



**UNIVERSIDAD  
DE ANTIOQUIA**

**Actualización del aplicativo SADES a estándares de  
desarrollo modernos**

Autor(es)

Juan Carlos Londoño Gómez

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Sistemas

Medellín, Colombia

2021



**Actualización del aplicativo SADES a estándares de desarrollo modernos**

**Juan Carlos Londoño Gómez**

Informe presentado como requisito para optar al título de:

**Ingeniero de Sistemas**

Asesores (a):

José Ignacio López Vélez - Interno

John Sergio Ramírez Castañeda - Externo

Modalidad:

Práctica empresarial

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Sistemas

Medellín, Colombia

2021.

## Resumen

SADES<sup>1</sup> es una herramienta que nace como una alternativa a SMNET<sup>2</sup>, la cuál es la herramienta principal de gestión de servicios de cara a prestar una atención técnica a los servicios fijos de la compañía **TIGO**. SADES fue creada debido a las constantes fallas que presenta SMNET, lo cual los dejaba a la compañía sin ninguna opción de cara a la atención al cliente, y aunque el proyecto arrancó, no se hizo de la mejor manera debido a que no se siguió ningún patrón ni principio de buenas prácticas, lo que llevó a que en poco tiempo de iniciar la implementación se hiciera mucho más complicado incorporar nuevas funcionalidades y darle mantenimiento a las ya existentes, por lo que era necesario realizar una migración desde este aplicativo a uno nuevo que incluyera patrones y principios de buenas prácticas que permitieran mejorar los aspectos de mantenibilidad, escalabilidad, usabilidad, flexibilidad, seguridad y rendimiento, incorporando además herramientas de versionamiento de software como lo es GIT para aprovechar al máximo el talento humano que fue designado para llevar a cabo el proyecto, obteniendo como resultado un aplicativo mucho más estructurado que genera una reducción de costos y tiempo a la hora de llevar a cabo nuevas implementaciones y/o mantenimiento de las ya existentes.

**Palabras claves:** SADES, SMNET, Migración, Git, SCRUM, Buenas prácticas, Herramienta de soporte técnico, Versionamiento.

---

<sup>1</sup> SADES: Sistema Alterno de SMNET

<sup>2</sup> SMNET: Siplax Management NET

## **CONTENIDO**

INTRODUCCIÓN	<b>6</b>
SITUACIÓN PROBLEMÁTICA	6
OBJETIVOS	7
Objetivo General	7
Objetivos Específicos	7
METODOLOGÍA	7
ALCANCE	8
MARCO TEÓRICO	<b>9</b>
RESULTADOS Y ANÁLISIS	<b>10</b>
CONCLUSIONES Y TRABAJO FUTURO	<b>16</b>
CONCLUSIONES	16
TRABAJO FUTURO	17
REFERENCIAS	<b>18</b>
ANEXOS	<b>19</b>

## LISTA DE TABLAS

- **Tabla 1.** *funcionalidades SADES, antes de la migración.* [11](#)
- **Tabla 2.** *Roadmap de desarrollo SADES.* [13](#)

## LISTA DE FIGURAS

- **Figura 1.** *Indisponibilidad mensual de SMNET* [10](#)
- **Figura 2.** *Diagrama de flujo MVC* [12](#)
- **Figura 3.** *AHT asesores* [15](#)

