



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**INFORME DE TRABAJO  
PROFESIONAL EN ATL**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de

**INGENIERO EN COMPUTACIÓN**

**P R E S E N T A**

VÍCTOR MANUEL RICO CASTILLO

**ASESOR DE INFORME**

M.A. MIGUEL EDUARDO GONZÁLEZ CÁRDENAS



Ciudad Universitaria, Cd. Mx., 2017

# Agradecimientos

- A mi familia quienes siempre me han apoyado y son mi inspiración para seguir adelante en la vida.
- A mis jefes Gloria Morales, Juan Carillo Y Javier Morales quienes me brindaron la experiencia, el tiempo y el apoyo para realizar el trabajo.
- Al Maestro Víctor Pinilla que me brindo un gran apoyo en la realización de este informe y no pudo ser mi asesor por cuestiones de periodo sabático.
- A mi asesor Ing. Miguel González por la ayuda y la orientación para realizar este informe.
- A mis compañeros de trabajo quienes por su amistad y compartir sus conocimientos.
- A la UNAM y la Facultad de Ingeniería por brindarme los conocimientos y las habilidades para poder ingresar al mundo laboral.

# ÍNDICE

ÍNDICE .....	1
Introducción .....	3
Capítulo 1 Descripción de la empresa .....	4
1.1 ¿Qué es ATL? .....	4
1.2 Misión .....	4
1.3 Visión .....	4
1.4 Socios (Partners) .....	4
1.5 Servicios que brinda ATL .....	6
1.6 Cursos que imparte ATL .....	7
1.7 Certificaciones .....	7
1.8 Algunos de los clientes .....	8
1.9 Responsabilidad social .....	9
1.10 Plan “Mi primer empleo” .....	9
1.11 Historia de la empresa .....	9
1.12 Implementación de SCRUM .....	11
1.13 Organigrama .....	12
Capítulo 2 Descripción del puesto de trabajo .....	13
2.1 Perfil de un Consultor Jr. ....	13
2.2 ¿Para qué me contrataron? .....	14
2.3 Capacitación técnica como consultor jr. ....	16
2.4 Desarrollo de habilidades como consultor jr. ....	16
2.5 Actividades asignadas .....	17
Capítulo 3: Descripción de mi participación en la empresa .....	19
3.1 Mantenimiento y soporte .....	19
3.2 Capacitación .....	21
3.3 Desarrollo de Software .....	24
Conclusiones .....	34
Bibliografía .....	35

**Indice de Tablas e ilustraciones**

Tabla 1 Actividades que realizo en ATL.....	18
Ilustración 1 Socios de negocios de ATL.....	4
Ilustración 2 Logotipo de EC Council.....	5
Ilustración 3 Cursos que imparte ATL .....	7
Ilustración 4 Clientes de ATL.....	8
Ilustración 5 Participación social de ATL .....	9
Ilustración 6 Organigrama de ATL.....	12
Ilustración 7 Diseño del proyecto contadores .....	26
Ilustración 8 Diseño interfaz XML .....	29
Ilustración 9 Diseño servicios web primeras pruebas .....	31
Ilustración 10 Diseño servicios web liberación final.....	33

# Introducción

En este informe hago una descripción de las actividades que he realizado en la empresa ATL desde mi ingreso, en 2015. En él detallo brevemente cómo ha sido mi desarrollo profesional, que inició como becario y que gracias a mi desempeño he logrado obtener cada vez más responsabilidades hasta alcanzar las que corresponden a un Consultor Sr. que, de acuerdo a los perfiles de recursos humanos de la empresa, corresponden a un profesional con título universitario.

**El objetivo de este trabajo es describir mis actividades, enfatizando la participación y responsabilidades que he tenido en los proyectos que me han representado un reto y que gracias a la formación que obtuve en la Facultad de Ingeniería, pude cumplir exitosamente.**

El informe consiste de tres capítulos.

En el primero describo a la empresa ATL a partir de su misión y visión estratégicas de sus socios de negocios y de los servicios que brinda.

En el segundo, a partir de la descripción del puesto laboral, muestro cuál ha sido mi evolución profesional dentro de ATL y la respectiva asignación de responsabilidades.

En el tercero presento los proyectos más representativos de mi actividad profesional en los que he participado, ya que los considero como los que me han representado retos y me han formado dentro de la disciplina de la Ingeniería.

# Capítulo 1 Descripción de la empresa

## 1.1 ¿Qué es ATL?

**Aplicaciones Totales en Línea (ATL)** es una empresa 100% mexicana, cuenta con más de 10 años de experiencia en la industria informática, sus principales rubros son la capacitación, consultoría, desarrollo e integración de Sistemas, ofrece certificaciones de diversos fabricantes como *Microsoft*, *EC Council*, *IBM*, *Android ATC*, entre otros.

## 1.2 Misión

Nuestra misión es atender con excelencia los requerimientos informáticos de nuestros clientes para cumplir con altos estándares de calidad y cubrir al 100% las expectativas de nuestros clientes, pensando siempre en costos altamente competitivos.

## 1.3 Visión

La visión de ATL es ser la empresa mexicana líder del mercado en soluciones de Tecnologías de la Información (TI) ofreciendo excelencia, calidad y servicio a nuestros clientes, empleados y socios. Manejando nuestros negocios siempre con honestidad.

## 1.4 Socios (*Partners*)

Algunos socios de ATL son IAMCP, EC-Council, Android ATC, etc. (véase ilustración1)



Ilustración 1 Socios de negocios de ATL

### 1.4.1 ¿Por qué IAMCP?

*International Association of Microsoft Channel Partners (IAMCP)* es una asociación que representa a los mejores socios de Microsoft y tienen presencia mundial. Estar asociados con ellos permite generar oportunidades para poder lograr un crecimiento acelerado, ya

que a través de la interacción con otros miembros de la IAMCP, se desarrollan negocios de manera conjunta.

La Ing. Gloria Morales, directora general de ATL, es parte de la mesa directiva de la IAMCP México. Se encuentra en el puesto de *WOMAN IN TECHNOLOGY* el cual diseña programas para invitar a que más mujeres se integren en el mundo de las TI.

