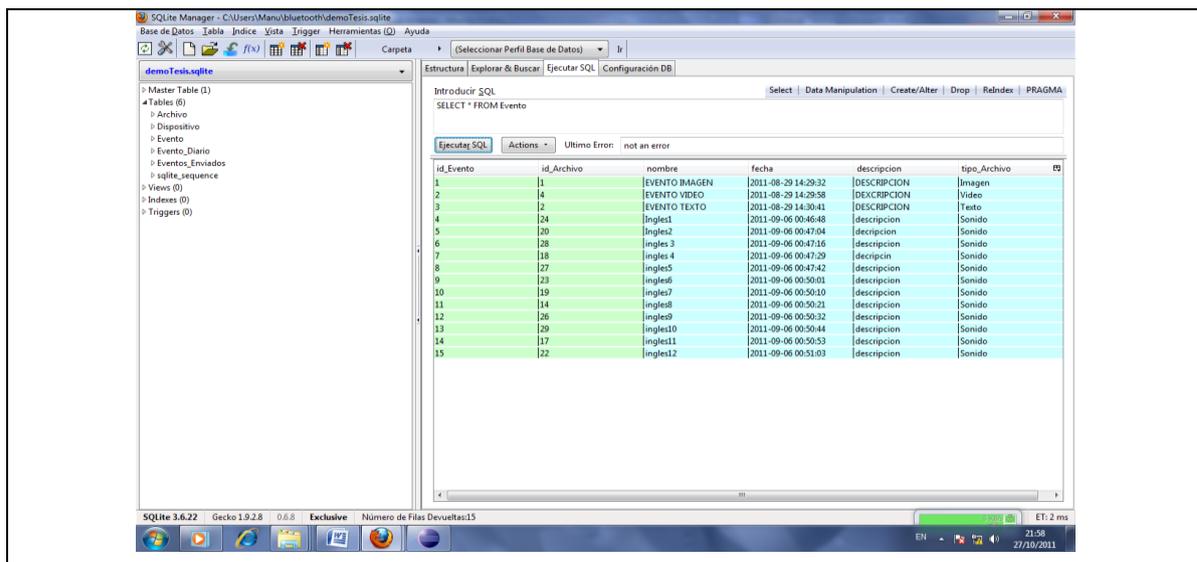
Fuente: Propia

Figura F.7. Seleccionar la perspectiva Flash.

Instalación SQLite Manager

Para un manejo de las tablas más fácil de la base de datos SQLite instalamos SQLite Manager.

- Abrimos el navegador Mozilla Firefox, damos clic en Herramientas / Complementos / buscamos el explorador & complemento SQLite Manager e instalamos.
- Una vez instalado el complemento SQLite Manager, damos clic en Herramientas / SQLite Manager. Abre una nueva ventana en la que se administra las tablas de una base de datos SQLite, como se muestra en la siguiente imagen.