# 1. INTRODUCCIÓN

SADES es una herramienta alterna a SMNET, herramienta que es utilizada para gestionar el servicio al cliente prestado por la compañía **TIGO**, dicha herramienta generó mucha expectativa a nivel administrativo, ya que prometía simplificar muchos de los procesos que se venían realizando, además de suplir las necesidades generadas durante las fallas de la herramienta principal SMNET. El problema surge cuando se inicia el desarrollo de SADES debido a su poca planeación y visualización a futuro, por lo que no se definió una arquitectura del proyecto, no se consideraron patrones de diseños, ni principios de buenas prácticas, al inicio no era tanto un problema, pero a medida que iba siendo más robusto el proyecto, se hacía más difícil de mantener e implementar nuevas funcionalidades. La solución planteada a este problema fue la de migrar las funcionalidades existentes a una nueva versión la cual incorpora el patrón de diseño de software conocido como **MVC**, ya que permitía separar los componentes principales y así mismo tener más control sobre el flujo de información en el aplicativo. Como resultado de aplicar este patrón también trajo consigo el uso de herramientas de versionamiento de software como GIT, la cual fue fundamental para lograr gestionar de manera más sencilla el trabajo en equipo y paralelo, logrando así grandes avances en muy poco tiempo durante la implementación de las funcionalidades. Durante este proceso fue necesario llevar a cabo una metodología tipo SCRUM que nos permitiera estar en constante comunicación, identificando fácilmente las dificultades y obstáculos que nos permitieran avanzar y generar ideas y aportes para sobrellevarlas de la mejor manera. Finalmente no sólo se logró migrar la herramienta si no se que se llevó hasta un punto en el que se encuentra actualmente en producción siendo utilizada por más de mil asistentes técnicos simultáneamente.

## 1.1. SITUACIÓN PROBLEMÁTICA

SADES como herramienta alterna, buscaba ofrecer una solución a las constantes fallas que se presentan en SMNET, no obstante, durante el inicio de su desarrollo se identificaron varios problemas, entre ellos, la falta de una herramienta de versionamiento que facilitara el trabajo en equipo y la ausencia de una arquitectura, patrón y/o principio de buenas prácticas que le permitiera al proyecto mantenerse y extenderse fácilmente a futuro.

## OBJETIVOS

### 1.1.1. Objetivo General

Adaptar el aplicativo S.A.D.E.S a los principios de buenas prácticas, aplicando patrones arquitectónicos y herramientas de versionamiento, que permitan disminuir los tiempos y costos de mantenimiento e implementación, incorporando además el desarrollo en paralelo por parte de los integrantes del equipo.

### 1.1.2. Objetivos Específicos

- Diseñar e implementar una nueva arquitectura general del proyecto.
- Versionar el proyecto y capacitar al equipo en el manejo de git.
- Migrar funcionalidades actuales a la nueva arquitectura.
- Evaluar el correcto funcionamiento de la nueva versión.
- Documentar el nuevo aplicativo.
- Contrastar el aplicativo anterior con el nuevo, indicando principales diferencias.
- Divulgar los resultados obtenidos.

## 1.2. METODOLOGÍA

Durante el desarrollo de este proyecto se utilizó la metodología de desarrollo ágil SCRUM [5]:

- Inicialmente se realizó un recorrido por el aplicativo para contextualizar y documentar el estado actual en que se encontraba, este recorrido se llevó a cabo con los integrantes del proyecto para conocer más a fondo acerca de la implementación, el stack de tecnologías, la distribución de los componentes, etc. Que se estaban utilizando.
- Se realizó una ceremonia inicial en la cual nos reunimos cada uno de los integrantes del proyecto incluido el *Product Owner*.

- Se realizaron reuniones semanales con el *Scrum Master* para revisar los avances de la semana anterior y definir las nuevas actividades que se iban a llevar a cabo durante la siguiente semana.
- Se llevaron a cabo reuniones diarias donde los integrantes del equipo de desarrollo presentaban lo que se había hecho, lo que faltaba y las dificultades que se habían presentado.
- Se elaboraron reuniones quincenales para capacitar a los desarrolladores en el uso de herramientas y estándares de desarrollo modernos.
- Al finalizar cada uno de los sprints se realizaron pruebas unitarias y luego de integración para validar el comportamiento esperado del proyecto.
- Finalmente se entregó una versión del aplicativo funcional junto a su respectiva documentación que permitiera evidenciar las principales diferencias del nuevo desarrollo con el antiguo.
- Se presentaron los resultados obtenidos durante todo el proceso de desarrollo.