### 1.4.2 ¿Por qué EC-Council?

Actualmente el uso de Redes de datos es necesario en cualquier organización que quiera tener una presencia notable en el Mercado. Cada día surgen nuevas amenazas que ponen en riesgo las operaciones de las empresas, el *International Council of E-Commerce Consultants (EC-Council)* es un organismo de certificación técnica con presencia mundial que se encuentra en 140 países, cuenta con 150,000 profesionales certificados. Organiza los eventos *Hacker Halted, TakeDownCon, Global Cyberlympics*, etc. Es el encargado de las certificaciones con mayor prestigio en el mercado de la seguridad informática, entre las más populares se encuentran *Certified Ethical Hacker, Security Analyst, Forensic Investigator y Secure Programmer*.



Ilustración 2 Logotipo de EC Council

ATL ha sido reconocida con el premio *CIRCLE OF EXCELLENCE* (vease ilustracion 2) en los años 2011, 2012 y 2014. Premio que se les da a los centros que demuestran un gran compromiso al proporcionar programas excepcionales de formación y certificación en el área de la seguridad informática.

### 1.4.3 ¿Por qué Android ATC?

En los últimos años el uso de los dispositivos móviles ha ido en aumento exponencialmente, desde hace algunos años supero la venta de los equipos de cómputo y es la opción preferible para conectarse a Internet. Existen dos sistemas operativos líderes que dominan el mercado, *iOS* de *Apple* y *Android* de *Google*, siendo *Android* el que se encuentra en un mayor número de dispositivos móviles. *Android Advanced Training Consultants (Android ATC)* es el encargado de proporcionar certificaciones y cursos a través de socios para mejorar las habilidades en *Android*.

ATL, por ser socio de *Android ATC* puede impartir los cursos de *Java Fundamentals for Android, Android Application Development, Android Security Essentials y Monetize Android Application*, así como las certificaciones *Android Certified Application Developer, Android Certified Application Engineer y Android Certified Trainer*.

#### 1.4.4 ¿Por qué Pearson VUE?

En la actualidad existe una gran competitividad entre los profesionales, es necesario que éstos aumenten su valor para el mercado. Una forma de aumentar el valor en el mercado es comprobando que se tienen ciertos conocimientos de un área en específico. *Pearson VUE* es el líder mundial en pruebas realizadas en computadora para la certificación en diversas áreas.

*Pearson VUE* llegó a ser el líder gracias a que los centros autorizados tienen que cumplir con ciertos estándares de seguridad para evitar los intentos de fraude y así mantener el nivel de calidad en las certificaciones, gracias a esto las personas que no cuenta con los conocimientos, no aprueba el examen.

Algunos líderes tecnológicos que ocupan los servicios de *Pearson VUE* para aplicar sus certificaciones son *Microsoft, Oracle, Cisco, etc.* *Pearson VUE* no solo se enfoca en el área de TI, las áreas que ocupan los servicios de *Pearson VUE* son Academias, Finanzas, Gobierno, Salud, Militar, etc. ATL es un centro autorizado para presentar exámenes de *PERSON VUE*.

### 1.5 Servicios que brinda ATL

- Capacitación oficial de los principales líderes del mercado en TI
- Capacitación a la medida de las necesidades del cliente.
- Detección de necesidades de capacitación.
- Cursos orientados a la productividad.
- Desarrollo de sistemas con altos estándares de calidad.
- *Outsourcing* de instructores-consultores certificados por proyecto o por evento.
- Análisis, diseño e implementación de servicios de red.
- Análisis diseño e implementación de aplicaciones cliente/servidor, en capas, distribuidas y/o empresariales (GUI, web y apps).
- Servicios de consultoría y soporte técnico.
- Renta de salas para cursos.
- Aplicación de exámenes de certificación.



## 1.6 Cursos que imparte ATL

ATL cuenta con un gran catálogo de los cursos que puede impartir (véase ilustración 3).



Ilustración 3 Cursos que imparte ATL

## 1.7 Certificaciones

El personal de ATL cuenta con diversas certificaciones, algunas de ellos son:

### 1.7.1 Microsoft

- MCSE (*Microsoft Certified Systems Engineer*)
- MCSD (*Microsoft Certified Solution Developer*)
- MCAD (*Microsoft Certified Application Developer*)
- MCTS (*Microsoft Certified Technology Specialist*)
- MCPD (*Microsoft Certified Professional Developer*)
- MCDBA (*Microsoft Certified Database Administrator*)

### 1.7.2 Java y Oracle

- DBA
- SCJA
- SCJP
- SCWCD
- OCA
- *Oracle Database Expert, Associate y Professional*

### 1.7.3 EC-Council

- *Ethical Hacker*
- *Forensic Investigator*
- *Security Specialist*
- *Network Security Administrator*
- *Secure Programmer (Java, .NET).*

### 1.7.4 Android ATC

- *Certified Application Developer*
- *Certified Application Engineer*
- *Certified Trainer*

## 1.8 Algunos de los clientes véase ilustración 4



Ilustración 4 Clientes de ATL

## 1.9 Responsabilidad social

ATL se ha unido en diversas ocasiones a la campaña "Juguetón... un regalo una sonrisa" (véase ilustración 5), para ayudar a incrementar el número de sonrisas y de ilusiones, y lograr que cada año sean menos los niños que no conocen la magia de los Reyes Magos, por lo que aportaremos un porcentaje de nuestras ventas de los meses de octubre, noviembre, diciembre y principios de enero a esta noble causa.



*Ilustración 5 Participación social de ATL*

## 1.10 Plan “Mi primer empleo”

ATL forma parte del plan “Mi primer empleo”; este plan lo maneja la Secretaría de Trabajo y Fomento al Empleo de la Ciudad de México y la Dirección General de Empleo, Capacitación y Fomento Cooperativo de la Ciudad de México. El plan consiste en que los jóvenes entre 16 y 29 años de edad con poca o sin ninguna experiencia se integren al sector productivo, comiencen a desarrollar sus competencias y adquirir experiencia laboral. Este plan brinda la oportunidad a los jóvenes para entrar al mundo laboral.

## 1.11 Historia de la empresa

### 1.11.1 ¿Cómo nació ATL?

ATL se fundó en el año 2003 en México D.F.; el país pasaba por una crisis financiera, los fundadores contaban con un empleo en otra compañía, pero no recibían el pago por sus servicios. Trabajaron tres meses sin recibir pago y tomaron la decisión de renunciar para emprender en algo propio. Así que decidieron fundar ATL con los pocos ahorros que tenían.

### 1.11.2 ¿Quién fundó ATL?

ATL fue fundada por tres personas, una pareja de recién casados y un padre con trillizas de un año. Los tres son destacados ingenieros en diversas áreas y con distintivas habilidades lo que le ayudó al funcionamiento de la empresa: la Ing. Gloria Morales en el área de desarrollo de aplicaciones de software, el Ing. Juan Carrillo en el área de capacitación en TI y el Ing. Javier Morales en el área de infraestructura de hardware.

