

Mercurio Development Kit (MDK) Versión 0.6



Contenido

1	INTRODUCCIÓN.....	4
2	MARCO TEÓRICO.....	5
2.1	ABONADO.....	5
2.2	APLICACIÓN INTERACTIVA.....	5
2.3	GARANTÍA REGULATORIA.....	5
2.4	MO.....	5
2.5	MT.....	5
2.6	OPERADOR.....	5
2.7	SUSCRIPTOR.....	5
3	ASPECTOS REGULATORIOS.....	6
3.1	VENEZUELA.....	6
3.1.1	<i>Mensaje no solicitado</i>	6
4	REQUERIMIENTOS TÉCNICOS.....	7
4.1	SISTEMA OPERATIVO.....	7
4.2	SOFTWARE DE APOYO.....	8
4.3	CONECTIVIDAD.....	8
5	FUNCIONALIDAD.....	10
5.1.1	<i>Envío de mensajes de texto</i>	10
5.1.1.1	enviarSMS.....	10
5.1.2	<i>Recepción de mensajes de texto</i>	11
5.1.2.1	configRecepciónSMS.....	11
5.1.2.2	recibirSMS.....	12
5.1.2.3	pararRecepcionSMS.....	12
6	CÓDIGOS DE ERROR.....	14
7	AMBIENTES DE USO.....	15
7.1	SERVICIO WEB (WEB SERVICE).....	15
7.2	JAVA.....	15
7.3	MICROSOFT .NET.....	15
7.4	MANEJADOR DE BASE DE DATOS RELACIONAL (RDBMS).....	15



Mercurio Development Kit (MDK) Manual de Referencia

Versión 0.6

1 Introducción

El MDK (Mercurio Development Kit) de Mercurio es un conjunto de herramientas que ayudan a cualquier desarrollador a habilitar sus aplicaciones para enviar y recibir mensajes de texto hacia y desde teléfonos con capacidad de mensajería de texto o multimedia.

Este documento es el manual de referencia para el uso del MDK, por lo que es un buen punto de partida para comenzar a usar Mercurio, el motor de la ubicuidad de la mensajería de texto.

El MDK se encuentra constantemente en desarrollo por lo que es importante que usted se asegure de contar con la última versión. El MDK es distribuido gratuitamente por Tedexis y toda la información puede ser obtenida en <http://www.tedexis.com/mercurio/mdk/>. En este momento el MDK es únicamente este documento.

2 Marco teórico

Esta sección tiene como objetivo definir claramente algunos conceptos utilizados a lo largo de este documento de manera de no dejar lugar a dudas o mal interpretaciones. A continuación se especifica cada uno de estos conceptos.

2.1 Abonado

Usuario de comunicación con capacidad de mensajería de texto SMS o MMS. Usualmente es un usuario de telefonía celular, pero podría ser de otro tipo de equipo.

2.2 Aplicación Interactiva

Aplicación que ante un mensaje que un abonado le envía, reacciona respondiendo con otro mensaje. Cada par de mensajes

Una aplicación interactiva puede desencadenar varios mensajes de envíos y respuesta.

2.3 Garantía regulatoria

La Garantía Regulatoria es un depósito que debe mantener el usuario para cubrir sanciones regulatorias en caso de un reclamo que implique el pago de una sanción. En el pasaporte se define la cantidad de mensajes que se quiere garantizar. Al hacer uso de Mercurio-X no se requiere de dicha garantía ya que Mercurio-X provee la garantía.

2.4 MO

Del inglés "Message Origination", hace referencia a la Emisión o Iniciación de mensajes.

2.5 MT

Del inglés "Message Termination", hace referencia a la Terminación o entrega final de un mensaje al abonado al que ha sido destinado el mensaje.

2.6 Operador

Empresa de telecomunicaciones dueña de una red de telecomunicaciones la cual, a través de tecnologías específicas, provee servicio de comunicación a sus abonados.

2.7 Suscriptor

Abonado suscrito a un canal de contenido manejado por la plataforma Mercurio.

Las operadoras llaman de esta manera a sus abonados, por lo que es importante tener claro el contexto en el cual la palabra "suscriptor" es utilizada. Por ejemplo, un "suscriptor celular" hace referencia a un abonado. Cuando un operador hace referencia a suscriptores, hace referencia a sus abonados.

3 Aspectos regulatorios

Es importante tener claras las restricciones regulatorias existentes en cada uno de los países en dónde se desea terminar mensajes (MT). Por restricción regulatoria hacemos referencia tanto a las regulaciones propias de las operadoras (regulaciones privadas) como aquellas provistas por entes regulatorios estatales.