### **1.3. ALCANCE**

Se cumplió con el alcance esperado, el cual estimaba migrar el aplicativo S.A.D.E.S a una nueva versión, incorporando las nuevas tecnologías y los principios básicos de buenas prácticas propuestos en cada uno de los objetivos, además, se lograron incorporar nuevas funcionalidades para el módulo de EDATEL, permitiendo así expandir el soporte brindado por el aplicativo a la tecnología de red cobre.

## 2. MARCO TEÓRICO

A lo largo del tiempo, los desarrollos de *software* se han convertido en herramientas fundamentales tanto para grandes, medianas y pequeñas empresas, ya que permiten agilizar procesos, disminuir esfuerzos y costos en los distintos temas en los que se vea involucrada la empresa, es por esto que no sólo basta con desarrollar *software* si no que además, hay que hacerlo bien, siguiendo estándares y principios de buenas prácticas de desarrollo, ya que si no lo hacemos, en un futuro no muy lejano en vez de ofrecernos beneficios, nos brinda problemas, los cuales pueden involucrar costos de mantenimiento, tiempo de implementación y/o modificación, etc.

Teniendo en cuenta lo anterior, es importante recalcar los beneficios que obtenemos al utilizar estándares y principios de buenas prácticas, tanto para diseñar como para desarrollar la arquitectura general de un proyecto de *software*. Y aunque inicialmente se nos pueda complicar la implementación de los mismos, las ventajas que nos ofrece a futuro valen realmente la pena, algunos de estos patrones y diseños son los siguientes:

- **Patrón de diseño MVC** [1]: Este patrón de diseño nos permite separar los componentes del Modelo, la Vista y el Controlador permitiendo trabajar en cada uno de estos al mismo tiempo sin que los demás componentes se vean realmente afectados, logrando así aprovechar al máximo el talento humano.
- **Versionamiento con GIT** [2]: Es una herramienta de versionamiento la cual tiene como propósito llevar un registro de los cambios realizados y coordinar el trabajo que realizan varias personas sobre archivos compartidos en un repositorio. Esta herramienta junto con el patrón MVC, nos ayuda a potenciar el trabajo en equipo.
- **Código Limpio** [3]: Es un principio de buenas prácticas que nos ayuda a crear código intuitivo, y fácil de entender y modificar, y es muy importante ya que un sistema nunca está totalmente finalizado, debido a que existe la necesidad de realizar actualizaciones y nuevas funcionalidades, lo que se logra mucho más fácil si tenemos un código limpio.

- **Principios SOLID** [4]: Son un grupo de 5 principios (**S**ingle responsibility, **O**pen/closed, **L**iskov substitution, **I**nterface segregation, **D**ependency inversion), que nos permite desarrollar sistemas que son fáciles de mantener y ampliar funcionalidades fácilmente sin tener que realizar grandes modificaciones en el código existente.
- **SCRUM** [5]: Es una metodología de desarrollo ágil que consiste en dividir las diferentes etapas de un proyecto en pequeños bloques o *sprints*, durante estos *sprints*, suelen hacerse reuniones diarias donde se comparte lo que se ha hecho, lo que falta por hacer y qué dificultades se presentaron, con el objetivo de ir revisando y mejorando las fases anteriores del proyecto.

En consecuencia, la no implementación de los mismos pueden traernos grandes riesgos dentro de la compañía, ya que el desarrollo puede presentar diversas fallas o necesite mejoras y/o nuevas implementaciones, lo cual va a generar un costo desmedido no sólo en términos económicos sino del tiempo que pueda tomar dicha diligencia [6].