### **1.11.3 Algunos clientes**

#### **1.11.3.1 Primer cliente, La Moderna**

En el año de 2003 ATL ganó una licitación para desarrollar un proyecto con La Moderna (que en ese momento no supo que fue el primer cliente de la nueva empresa). Este proyecto consistió en una interfaz entre una báscula para camiones de transporte y el sistema interno de la empresa; su función fue registrar el peso de los camiones en la báscula, a la entrada y la salida de la zona de carga, e introducía la información al sistema para llevar un control de los insumos.

El principal reto que tuvo este proyecto fue trabajar con una impresora que tenía controladores obsoletos; la empresa no estuvo de acuerdo en actualizar la impresora y ATL tuvo que crear controladores funcionales. Este aspecto aparentemente intrascendente estuvo a punto de echar abajo el trabajo realizado. Este sistema se encontró en producción por casi 10 años cumpliendo así su objetivo. En el año 2016, ATL regresó a trabajar con La Moderna.

#### **1.11.3.2 Instituto Mexicano del Petróleo**

ATL, además de realizar trabajos para desarrollo de software también capacita en TI. En 2007, el IMP requería un curso sobre Visual Studio .NET, en donde se tenían que capacitar a los alumnos en las mejores prácticas de desarrollo. Así surgió el “El Taller de Visual Studio NET Orientado a la Productividad” y su descendiente “El Taller de Java Orientado a la Productividad”.

#### **1.11.3.3 Grupo Escato**

En 2011, ATL trabajó con en conjunto Grupo Escato para participar en una licitación consistente en crear en sistema de revisión fiscal aleatoria en las aduanas que incluía el mobiliario (*display*) donde se instaló el semáforo: al presionar un botón, aleatoriamente si la luz es roja se revisa en el envío y si es verde sigue adelante.

El principal reto de este sistema fue la liberación de la aplicación en todas las aduanas del país. El reto se resolvió reuniendo un grupo de personas que fueron capacitadas en la instalación del sistema en las diferentes aduanas de tal forma que se logró cumplir con los tiempos establecidos por el cliente. El proyecto culminó de forma exitosa, actualmente siguen los *displays* de leds de este proyecto funcionando en las aduanas.

ATL ha realizado una gran cantidad de desarrollos y capacitaciones con altibajos, siempre aprendiendo para seguir mejorando. A pesar de ser una empresa pequeña ha trabajado a lo largo del país e incluso en el extranjero. Cuenta con personal muy capaz y fuerte en las disciplinas, tecnologías y lenguajes más usados en el mercado. No sólo se cuenta con la teoría, sino también con su aplicación, ya que hasta el personal más joven se ve inmerso en los desarrollos, inicialmente con un seguimiento y guía hasta que sean autosuficientes.

## **1.12 Implementación de SCRUM**

SCRUM es un *framework* (Marco de trabajo para el desarrollo) ágil para administrar proyectos. Emplea un enfoque adaptativo e iterativo para desarrollar y administrar el proyecto. ATL ha empleado Scrum de forma exitosa en los proyectos de desarrollo y apoyamos a nuestros clientes en la adopción de este framework.

## 1.13 Organigrama

La ilustración 6 representa el organigrama de ATL.