A continuación se presentan las restricciones regulatorias por país. Cada restricción está claramente identificada como "Privada" o "Legal", lo cual explicita si la regulación es provista por las operadoras (privada) o por un ente público (Legal).

3.1 Venezuela

3.1.1 Mensaje no solicitado

Para poder terminar un mensaje en un abonado se requiere que el emisor tenga una prueba de la solicitud del mensaje por parte del abonado.

Esta regulación es de índole privada y cada operador tiene su propia política de verificación como de sanción contra quien haga caso omiso a la regulación.

Política de Verificación: Excepcional: verifica la prueba de solicitud del mensaje sólo en caso de una queja o reclamo del abonado.

Política de Sanción: Cobra al emisor una penalidad.

4 Requerimientos Técnicos

Para poder utilizar el MDK de Mercurio, éste debe ser instalado en un servidor que cumpla con las siguientes características:

4.1 Sistema Operativo

A continuación se enumeran los sistemas operativos certificados sobre los cuales corre el MDK:

- Windows - Windows 95, Windows 98, Windows NT, Windows 2000, Windows ME, Windows CE
- Linux (cualquier sabor como: Red Hat, Mandrake, Ark Linux, SuSE, Conectiva, PLD, Gentoo, Debian, etc.)
- Novell
- Unix
 - NeXTStep
 - Solaris Operating Environment - Solaris-SPARC, Solaris-Intel
 - SunOS
 - BSDi
 - Tenon Mach Ten BSD (for Macintosh)
 - IRIX
 - HP/UX
 - AIX
 - Univel SVR4
 - OSF
 - Fujitsu UXP/DS
 - DGUX
 - MP/RAS
- Variantes de BSD
 - FreeBSD
 - NetBSD
 - OpenBSD
- OS/400
- OS/390,
- z/OS
- Sistemas de Operación embebidos o de tiempo real
 - eCOS
 - VxWorks
 - pSOS
 - ThreadX
 - SMX
 - Nucleus
 - RTEMS
- AtheOS
- BeOS

- AmigaOS
- MiNT
- FreeMiNT
- Plan9

Los procesadores soportados bajo los sistemas operativos indicados son:

- x86 / IA32
- ARM
- MIPS
- PowerPC
- Sparc
- SuperH
- m68k
- CRIS
- S390
- IA64
- Alpha
- HP PA RISC
- Procesador zSeries

Es importante resaltar que en la medida que un cliente desee correr el MDK en una plataforma no certificada, Tedexis puede trabajar con el cliente para portar el MDK a la plataforma requerida.

4.2 Software de apoyo

Para la ejecución de las funcionalidades del MDK es necesario tener los siguientes softwares en el equipo de producción

- 1.1. Ambiente de ejecución java jre 1.4 o superior
- 1.2. Servidor web con soporte a servlets y jsp (Sólo para el uso de Oz¹)

4.3 Hardware recomendado

El equipo mínimo para instalar el mdk debe tener las siguientes características:

- Pentium 3 o superior,
- 256 MB de memoria
- 100MB espacio libre en disco.

Estos requerimientos pueden ser modificados según la evaluación que haga el personal de consultoría en función al número de mensajes que se requieran transmitir.

¹ Oz es una aplicación de la suite de Tedexis que automatiza el proceso del MDK con interfaces ya prehechas

4.4 Conectividad

Si usted desea terminar mensajes en teléfonos celulares, basta con que el MDK tenga acceso a Internet, en particular Acceso al servidor de twindivision (64.76.64.187) a través de los puertos 8080, 8081 y 8100. La comunicación es bidireccional.

En el caso de que desee recibir mensajes interactivos, el MDK debe tener un IP válido en Internet ya sea porque la máquina en donde corre lo tiene, o existe una configuración de enrutamiento IP en los enrutadores y firewalls que permitan el acceso al MDK a través de una IP válida desde Internet, de igual forma se requiere la apertura de los puertos mencionados anteriormente.

Dependiendo del caso, Tedexis puede habilitar una conexión vía VPN o red privada entre el MDK y Mercurio, de manera de garantizar seguridad y/o calidad en la transmisión de los mensajes.

5 Funcionalidad

A continuación se describen tanto las funcionalidades de Mercurio como las diferentes alternativas que existen actualmente para uso; escoja el ambiente de desarrollo de su preferencia y vea como es fácil utilizar Mercurio.

Independientemente de la forma como usted use Mercurio, siempre encontrará la misma funcionalidad. Esta sección describe la funcionalidad de manera de poder lograr la interacción vía SMS/MMS con dispositivos telefónicos.