### 3. RESULTADOS Y ANÁLISIS

SADES, nace como una idea de contingencia a las constantes fallas que se presentan en la plataforma principal de atención al cliente (SMNET) dentro de la compañía **TIGO**, tal y como se puede evidenciar en la siguiente gráfica:



Figura 1. Indisponibilidad mensual de SMNET

Pero no es aquí realmente donde se evidencia el problema, por el contrario, SADES nos está ofreciendo una solución a una problemática ya identificada, sin embargo, el problema se presentó durante la etapa de implementación, ya que se inició su desarrollo sin antes haber definido la arquitectura, patrones, principios de buenas prácticas y herramientas que respaldaran el proyecto. Por lo tanto, no se tenía del todo claro cuál era el flujo de la información ni la ubicación de los componentes, entre otros “anti-patrones” que destinaban el proyecto a convertirse en algo difícil de mantener y extender a medida que fuera creciendo en funcionalidades.

Afortunadamente, cuando se evidenció el problema, el proyecto se encontraba en una etapa temprana en la cual aún contaba con muy pocas funcionalidades, tal y como se puede observar en la *tabla 1*. Pudiendo así evaluar el impacto que tendría continuar desarrollando bajo esas condiciones o migrar la plataforma a una nueva versión que cubriera todas las falencias mencionadas anteriormente, tomando así la decisión final de migrar la plataforma.

<b>Tecnología HFC<sup>3</sup></b>
Consultar Portafolio
Estado CMTS
Estado Cable Modem
Trafico Downstream
Perfil LDAP
Interfaz del Nodo

*Tabla 1. funcionalidades SADES, antes de la migración.*

---

<sup>3</sup> HFC: Híbrido de Fibra Coaxial.

La migración del proyecto, consistió en adaptar las funcionalidades ya existentes a una nueva versión del proyecto que incluyera el patrón de diseño MVC en el IDE de desarrollo Microsoft Visual Studio, el cual ya cuenta con un Framework para esto (.NET Framework).

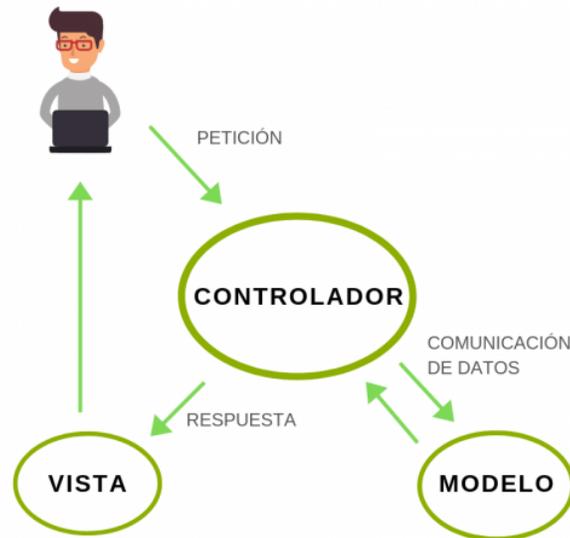


Figura 2. Diagrama de flujo MVC

Dicha tarea no fue nada sencilla, debido a que inicialmente la curva de aprendizaje fue un poco empinada y era necesario interiorizar muy bien cada uno de los conceptos definidos por MVC y la aplicación de los mismos (*figura 2*), pero una vez superada esta etapa, el proceso de implementación fue mucho más sencillo gracias a las facilidades que nos ofrece el framework.

Todo este proceso de migración se realizó bajo una metodología tipo SCRUM, para la cual se planearon reuniones diarias en horas de la mañana, lo que nos permitió estar en constante comunicación con todo el equipo, informando sobre avances, dificultades y/o oportunidades que se presentaban. Además, se contó con un excelente líder (SCRUM Master), el cual aparte de encaminar la implementación del proyecto, estuvo siempre al

tanto de realizar las gestiones necesarias con las demás áreas administrativas para poder así tener acceso a la información que el aplicativo requería.