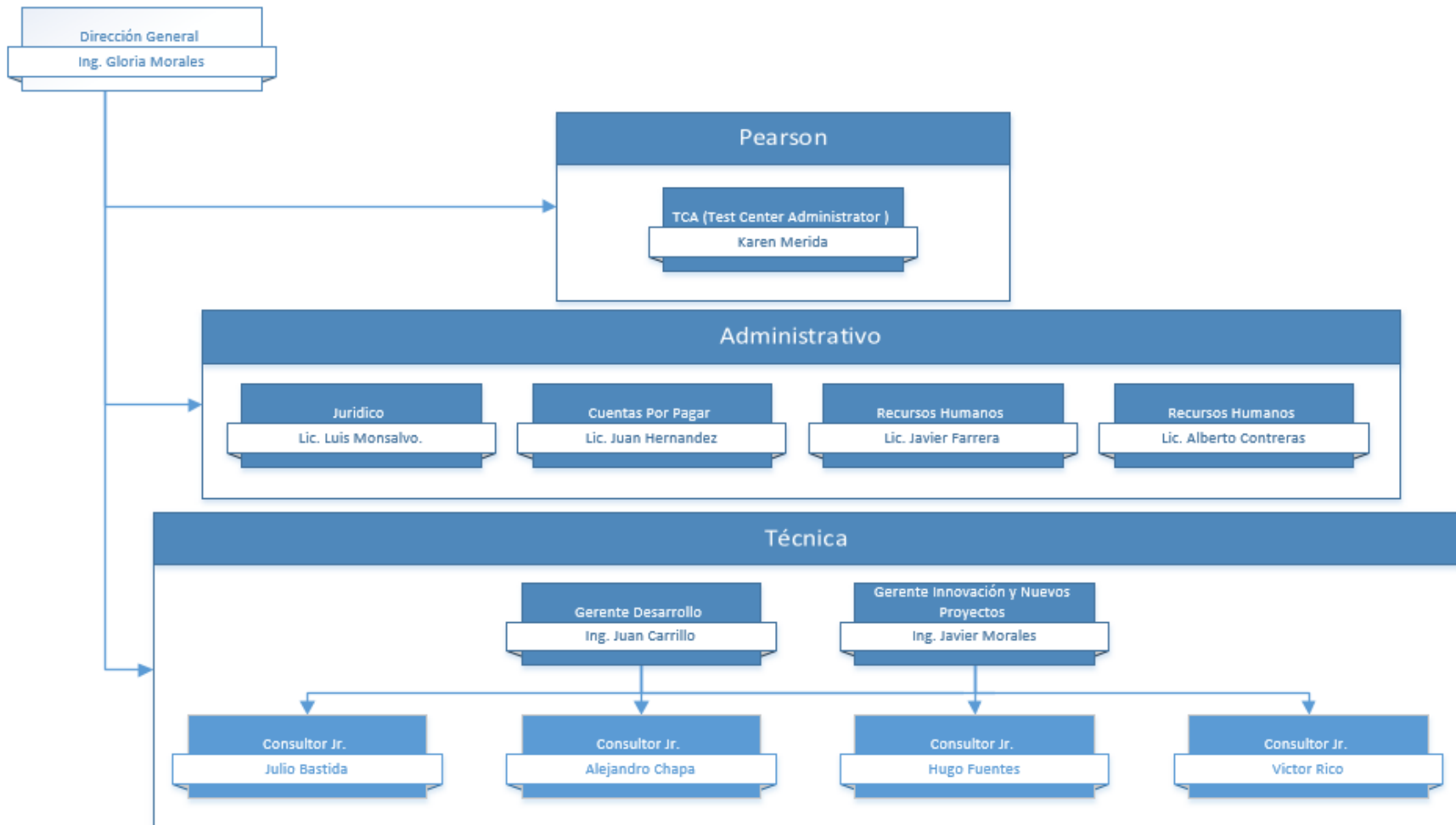


Ilustración 6 Organigrama de ATL

# Capítulo 2 Descripción del puesto de trabajo

## 2.1 Perfil de un Consultor Jr.

### 2.1.1 Datos generales

El perfil corresponde a un estudiante o egresado de carreras de ingeniería de sistemas o carreras afines, o gracias a sus conocimientos se encuentra capacitado para realizar programas en uno o más lenguajes de programación, impartir capacitación, realizar mantenimiento y soporte de equipos y redes.

### 2.1.2 Propósito del puesto

El propósito de un consultor Jr. es participar integralmente en los proyectos desarrollados por la empresa y de acuerdo con la planeación del mismo, ejecutar las actividades encomendadas (mantenimiento y soporte, desarrollo y capacitación) para cumplir con los requerimientos establecidos bajo la supervisión de un consultor Sr.

### 2.1.3 Descripción del puesto

El consultor Jr. se encarga de traducir las especificaciones que proporciona el analista en un sistema informático (software) por medio de la programación en diferentes lenguajes, se mantiene actualizado en temas de TI para poder dar capacitación, da mantenimiento y soporte a los equipos de cómputo; todo esto dependiendo de las características del proyecto al que sea asignado.

Además, debe poseer valores y actitudes tales como iniciativa, creatividad, trabajo en equipo, responsabilidad, autoaprendizaje, liderazgo, etc.

### 2.1.4 Conocimientos

En el nivel Jr., para desarrollar eficientemente estas actividades debe poseer al menos los siguientes conocimientos: arquitectura de computadoras, redes, instalación de software, conocimientos en lenguajes de programación, programación orientada a objetos y habilidades para la enseñanza. Adicionalmente poseer al menos una certificación en algún lenguaje de programación y la experiencia laboral no es necesaria.

### 2.1.5 Funciones específicas

- Trasladar las especificaciones hechas por el analista para poder codificar el programa.
- Participar en el diseño del sistema.
- Codificar el programa.
- Realizar pruebas y corrección de errores.
- Elaborar gráficos y diagramas para describir y determinar en qué secuencias habrá que proceder al registro y tratamiento de los datos.
- Desarrollar y proporcionar documentación detallada sobre los programas informáticos, utilizando para ello diversos lenguajes de programación.
- Mantener actualizados los programas.
- Dominar manuales de las capacitaciones a impartir.
- Crear o modificar manuales.
- Mantenerse actualizado en los temas de las capacitaciones.
- Armado de equipos.
- Configuración de equipos.
- Configuración de redes.
- Otras más definidas por el proyecto que se asigne

## 2.2 ¿Para qué me contrataron?

Ingresé a ATL como becario en el 2015, en ese momento la empresa buscaba personas que contaran con un perfil conformado de conocimientos de algún lenguaje de programación (de preferencia C# o PHP), nociones de Programación Orientada a Objetos y de creación y análisis de algoritmos. El objetivo de la empresa era ingresar a las personas seleccionadas en un plan de capacitación que cubría diferentes áreas de TI, para posteriormente implementar los conocimientos adquiridos en un proyecto real bajo la supervisión del gerente de desarrollo y el gerente de innovación (ambos consultores Sr). Yo fui seleccionado para entrar al plan de capacitación.

### 2.2.1 En qué consistió el plan de capacitación

Entre las primeras capacitaciones que recibí se encuentran *UML*, *POO*, *C#*, *JAVA*, Creación de páginas *WEB* con *JAVA WEB*, *.NET*, base de datos *Oracle* y *SQL server*.

#### 2.2.1.1 ¿Por qué UML?

*Unified Modeling Language (UML*, lenguaje unificado de modelado) es el lenguaje más conocido para modelar sistemas de software permitiendo visualizar gráficamente el modelo del sistema (Rumbaugh, Jacobson, y Booch, 2000). Debido a su gran importancia en el área de la Ingeniería de Software se decidió capacitarme en los conceptos de *UML*.



¿Cuál fue la diferencia entre lo aprendido en la Facultad y en ATL? Las diferencias principales es que se nos proporcionó un caso real con un problema a solucionar, nos enseñaron como se crean los diagramas UML; la mayoría de los libros nos dicen que significa cada elemento, no como se abstrae la información del mundo real a los diagramas.

#### **2.2.1.2 ¿Por qué POO?**

La Programación Orientada a Objetos (POO) es un paradigma de desarrollo que comenzó a popularizarse en los años 1980 reemplazando a la programación estructurada e integrando nuevos conceptos como herencia, polimorfismo, eventos, etc. Sus principales características son la reutilización de código, que permite abstraer los elementos involucrados en el proyecto para generar un modelo cercano a como se trabaja en el mundo real (Durán, Gutiérrez y Pimentel, 2007).

¿Cuál fue la diferencia entre lo aprendidos en la Facultad y en ATL? En el plan de estudios que yo curse no se encontraba la materia de POO; si querías aprender este paradigma era necesario tomar cursos externos. En mi caso yo tome un par de cursos de PROTECO (Programa de Tecnología en Cómputo) que se imparten en la Facultad de Ingeniería de la UNAM.

#### **2.2.1.3 ¿Por qué C#?**

C# es un lenguaje estandarizado de POO que forma parte del framework .NET (Microsoft 2017). Es un lenguaje que se encuentra entre los 5 más populares del mercado, se ubica en estas posiciones gracias a su poder y estabilidad (Claudio, 2016).

Una vez más, lo básico de C# lo aprendí en los cursos de PROTECO, que se imparten en la Facultad de Ingeniería de la UNAM ya que éste lenguaje no se incluía en el plan de estudios que me correspondió cursar.