A continuación se muestran cada una de las funcionalidades disponibles, indicando la función o método que ha de ser llamado en el ambiente que usted use. Se indican los códigos de error básicos. Más adelante se encuentra un capítulo con la lista completa de posibles códigos de error y su significado.

Por "firma" se hace referencia a la descripción de los argumentos y su respectivo orden como han de ser llamados para lograr la función.

5.1.1 Envío de mensajes de texto

5.1.1.1 enviarSMS

`enviarSMS (pasaporte, clave, codigoPais, codigoArea, telefono, mensaje)`

Parámetro	Tipo	Descripción	Ejemplo ²
Pasaporte	Entero	ID del pasaporte obtenido de Tedexis	10030001
Clave	Texto	Clave secreta alfanumérica. Provista inicialmente por Tedexis al otorgar el pasaporte.	55365
CodigoPais	Entero	Código del país del número al cual va a ser enviado el mensaje.	58
CodigoArea	Entero	Código del área del número al cual va a ser enviado el mensaje.	416
Telefono	Entero	Número dentro de la red del operador telefónico.	6229864
Mensaje	Texto	Mensaje de texto	Hola!, mundo inalámbrico.

Retorno:

Tipo	Entero
Significado	Estado de error (0 sin error)

Comportamiento:

1. Sólo intenta enviar el mensaje si y sólo si se cumple con:
 - a. ID de pasaporte válido
 - b. Clave correcta del pasaporte

² El ejemplo refleja el envío de un mensaje al 0416-622.9864 en Venezuela.

- c. Saldo (o crédito) en el pasaporte para enviar el mensaje
- d. La red Mercurio tiene una ruta de despacho para el Código de País y de Área utilizados.
2. Por defecto, el mensaje será enviado truncando el mensaje si éste excede la longitud máxima de 129 caracteres (siendo esto configurable).
3. El mensaje será despachado inmediatamente. Mercurio es un carrier Tier 1 con muchas operadoras, por lo que puede garantizar la entrega directamente hasta el operador de telefonía. Si el operador o red de despacho internacional no pueden recibir el mensaje, Mercurio lo tendrá en tránsito hasta 24 horas. Después de ese tiempo será eliminado de la plataforma siendo reportado en una lista de mensajes eliminados.³
4. Si el mensaje no puede ser despachado por algún motivo, el código de error retornado indica la causa.⁴

5.1.2 Recepción de mensajes de texto

Para recibir mensajes de texto, se deben utilizar dos funciones:

1. Configuración del mecanismo de recepción de SMS (`configRecepcionSMS()`)
2. Inicio de recepción (`recibirSMS()`)
3. Parada del proceso de recepción de SMS (`pararRecepcionSMS()`)

Para poder recibir mensajes de texto el Pasaporte debe estar configurado para ello. Su configuración puede hacerse al momento de adquirirlo o en cualquier momento posterior. Dicha configuración incluye la información del número al cual el usuario de telefonía ha de enviar los mensajes de texto para que lleguen al pasaporte.

5.1.2.1 configRecepciónSMS

`configRecepcionSMS (pasaporte, clave, RECEPTOR)`

Parámetro	Tipo	Descripción	Ejemplo ⁵
Pasaporte	Entero	ID del pasaporte obtenido de Tedexis	10030001
Clave	Texto	Clave secreta alfanumérica. Provista inicialmente por Tedexis al otorgar el pasaporte.	55365
RECEPTOR	OBJETO	Objeto cuya función será procesar el mensaje que llegue.	Objeto Java o Servicio Web.

Retorno:

Tipo	Entero
Significado	Estado de error (0 sin error)

Comportamiento:

³ Si se desea un servicio con mayor latencia de mensajes en tránsito en la plataforma, o entrega garantizada por contrato, puede ser negociado con Tedexis a un costo particular, siendo configuradas estas características en el pasaporte.

⁴ No aplica cuando el servicio es garantizado.

⁵ El ejemplo refleja el envío de un mensaje al 0416-622.9864 en Venezuela.