Los *sprints* que se llevaron a cabo durante este periodo fueron los siguientes:

# Sprint	Duración	Actividad
<b>Tecnología HFC TIGO</b>		
<b>1</b>	6 Semanas	Adaptar funcionalidades existentes a la nueva versión
<b>ADSL EDATEL</b>		
<b>2</b>	2 Semanas	Consultar cliente en portafolio.
<b>3</b>	1 Semana	Consultar el estado y perfil del DSLAM del cliente.
<b>4</b>	1 Semana	Consultar la configuración LDAP
<b>5</b>	1 Semana	Consultar información sobre la configuración de la VLANs.
<b>6</b>	1 Semana	Incorporar botones de acción refrescar, reiniciar y configurar puerto DSLAM
<b>7</b>	1 Semana	Incorporar botones de acción de configurar/eliminar VLANs.
<b>8</b>	1 Semana	Consultar información RADIUS
<b>Compartido entre tecnologías</b>		
<b>9</b>	1 Semana	Incorporar autenticación y autorización.
<b>10</b>	1 Semana	Consultar 4C
<b>11</b>	2 Semanas	Consultar el estado del servicio de los vecinos más cercanos
<b>12</b>	1 Semana	Consultar Pedidos e Incidentes

*Tabla 2. Roadmap de desarrollo SADES*

Durante el proceso de implementación de este *roadmap* se hizo uso de la herramienta de versionamiento **GIT** (ver anexo 1), la cual fue clave para aprovechar al máximo todo el talento humano que se encontraba detrás del proyecto trabajando paralelamente. Inicialmente el código fuente de proyecto se encontraba almacenado en Google Drive, lo que no suponía ninguna ventaja frente a las que ofrece GIT respecto a la gestión del mismo. Para este cambio fue necesario una capacitación a todo el equipo de desarrollo que le permitiera evidenciar las ventajas y oportunidades que suponen trabajar con GIT, además de un conjunto de ejercicios prácticos que le permitiera a los participantes aplicar el uso correcto del mismo.

Como resultado de este *roadmap* se obtuvo una versión completamente diferente (ver anexos 2 y 3) en donde no sólo se mejoró la estructura interna del proyecto a través de los principios de buenas prácticas, sino que también se mejoró en general la seguridad y la experiencia de usuario a través de un aplicativo mucho más intuitivo y dinámico.

La plataforma finalmente salió a producción en una versión Beta, en la que en un ambiente controlado se le dio acceso a 100 asesores técnicos de la empresa, y junto a estos se creó un canal de comunicación en el que constantemente se notificaron las fallas y posibles mejoras que pudieran incluirse dentro del aplicativo.

Posteriormente se eligieron 10 asesores para realizar pruebas sobre el rendimiento que obtienen los asesores utilizando SMNET y luego SADES, obteniendo como resultado que no sólo cumple su propósito inicial el cual era ofrecer una alternativa al sistema SMNET mientras presenta fallos, sino que también se notó una disminución en el AHT<sup>4</sup> (*Figura 3*) que en pocas palabras significa el tiempo promedio que tarda un asesor en realizar la gestión con un cliente que se encuentra en llamada, esta disminución se da gracias a que la información se encuentra más compacta, por lo que simplifica el proceso.

---

<sup>4</sup> AHT: Average Handle Time

UsuarioRed	Creador Cx	AHT	
		Del 01 al 23 de Junio	Del 24 al 30 de Junio
Hvillamo	Harold Arley Villa Monsalve	468	415
Sboliers	Steven Eduardo Bolier Suarez	370	336
Amoligal	Adriana Marcela Molina Galaraga	433	383
Dricaurt	Deiwid Fabian Ricaurte Olivos	365	334
Kmolanos	Karen Maryed Molano Sanchez	378	347
Msolarbo	Maria Del Carmen Solar Bolaños	477	462
Jarbolva	Johan Sebastian Arboleda Vasque	413	404
Ggarccar	Glaudis Janeth Garcia Caro	474	454
Jcastanb	Jesus Alberto Castano Burgos	397	369
Jloppolo	Jenny Cristina Lopez Polo	405	378

*Figura 3. AHT asesores*

Como se puede observar en la figura anterior, La medición se realizó durante el mes de junio, para el cual los valores del AHT del 1 al 23 están asociados al uso de SMNET, mientras que el AHT del 24 al 30 es el resultado de los mismos asesores utilizando SADES. Esta disminución se traduce en dinero, ya que al poder atender a los usuarios en menos tiempo, se pueden atender más usuarios por día, y en gran escala se traduce en dinero que la empresa puede ahorrarse durante esta operación.