#### **2.2.1.4 ¿Por qué JAVA?**

Java es un lenguaje de POO propiedad de *Oracle*, su principal característica es que es multiplataforma; su lema es “Programa una vez, ejecuta donde sea” (Oracle,2017), es un lenguaje que se encuentra entre los 3 más populares del mercado (Claudio, 2016).

#### **2.2.1.5 ¿Por qué la Base de Datos de Oracle?**

Oracle Data Base es un motor de base de datos relacional desarrollado por Oracle, es el motor más usado del mundo, puede ejecutarse en un gran número de plataformas desde una PC estándar hasta una supercomputadora, implementa un lenguaje de programación que potencializa su efectividad (Oracle, 2017).

### 2.2.1.6 ¿Por qué SQL Server?

SQL Server es un motor de base de datos relacional desarrollado por Microsoft, puede ejecutarse en SO Microsoft. Debido a que tiene una gran variedad de versiones, las empresas pueden comprar la versión que más se adapte a sus necesidades. También cuenta con una versión gratuita que trabaja de forma local (Microsoft, 2017).

Me desempeñé como becario por alrededor de uno y en virtud de que cumplí con las expectativas de la empresa, a inicios de 2016 fui contratado como consultor jr.

Mi capacitación ha sido continua desde entonces, tanto en el área técnica como en el desarrollo de otro tipo de habilidades en las áreas técnicas SCRUM, PSP, etc. También lo ha sido para desarrollar habilidades para hablar en público, finanzas y liderazgo.

## 2.3 Capacitación técnica como consultor jr.

### 2.3.1 SCRUM

*Scrum* es un *framework* ágil para la administración de proyectos. Las principales ventajas al trabajar con él son:

- Entrega mensual de liberaciones tangibles del proyecto.
- Flexibilidad y adaptación a nuevos requerimientos.
- Gran productividad y calidad.
- Un equipo motivado y auto gestionado.
- Gran comunicación entre el cliente y el área de desarrollo.
- Mejora continua.

Obteniendo así un mejor producto de *software* (*Agile Academy*, 2015).

### 2.3.2 PSP

El *PSP* (*Personal Software Process*) son un conjunto de prácticas que nos ayudan a controlar, administrar y manejar el desarrollo de software. Está estructurado por niveles, cada uno de ellos mejorando las prácticas de los desarrollos anteriores. PSP nos brinda información para determinar nuestras fortalezas y debilidades (*Agile Academy*, 2015).

## 2.4 Desarrollo de habilidades como consultor jr.

### 2.4.1 Finanzas personales

No sólo hay que ganar dinero, es necesario saber cómo administrarlo. En el curso nos mostraron técnicas para la administración del dinero, generar planes de ahorro a futuro, evitar gastos innecesarios y realizar planes para pagar deudas.

## **2.4.2 La palabra**

En la actualidad “hablar” es una habilidad muy importante que puede hacer una gran diferencia. El curso de la palabra es un curso que nos apoya a mejorar la habilidad del habla, empleando técnicas que permiten manejar el cuerpo, la voz y los sentimientos para hacer frente ante diversas situaciones.

## **2.4.3 Liderazgo efectivo**

Influir de manera positiva a las personas, trae como consecuencia que éstas te sigan por decisión y no por imposición, en el curso se nos mostró que ser un líder efectivo es aquel que inspira tranquilidad y fuerza para lograr las cosas.

## **2.5 Actividades asignadas**

Posteriormente a las primeras capacitaciones se me asignaron las siguientes actividades:

### **2.5.1 Mantenimiento y soporte**

Como ATL es un centro de capacitación se imparte una gran variedad de cursos. Una de las primeras actividades que me fueron asignadas fue el armado y preparación de las salas donde se impartieron los cursos, esta actividad la sigo realizando y varía de complejidad dependiendo del curso o cursos que se vayan a impartir y el número de participantes que se tendrá.

### **2.5.2 Capacitación**

Los primeros cursos de capacitación que se me asignaron fueron Java y Visual Basic .NET. Al ser mis primeros cursos sólo impartí la parte básica.

### **2.5.3 Desarrollo**

El primer proyecto de desarrollo que se me asignó tuvo una duración aproximada de 6 meses. En este proyecto, a partir de casos de usos previamente diseñados, realicé el diseño de la base de datos y programé la sección correspondiente.

Al concluir con algún proyecto se me asignan nuevas tareas a realizar, por ejemplo, cuando se terminó el proyecto de desarrollo me involucraron en las pruebas (*Testing*) en un proyecto más con el fin de verificar la funcionalidad de la aplicación antes de ser liberada a producción y asegurarse así que no presentara comportamiento inesperado.

Las tareas que realizo al día de hoy, se encuentran en la siguiente tabla (véase Tabla 1).

<b>Área</b>	<b>Descripción de las actividades</b>
Mantenimiento y soporte	Armado de equipos, Instalación, configuración y actualización de software, Creación y configuración de máquinas virtuales.
Capacitación	Comprensión de manuales. Creación de manuales y laboratorios prácticos, Realizar certificaciones, mantenerme actualizado en los temas que se impartirán. Entrevistas a futuros becarios.
Desarrollo de software	Pruebas

*Tabla 1 Actividades que realizo en ATL*

# Capítulo 3: Descripción de mi participación en la empresa

## 3.1 Mantenimiento y soporte

La complejidad de estas tareas es variable; en el mejor de los casos es armar, colocar el equipo e instalar un software. En otras ocasiones resulta más complejo ya que se debe crear desde cero un ambiente virtual (generalmente en *VMware player*) siguiendo una metodología para reducir las posibles fallas y reducir los riesgos.

La metodología que sigo es la siguiente:

1. Análisis de la infraestructura mínima que se requerirá para el curso.
2. Buscar el equipo que cumpla con los requerimientos de hardware.
3. Armar el equipo.
4. Realizar la instalación del sistema operativo (Si es requerida), en algunos cursos se especifica el sistema operativo necesario para el buen funcionamiento de los *MOC (Microsoft Official Courses)*.
5. Agregar los archivos requeridos para el curso.
6. Instalar el *software* requerido para el curso.
7. Configurar el equipo.
8. Verificar el funcionamiento del equipo.
9. Dar soporte (si es necesario).

Uno de los ejemplos más significativos de esta actividad, en la que estuve involucrado es el siguiente proyecto

### 3.1.1 Necesidades del cliente

El cliente requería una capacitación con temas que involucraba 4 *MOC* sobre virtualización.