1. Sólo configura la recepción de mensajes SMS, no inicia el proceso de recepción.

5.1.2.2 recibirSMS

`recibirSMS (pasaporte, clave)`

Parámetro	Tipo	Descripción	Ejemplo ⁶
Pasaporte	Entero	ID del pasaporte obtenido de Tedexis	10030001
Clave	Texto	Clave secreta alfanumérica. Provista inicialmente por Tedexis al otorgar el pasaporte.	55365

Retorno:

Tipo	Entero
Significado	Estado de error (0 sin error, -1 si el proceso se encontraba activo)

Comportamiento:

1. Inicia el proceso de recepción de mensajes de texto si y sólo si ha sido previamente configurado.
2. Los mensajes que se encontraran pendientes por ser procesados aún en la plataforma serán empujados por Mercurio al Objeto receptor configurado previamente.
3. Si en algún momento la plataforma no recibe respuesta de la aplicación, además de enviar una notificación vía email a la dirección configurada en el pasaporte, mantendrá en tránsito en la plataforma los mensajes no procesados durante la latencia indicada en el pasaporte.

5.1.2.3 pararRecepcionSMS

`pararRecepcionSMS (pasaporte, clave)`

Parámetro	Tipo	Descripción	Ejemplo ⁷
Pasaporte	Entero	ID del pasaporte obtenido de Tedexis	10030001
Clave	Texto	Clave secreta alfanumérica. Provista inicialmente por Tedexis al otorgar el pasaporte.	55365

Retorno:

Tipo	Entero
Significado	Estado de error (0 sin error, -1 si el proceso se encontraba detenido)

Comportamiento:

⁶ El ejemplo refleja el envío de un mensaje al 0416-622.9864 en Venezuela.

⁷ El ejemplo refleja el envío de un mensaje al 0416-622.9864 en Venezuela.

1. Detiene el proceso de recepción de mensajes de texto.

6 Códigos de error

A continuación se presenta la tabla de posibles códigos de error del sistema Mercurio.

EXITO=0;

ERROR_INFRAESTRUCTURA=-100;

ERROR_INTERNO=-101;

OPERADOR_FUERA_DE_LINEA=-102;

RECEPTOR_NO_DISPONIBLE=-103;

DESTINO_NO_ACEPTADO_POR_OPERADOR=-104;

NO_AUTENTICADO=-105;

SERVICIO_NO_DISPONIBLE=-106;

RECEPCION_ESTABA_EN_PROCESO=-201;

RECEPCION_NO_ESTABA_EN_PROCESO=-202;

NO_CONFIGURADO=-203;

COD_PAIS_NO_SOPORTADO=-204;

COD_AREA_NO_SOPORTADO=-205;

NUMERO_NO_SOPORTADO=-206;

MENSAJE_RECHAZADO=-207;

ERROR_NUMERO_SUMINISTRADO=-208;

PASAPORTE_EXISTENTE=-209;

7 Ambientes de uso

A continuación se describen las idiosincrasias del uso de Mercurio a través de varias formas. En siguientes versiones de este manual incluirán ejemplos, los cuales se encontrarán disponibles en la distribución del MDK.

7.1 Servicio web (web service)

Mercurio está disponible en Internet como un servicio web cuya descripción está disponible en <http://mercurio.tedexis.com:8081/merlin/mercurio.wsdl>

7.2 Java

El MDK provee un paquete Java el cual provee los objetos que implementan las funcionalidades antes indicadas. Basta con incluir el paquete y hacer uso de las clases.

7.3 Microsoft .NET

En el ambiente de desarrollo del Visual Studio .NET de Microsoft existe una opción para usar servicios web. Al activar esta opción con la dirección del WSDL de Mercurio (<http://mercurio.tedexis.com:8081/merlin/mercurio.wsdl>), aparecerá en el ambiente de desarrollo un objeto nuevo llamado "Mercurio".

Una vez teniendo disponible el objeto "Mercurio" en el ambiente de desarrollo solo hace falta hacer uso de sus funcionalidades descritas en el capítulo "Funcionalidad".

7.4 Manejador de Base de Datos Relacional (RDBMS)

Cuando se desea instalar el MDK para ser utilizado a través de un manejador de base de datos relacional (RDBMS), los siguientes manejadores están soportados por el MDK:

- Oracle 8, 9i, 10g
- Sybase
- Informix
- DB2
- Microsoft SQL Server
- CACHE
- MySQL 3.23, 4.0
- PostgreSQL 7.1.2, 7.2, 7.3, 7.4
- Daffodil DBv3.1
- Interbase 6.0.1 (Open Source) with Firebird InterClient 2.01
- HypersonicSQL 1.61, 1.7.0
- Mckoi SQL 0.93
- Data Replication Tool
- Progress 9
- Pointbase Embedded 4.3
- SAP DB 7.3
- Ingres
- FrontBase

Es importante resaltar que en la medida que un cliente desee correr el MDK en un manejador de base de datos no certificada, Tedexis puede trabajar con el cliente para portar el MDK a la plataforma requerida.