Finalmente se realiza la entrega del aplicativo a las personas que van a quedar encargadas, pero antes fue necesario dictar un curso del uso correcto de MVC, desde la agrupación de componentes, hasta la comunicación entre ellos. A este curso asistieron 6 personas en total (ver anexo 4), de las cuales 2 de esas iban a recibir el proyecto SADES, el curso se dictó desde el 25 de mayo hasta el 30 de junio, cada martes y miércoles, con una intensidad horario de 2 horas por sesión. Posteriormente el aplicativo fue entregado a estas dos personas, que con la preparación adquirida en el curso facilitó mucho el proceso.

Como resultado de este proyecto, se obtuvo un aplicativo mejor estructurado, con una arquitectura y principios de buenas prácticas que facilitó no solo el proceso de implementación, sino también el trabajo en equipo que este permitía al integrarse con la herramienta GIT. Se realizó la migración de todas las funcionalidades de la versión antigua de SADES a la nueva versión sin ningún problema, reutilizando mucha de la información y lógica ya existente, y debido al impacto positivo que trajo esta migración se incorporaron nuevas funcionalidades (*tabla 2*) que le permitieron al aplicativo ser mucho más robusto y completo. Las pruebas se realizaron en un ambiente controlado, donde se eligió a un grupo de asesores para que utilizaran el aplicativo y de la misma manera reportaran fallas, recomendaciones y/o oportunidades que identificaran en el aplicativo, posteriormente en una etapa más estable se realizó la comparación con la versión antigua, donde no sólo fue evidente la mejora a nivel gráfico mejorando así la experiencia de usuario (*anexo 2 y 3*), sino también a nivel de código de fuente donde es evidente la arquitectura, el flujo de la información, la organización de los componentes y la documentación de los mismos, logrando así cumplir con el objetivo principal de este proyecto.

## **4. CONCLUSIONES Y TRABAJO FUTURO**

### **4.1. CONCLUSIONES**

Durante el desarrollo de proyectos siempre es importante llevar a cabo una etapa previa de análisis y diseño, que permita considerar la incorporación de una arquitectura, patrones de diseño y principios de buenas prácticas, ya que estos son una base fundamental para realizar desarrollos de calidad. Las decisiones tomadas durante estas etapas previas al desarrollo son fundamentales ya que desde allí buscamos las mejores condiciones que permitan satisfacer los requerimientos del sistema. Además, la integración de proyectos con herramientas de versionamiento como GIT, siempre serán de utilidad a lo largo del desarrollo, ya que permiten llevar un control más estricto sobre los cambios que allí se realizan y facilitan el trabajo en equipo. Es importante mencionar que aunque SADES surgió como una alternativa, poco a poco se fue convirtiendo en la herramienta preferida de los asesores gracias a las ventajas y simplicidad que ofrece durante el proceso de atención al cliente.

#### **4.2. TRABAJO FUTURO**

Como continuación de este proyecto se plantea incorporar la tecnología GPON, que hasta ahora aunque muy pocos clientes cuenten con ella, es un mercado que empieza a expandirse por todos los beneficios que ofrece la fibra óptica respecto a la calidad del servicio, además sugiere incluir el diagnóstico de otras empresas filiales de las cuales TIGO es la encargada de brindar asistencia técnica, empresas como por ejemplo Edatel, ETP, Emtelco, entre otras. Logrando así una cobertura mayor a la hora de acompañamiento técnico.