### 3.1.2 Situaciones

Los *MOC* cuentan con especificaciones que indican cuáles deben de ser las características que deben tener el equipo (*hardware* y *software*) y las instrucciones para configurarlo. Los *MOC* trabajan con máquinas virtuales que corren sobre *Hyper-V*; la situación que se tuvo fue que los *MOC* ocupan las mismas máquinas virtuales base para diferentes cursos, por lo que era necesario saber que archivos se iban a copiar, cuáles serían las instrucciones y cuándo se tendrían que ejecutar para evitar que la configuración de un *MOC* no afectara el funcionamiento de otro.

### **3.1.3 ¿Quién estuvo involucrado en el proyecto?**

En este proyecto estuvimos involucrados los 4 consultores Jr., el gerente de innovación y el gerente de desarrollo.

### **3.1.4 Roles de cada participante**

Consultores Jr.: seleccionar configurar y probar el equipo, instalar el software necesario.

Gerente de Innovación: Fui el instructor asignado al curso.

Gerente de desarrollo: Creó el manual para la instalación de los MOC.

### **3.1.5 Análisis de la infraestructura (paso 1)**

El Gerente de Desarrollo analizó las instrucciones de cada *MOC* para saber cómo se tendría que realizar la instalación, y estableció los requerimientos mínimos de cada equipo. De tal forma que generó un nuevo manual temporal de instalación y los nuevos requerimientos mínimos del hardware.

### **3.1.6 Búsqueda el equipo que cumpla con los requerimientos de hardware (paso 2)**

Los consultores jr. buscamos los equipos que cumplieran con los nuevos requerimientos mínimos de hardware:

- Procesador con *Intel Virtualization Technology (Intel VT)* o *AMD Virtualization (AMD-V)*.
- 300 GB libres en disco duro.
- 16 GB o más en memoria RAM.
- Adaptador de Red.

Después del armado del equipo (paso 3) se verificó que se tuviera el sistema operativo indicado, que en este caso fue Windows Server 2012 R2; de no ser el caso debió instalarse (paso 4).

### **3.1.7 Agregar los archivos requeridos para el curso (Paso 5)**

Se copiaron en cada equipo los archivos necesarios para ejecutar las máquinas virtuales, así como los manuales y laboratorios que cada MOC.

### **3.1.8 Instalar el software requerido para el curso y configurar el equipo (paso 6 y 7)**

Una vez que ya se tenían los archivos en el equipo fue necesario habilitar el *Hyper-V* (Programa para virtualización). Después de habilitarlo se instalaron y abrieron cada una

de las máquinas virtuales, instalando y configurando el software requerido en cada una de ellas:

- Rearmar el SO (que el SO no muestre mensaje que no es una copia original)
- Rearmar Office para los MOC que lo necesitaban (Habilitar por treinta días la licencia de Office).
- Configurar la Red.

### **3.1.9 Verificar el funcionamiento del equipo (paso 8)**

Una vez ya instalado todo el ambiente para dar los cursos se realizaron algunas pruebas para verificar que todo estuviera funcionando correctamente:

- Verificar que exista conexión entre la máquina Host y las máquinas virtuales.
- Verificar que exista conexión entre las máquinas virtuales.
- Verificar que el Software instalado se ejecute de forma correcta.

### **3.1.10 Dar soporte (paso 9)**

No fue necesario dar soporte ya que no se tuvo ningún inconveniente durante el curso.

He participado en 3 proyectos de este estilo, que consisten en más que leer un manual y hacer una la instalación. También me ha tocado crear máquinas virtuales donde tengo que tomar la decisión de qué sistema operativo elegir, la capacidad de disco duro y la memoria RAM; estas configuraciones deben tener características de las máquinas reales que comúnmente ocuparan los participantes.

## **3.2 Capacitación**

En varias ocasiones me ha tocado ser instructor de una serie de cursos, principalmente de programación y de la paquetería de office. Cuando me toca ser el instructor de un curso, si ya se cuentan con los manuales es necesario que los revise y se realicen las prácticas indicadas en él para verificar que las prácticas estén ayudando a la comprensión de los temas y que todo lo pedido se haya visto en los temas expuestos.

Algunas veces los cursos que piden los clientes son personalizados con temas en específico y es necesario crear o actualizar los manuales que se entregarán, así como crear los laboratorios para asegurar la comprensión de los temas.

Como es sabido, las áreas relacionadas con las TI están en constante crecimiento y evolución por lo que de vital importancia que me mantenga actualizado en la materia. Es necesario contar con una certificación oficial que garantice que se cuenta con conocimientos sólidos de cada uno de los cursos que serán impartidos

La metodología que se sigue para una capacitación es la siguiente:

1. Analizar las necesidades del cliente.
2. Buscar o diseñar un temario que satisfaga las necesidades del cliente.
3. Proponerle al cliente un plan de capacitación.
4. Elegir o crear los manuales de la capacitación.
5. Hacer los laboratorios.
6. Revisar y comprender el manual.
7. Impartición de la capacitación.

A continuación, describo una capacitación personalizada que pidió una empresa de telefonía móvil a ATL.

### **3.2.1 Necesidades del cliente**

El cliente requería que un grupo de aproximadamente 45 desarrolladores fueran capacitados en Java, Programación Orientada a Objetos, el manejo de Hilos y socket y estructuras de datos y que realizaran una aplicación para interactuar con una base de datos Oracle (esta aplicación no debería ocupar herramientas gráficas para crearse).

### **3.2.2 Situaciones**

La mayoría de los estudiantes contaba con 5 a 20 años de experiencia en el área desarrollo; sin embargo, el lenguaje que utilizaban tenía un paradigma estructurado, como Fortran; desconocían en su totalidad la POO. Asimismo, hubo que contender con los siguientes detalles:

- Desconocían el lenguaje de Java.
- Algunos fueron obligaron a capacitarse.
- No se podría tener un solo grupo, por el tamaño de las aulas.
- La capacitación se dio en sábado, día que ellos descansaban.
- Era una gran cantidad de temas en comparación a la duración del curso.

### **3.2.3 ¿Quién participó en el proyecto?**

En este proyecto estuvieron involucrados 3 Consultores Jr., la directora general, el gerente de innovación y el gerente de desarrollo.



### **3.2.4 Roles de los involucrados**

Consultores Jr.: Instructores del curso

Directora general: Evaluar a los instructores y el material del curso.

Gerente de innovación: Actualizar y verificar los conocimientos de los instructores, crear el material para el curso.

Gerente de desarrollo: Evaluar a los instructores y el material del curso.