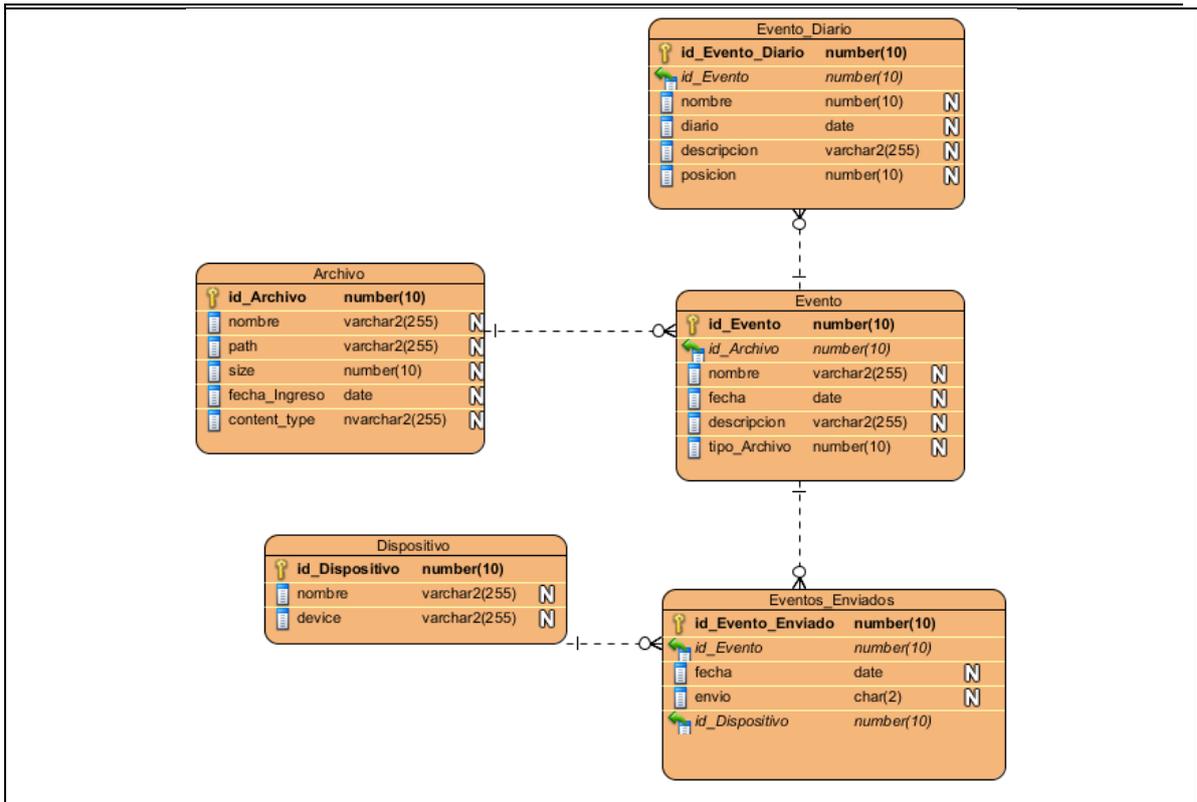
Fuente: Propia

Figura F.8. IDE para el manejo de la base de datos SQLite.

Diagrama Entidad Relación

La base de datos es el lugar donde la información quedara almacenada tras la interacción entre el usuario y el sistema. Es sin duda una pieza clave de la aplicación puesto que sin un gestor de base de datos, no se podría dejar constancia de los posibles cambios producidos.

En el diagrama entidad-relación con sus tablas y sus relaciones a implementar en sistema de Marketing por Bluetooth es el siguiente.



Fuente: Propia

Figura F.9. Diagrama Entidad Relación del sistema BlueMarking.

DESCRIPCION DEL SISTEMA BLUEMARKING

BlueMarking es un sistema que propone una solución para enviar contenido utilizando campañas de Marketing de proximidad Bluetooth a un bajo costo.

Nuestra Aplicación está formada por dos módulos:

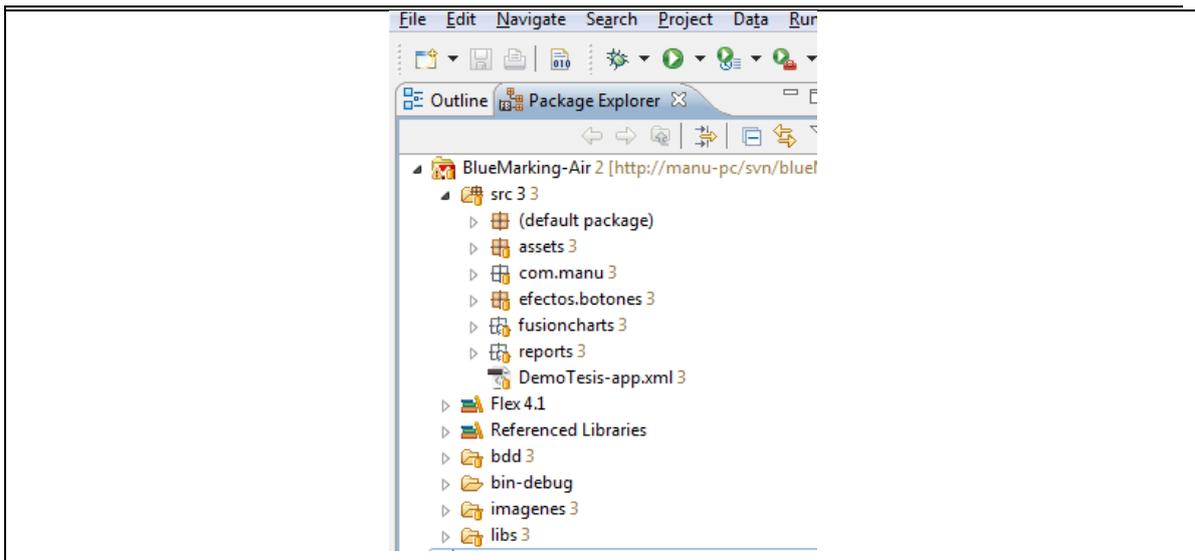
1. BlueMarking-Air

Este módulo contiene toda la interfaz de usuario, con un diseño muy amigable, está desarrollado con Adobe AIR y está compuesto por módulos .xml, componentes .xml, clases .as y los archivos .swf que utilizamos en la generación de los gráficos estadísticos.

El módulo BlueMarking-Air está construido por las siguientes carpetas:

- **src.-** En esta carpeta se encuentra todos archivos .xml y las clases .as con todo el código necesario para el presente sistema. En esta carpeta agrupamos los archivos en los siguientes paquetes.
 - **Default package.-** Aquí la página principal del sistema, esta clase .xml es la que ejecutamos y permite visualizar el sistema.
 - **Assets.-** Aquí se encuentran todos los iconos e imágenes que se utilizan en el sistema.
 - **com.manu.-** Aquí se encuentran todas las clases .as, componentes reutilizables y módulos que forman parte del sistema
 - **efectos.botones.-** en este paquete se encuentra una clase EfectosBotones con las funciones para cambiar los estilos de todos los botones que se encuentran en el sistema.
 - **Fusioncharts.-** Aquí están todas las librerías .swf que necesitamos para la generación de las gráficas fusioncharts en la parte de los reportes.
 - **Reports.-** Aquí están todos los archivos .xml que utilizamos para la generación de los EXCEL's.
- **bdd.-** En esta carpeta se encuentra los script, consultas y diseño de la base de datos.
- **bin-debug.-** En esta carpeta se encuentran todos los archivos compilados o binarios
- **imágenes.-** En esta carpeta se encuentra las imágenes que utilizamos en el sistema.
- **libs.-** En esta carpeta se encuentra todas las librerías .swf y .jar auxiliares para que sistema pueda realizar todas las funciones de conectividad.

En la siguiente gráfica podemos ver la estructura del módulo BlueMarking-Air.



Fuente: Propia

Figura F.10. Estructura del módulo BlueMarking-AIR.