## REFERENCIAS

- [1] Vyas, A. (s. f.). *MVC Pattern - Anshul vyas*. Medium. Recuperado 1 de marzo de 2021, de <https://medium.com/@anshul.vyas380/mvc-pattern-3b5366e60ce4>
- [2] *Git - What is Git?* (s. f.). git-scm. Recuperado 1 de marzo de 2021, de <https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F>
- [3] Rodríguez, M. A. (2011, 30 septiembre). *Clean Code: reglas y principios*. Adictos al trabajo. <https://www.adictosaltrabajo.com/2011/09/30/clean-code-reglas-principios/>
- [4] Díez, J. S. (s. f.). *SOLID: los 5 principios que te ayudarán a desarrollar software de calidad*. Profile Software Services. Recuperado 1 de marzo de 2021, de [https://profile.es/blog/principios-solid-desarrollo-software-calidad/#:%7E:text=Los%20principios%20SOLID,Open%2FClosed%20Principle%20\(OCP\)](https://profile.es/blog/principios-solid-desarrollo-software-calidad/#:%7E:text=Los%20principios%20SOLID,Open%2FClosed%20Principle%20(OCP))
- [5] Canive, T. (s. f.). *Metodología SCRUM: ¿qué es y cómo aplicarlo en tu trabajo?* | Sinnaps. Gestor de proyectos online. Recuperado 1 de marzo de 2021, de <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-scrum>
- [6] Standish Group. (2015). CHAOS REPORT 2015 - The Standish Group. [https://www.standishgroup.com/sample\\_research\\_files/CHAOSReport2015-Final.pdf](https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf)

# ANEXOS

- Anexo 1. *Proyecto en GitHub*

Overview   Yours   Active   Stale   All branches   Search branches...

**Default branch**

master Updated 18 days ago by JuanLondonho   Default

**Your branches**

merge_edate1	Updated 2 days ago by ElianaMarcela	0   22	New pull request	✎	🗑
develop	Updated 18 days ago by JuanLondonho	0   0	New pull request	✎	🗑
PedidoseIncidentes	Updated last month by JuanLondonho	38   0	New pull request	✎	🗑
vecinos	Updated 2 months ago by JuanLondonho	73   0	New pull request	✎	🗑
AgruparProductos	Updated 3 months ago by JuanLondonho	98   0	New pull request	✎	🗑

[View more of your branches >](#)

- Anexo 2. SADES antiguo.

Bienvenido Juan Carlos Londoño Gomez

Cédula:

[Consultar](#)

ID Servicio:

Documento:

Nombre:

Mac:

[Limpiar](#)

### Direcciones

Seleccionar	Dirección	Departamento	Municipio	Barrio
<input type="checkbox"/>	CL # 85 - 24	ANTIOQUIA	MEDELLIN	EL DIAMANTE

### Productos

Dirección: CL 80 # 85 A - 24

Producto	Servicio_ID	Tecnología	Estado_Servicio	Estado_Equipo	Velocidad_Siebel	Identificador_Facturacion	Plan_Comercial	Modelo_PNNI	MAC_Datos	MAC_Voz	Marca_PNNI	CMTS_Hostname	CMTS_Nickname	Alias	Tipo_CMTS	CDI	Nodo	Amp	
<input checked="" type="radio"/>	Internet	1-INTERNET	ACTIVO	RECUPERADO	120MB	1-INTERNET	INTERNET 120MB VIENE DE INTERNET 60MB	TG2482	748A0DDE1C8B	748A0DDE1C8C	Arnis	10.14.112.1	MDR-CAR-CMTS-02	MDR-CAR-CMTS-13422_F46	Arnis	CARIBE	13422	010	0

HFC

Acciones de comando

[Refrescar](#)
[Remostrar](#)
[Borrar](#)
[Interfaz](#)