### **3.2.5 Propuesta hecha al cliente**

La directora de la empresa y el gerente de innovación hicieron el análisis de las necesidades que tenía el cliente (paso 1), una vez terminado el análisis diseñaron el plan de capacitación (paso 2). El plan propuesto al cliente consistió en dividir el grupo de 45 personas en 3 grupos de 15, se les impartió el curso de manera paralela a cada grupo, se propuso un temario con los temas más importantes y específicos que se dieran en el tiempo indicado, al final del curso los participantes crearían una aplicación que realizara las siguientes acciones alta, baja, modificación y eliminación de registros de una base de datos. Habría tres instructores que impartirían los mismos temas, en cada grupo (paso 3).

### **3.2.6 Preparación del manual**

El gerente de innovación fue el encargado de realizar el manual y los laboratorios, que se ocuparían durante la capacitación; A mí me tocó crear la aplicación que interactuara con la base de datos que se le solicitaría elaborar a los participantes, de conformidad con la petición del cliente (no utilizar herramientas gráficas de arrastre). Para lograr este objetivo tomé la decisión de contemplar en el diseño de la aplicación el nivel de complejidad del curso y el perfil de los participantes, así como el tiempo disponible (paso 4 y 5).

Mi conclusión fue una interfaz gráfica simple, que funcionaría como una agenda, en la cual se pudieran dar de alta, baja, modificar y visualizar a los usuarios.

El patrón de diseño que seguí fue MVC; también decidí distribuir la aplicación en capas para que los participantes pudieran observar las ventajas de la POO, como la reutilización del código.

### **3.2.7 Capacitación de los instructores (paso 6).**

Debido a que los conocimientos deberían ser uniformes en los tres grupos que conformaron la capacitación se decidió que los tres instructores (consultores Jr., yo entre ellos) tomáramos un curso de Java impartido por el Gerente de Innovación y nuevos

proyectos, sin importar los conocimientos previos ya adquiridos en dicho lenguaje. Este curso contempló los requerimientos indicados por el cliente, particularmente en la POO

Sobre este último aspecto, para asegurar que la comprensión de este paradigma fuera lo más uniforme posible en los tres grupos se elaboró un ejemplo teórico conformado con los conceptos fundamentales de esta tecnología: herencia, sobrecarga, sobre escritura y polimorfismo.

La dinámica que se siguió fue que cada instructor impartiera un tema de la POO al resto del equipo (los 2 instructores restantes, el gerente de desarrollo, gerente de innovación y directora general) para así poder hacerle preguntas y observaciones sobre lo expuesto.

### **3.2.8 Impartición del curso (paso 7)**

El curso se impartió en 5 sesiones de 8 horas los sábados, sin ningún contratiempo.

### **3.2.9 Resultados**

Al terminar el curso se realizó una encuesta de calidad para ver la satisfacción de los participantes, en esta encuesta las calificaciones generales de los tres instructores fueron de buena a excelente. Pero la mejor métrica para medir la satisfacción del cliente es que sigue solicitando cursos a ATL.

## **3.3 Desarrollo de Software**

En el área en donde estoy más involucrado es desarrollo de software ya que satisface mis deseos de desarrollo profesional. Por lo anterior, he buscado participar en diversos proyectos en diferentes etapas del ciclo de vida del software; en algunos proyectos sólo lo he hecho en una etapa y en otros desde que se concepción hasta su liberación.

A continuación, describo mi participación en los proyectos que me resultan más significativos.

### **3.3.1 Proyecto contadores**

Éste fue el primer proyecto de desarrollo en el que estuve involucrado en ATL.

#### **3.3.1.1 ¿Quién participó?**

El equipo de trabajo estaba conformado por el gerente de desarrollo y dos becarios (En ese momento yo era becario).

#### **3.3.1.2 Roles de los involucrados**

Gerente de desarrollo (líder del proyecto): supervisar y administrar el proyecto.

Becarios: diseño de la base de datos y programación. Yo fui el encargado del diseño de la base de datos.

### **3.3.1.3 Necesidades de cliente**

Este proyecto consistió en una aplicación web encargada de controlar los cursos impartidos por una asociación, la afiliación de sus socios y su forma de pago incluyendo la facturación electrónica. Este proyecto cumplía un año sin avanzar por lo que la empresa recurrió a ATL para completarlo.

La aplicación debía de contar con lo siguiente:

- Catálogo de instructores, incluyendo información personal, fotografía y CV.
- Catálogo de los cursos que se podían impartir, incluyendo nombre, tipo, clave, descripción, puntos, etc.
- Catálogo de sedes.
- Catálogo de clasificación de los cursos.
- Catálogo de los tipos de asociados.
- Los cursos programados, a partir del catálogo de cursos, donde se pudiera especificar el número de participantes, el horario, la sede, etc.
- Los alumnos inscritos a cada curso.
- Cantidad de puntos por alumno.
- Pagos
- Facturas
- Membresías
- Constancias
- Exámenes.

### **3.3.1.4 Mi participación en el proyecto**

La primera tarea que me asignaron fue asimilar los casos de uso que ya había sido analizados y mejorados previamente. A mí me correspondió el diseño de la base de datos de lo relacionado con la administración de los cursos.

Una vez que comprendí los casos de uso proseguí a realizar el diseño de la base de datos. La base de datos con la que trabajé fue SQL Server 2008, por lo tanto, sería una base de datos relacional. Recordando mis clases de base de datos en la Facultad, primero creé el modelo entidad relación y después lo pasé al modelo relacional. Después de algunas observaciones hechas por el líder del proyecto comencé a generar los scripts para la creación de las tablas. Creé aproximadamente 25 tablas para administrar todo lo relacionado a la administración de los cursos, cada tabla contaba con una llave primaria auto incremental, así como la fecha de la creación y modificación de registro, llaves foráneas (si eran necesarias) para poder controlar y asegurar la integridad de los datos.

Creé vistas para las sentencias más recurrentes, así como procedimientos almacenados para cada tabla para poder manejar la inserción, la modificación y la eliminación de los datos; esto para aumentar la seguridad y el rendimiento de la aplicación.

Al terminar la creación de las tablas y los elementos de la base de datos comencé el desarrollo de la aplicación. La tecnología para desarrollar la aplicación fue Web Form .NET framework 4.0 con C# como lenguaje de programación; el proyecto estaba dividido en capas para aumentar su seguridad y el patrón de diseño que el cliente pidió fue Modelo - Vista – Presentador.