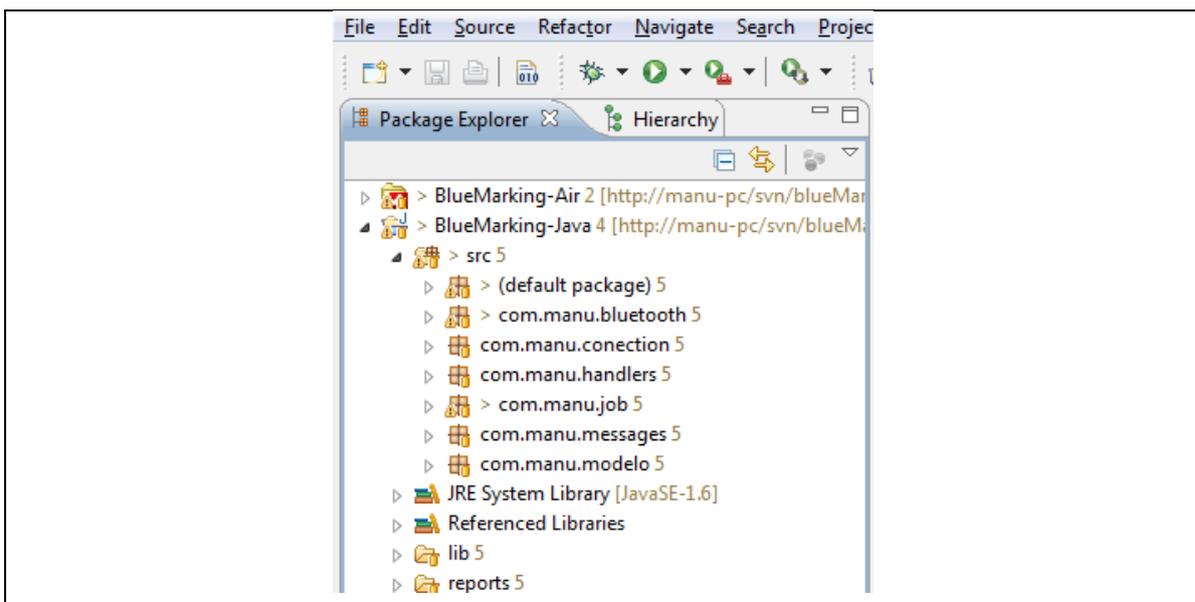
2. BlueMarking-java

Este módulo contiene todo el manejo del dispositivo Bluetooth y la transferencia de Eventos con publicidad y propaganda a los teléfonos celulares, este módulo está desarrollado en Java como una aplicación de escritorio.

El módulo BlueMarking-Java está construido por las siguientes carpetas:

- **src.-** En esta carpeta se encuentra todas las clases que utilizamos en el módulo BlueMarking-Java. . En esta carpeta agrupamos los archivos en los siguientes paquetes.
 - **Default package.-** Aquí encontramos la clase principal de nuestra aplicación Java, esta clase contiene el método Main, que el que ejecuta la aplicación de escritorio.
 - **com.manu.Bluetooth.-** Aquí se encuentran todas las clases para el manejo y conexión de Bluetooth entre el sistema y un dispositivo móvil.
 - **com.manu.conection.-** Aquí se encuentran la clase Coneccion.java que tiene todos los métodos de conexión a la base de datos y los métodos de consultas y de inserción a la base SQLite
 - **com.manu.handlers.-** Aquí se encuentran la clase QuestionHandler.java que es un controlador que debe tener una función handleMessage, encargada de la conexión entre AIR y Java.

- **com.manu.job.-**
 - **com.manu.messages.-** Aquí se encuentran la clase MessageEvento.java, esta tiene las propiedades que necesitamos en el puente y una variable para diferenciar de los distintos tipos de mensajes que están en el puente.
 - **com.manu.modelo.-** Aquí se encuentran todas las clases mapeadas de la base de datos.
- **libs.-** En esta carpeta se encuentra todas las librerías .jar necesarios para poder generar los reportes, conectarse al módulo BlueMarking-Air y para poder acceder a la base de datos SQLite.
- **Reports.-** Aquí están todos los archivos .jrxml que utilizamos para la generación de los EXCEL's.



Fuente: Propia

Figura F.11. Estructura del módulo BlueMarking-Java.

Comunicación entre el módulo BlueMarking-Air y BlueMarking-Java

Para poder obtener una comunicación entre una aplicación AIR y una aplicación Java utilizamos Merapi, que es un puente que permite la comunicación a través del uso de los mensajes. Los mensajes pueden contener datos complejos y los protocolos de AMF de serialización de datos se utilizan para la transferencia de datos.