#### Información General Cable Módem

Dirección Cliente	CL # 85 - 24
Nombre CMTS	mdr-car-cmts-02
Dirección MAC	748a 0dde 1c8b
Estado CMTS	<span style="color: green;">Operational</span>
Estado CM	<span style="color: green;">online</span>

```

sflid dir curr aspd sid Sched Prio Macstate Macstst Adminstate Throughput packets/sec
State Type
244993 US act - 13958 BE 1 14498080 10900 0 1832 2
244994 DS act - 0 BE 1 14498080 10900 0 3296 3
mdr-car-cmts-02 show cable modem 748a.0dde.1c8b detail | Include CPE
Current CPEs: 1 Active, 1 Pre-Active, 1 Pre-Active, 1 Pre-Active
Current CPEs: 1 Active, 1 Pre-Active, 1 Pre-Active, 1 Pre-Active
<CPE (ERR) 748a.0dde.1c8b IP=190.201.5.748 Mac=748a0dde1c8b Admin=1 IP=190.201.5.748
<CPE (ERR) 748a.0dde.1c8b IP=190.201.5.748 Mac=748a0dde1c8b Admin=1 IP=190.201.5.748
mdr-car-cmts-02 show cable modem 748a.0dde.1c8b detail | include macs

```

- Anexo 3. SADES actualizado

SADES Nacional
Juan Carlos Londoño Gomez. Log off

Consultar por:

Información Cliente i

**Documento**  **Nombre** JUAN CARLOS LONDONO GOMEZ **Dirección** CL # 85 A **Barrio** EL DIAMANTE **Municipio** MEDELLIN **Departamento** ANTIOQUIA

Información Servicios - Equipos v Vecinos

Fuente	Producto	Servicio_ID	Tecnología	Estado_Servicio	Velocidad	Id_Facturacion	Plan_Comercial
+	SIEBEL	Internet	HFC	ACTIVO	120MB	10000000000000000000	INTERNET 120MB VIENE DE INTERNET 60MB

Consulta HFC

Información Cable Módem v Consola

Información General Cable Módem	
<b>CMTS</b>	mde-car-cmts-02
<b>Alias Nodo</b>	13422_P46
<b>Estado en CMTS</b>	Operational <span style="color: green;">OK</span>
<b>Dirección MAC</b>	748A0DDE1C8B
<b>Referencia Equipo</b>	TG248ZAP2-85
<b>Tiempo Encendido</b>	11d 20h 51m 29s 580ms
<b>IP Navegación</b>	10.145.10.51 48E J
<b>Tráfico Instantáneo</b>	Downstream: 0.00 Mbps Upstream: 0.00 Mbps
<b>Acciones</b>	<div style="display: flex; gap: 5px;"> <div style="border: 1px solid #ccc; padding: 2px 5px;">Refrescar</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">Reiniciar</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">Borrar</div> <div style="border: 1px solid #ccc; padding: 2px 5px;">Interfaz</div> </div>
<b>Máximo Tráfico</b>	Downstream: 137.33 Mbps Upstream: 13.73 Mbps

- Anexo 4. Confirmación de asistencia.

### Contenido del curso:

- Instalación de las herramientas requeridas durante el curso.
- Explicación del patrón MVC y sus beneficios.
- Cómo crear proyecto MVC y revisión de su estructura.
- Qué es un controlador, un modelo y una vista.
- Comunicación entre componentes MVC.
- Implementación de autenticación y autorización utilizando MVC.
- Uso de los DataAnnotations
- Conexión a base de datos.

Durante todo el curso se utilizó GIT como herramienta de versionamiento para el código fuente.

CONFIRMACIÓN DE ASISTENCIA
<p>Nombre completo</p> <p>6 respuestas</p> <p>John Sergio Ramirez Castañeda</p> <p>Saul Ignacio Suarez Rivera</p> <p>Luis fernando Suarez Giraldo</p> <p>Eliana Marcela Vélez Jaramillo</p> <p>Angel Mauricio Arango López</p> <p>Juan carlos Calle Garcia</p>
<p>Correo electrónico</p> <p>6 respuestas</p> <p>John.Ramirez.Castaneda@tigo.com.co</p> <p>saul.suarez@tigo.com.co</p> <p>Luis.Suarez.Giraldo@tigo.com.co</p> <p>Eliana.M.Velez@tigo.com.co</p> <p>Angel.Arango.L@tigo.com.co</p> <p>juan.c.calle@tigo.com.co</p>