El patrón de diseño Modelo – Vista – Presentador (MVP) deriva del Modelo-Vista-Controlador (MVC) con la diferencia que se busca tener una mayor separación de la lógica de negocio respecto a la lógica de la interfaz gráfica para facilitar el mantenimiento del código (Microsoft, 2017).

Las capas en las que se dividió la aplicación fueron:

- De datos, en esta capa se almacenaron los scripts, las sentencias y llamados a procedimientos almacenados.
- De conexión, en esta capa se almacenó toda la interacción con la base de datos.
- De modelo y lógica, en esta capa se crearon los modelos que se ocuparon en el proyecto y la lógica del programa.

De presentación, todas las vistas y llamados a las vistas (véase ilustración 7).

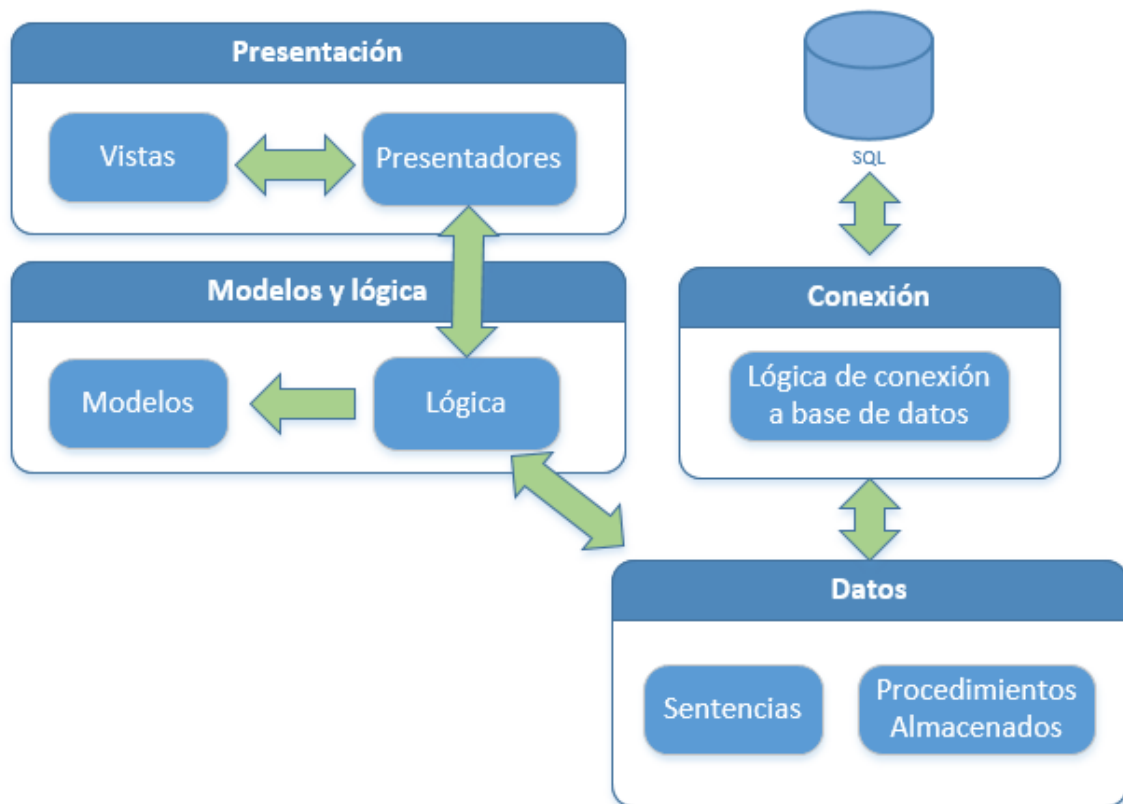


Ilustración 7 Diseño del proyecto contadores

La duración aproximada de este proyecto fue de 6 meses. En este periodo asistí a reuniones donde se mostraron los avances a los usuarios finales quienes hacían observaciones del funcionamiento de la aplicación para modificarla y en otros casos agregar nuevas funcionalidades.

Para realizar una modificación o agregar una nueva funcionalidad, tuve antes que analizar su impacto y decidir su factibilidad.

### **3.3.1.5 Lecciones aprendidas**

Como mencioné este fue mi primer proyecto en ATL, el cual me brindó mucha experiencia, principalmente la convivencia con clientes reales, comprobar la frase “El cliente no sabe lo que quiere”, trabajar con tiempos definidos, llevar un registro de los cambios y avances que han realizado y la experiencia en el lenguaje que me ha servido para otros proyectos.

Otro proyecto que se asignó, en este caso como *outsourcing* por parte de la empresa a otra compañía fue el de lectura de archivos.

## **3.3.2 Interfaz XML**

En este proyecto participé como *outsourcing* en una empresa que requería de un desarrollador.

### **3.3.2.1 ¿Quién participó en el proyecto?**

Líder de desarrollo y Consultor Jr. (yo).

### **3.3.2.2 Roles de los involucrados**

Líder de desarrollo: Desarrollar y asignar las actividades al consultor Jr. Solicitadas por el cliente.

Consultor Jr.: Desarrollar las actividades asignadas.

### **3.3.2.3 Necesidades del cliente**

Contar con un desarrollador que manejara Visual Basic .NET el cual debería de estar en sus instalaciones (La empresa se encuentra en Toluca, Estado de México) para verificar, corregir o crear nuevo software.

### **3.3.2.4 Primeras semanas del proyecto**

En las primeras dos semanas del proyecto el gerente de desarrollo me asignaba determinadas actividades tales como la creación de páginas, consultas a la base de datos, creación de modelos, entre otras. Al concluir las era necesario generar un reporte de mi trabajo día a día; este reporte me lo firmaba el líder del área de desarrollo del cliente y yo los entregaba los fines de semana a ATL. Cada viernes el gerente de desarrollo de ATL iba a la empresa del cliente a verificar el avance, entregar el software adicional que

él también desarrollaba y a obtener retroalimentación sobre mi participación en la empresa. Gracias a mi desempeño solo fui supervisado por 2 semanas, posteriormente el gerente de desarrollo me asignó las tareas que él realizaba y sólo me preguntaba periódicamente el estatus del proyecto.

### **3.3.2.5 Mi participación en el proyecto**

La tarea que se me asignó al inicio fue comprobar que un servicio de Windows realizaría de forma correcta la lectura de tres tipos de archivos XML procedentes de diferentes rutas de un servidor, dependiendo su contenido, estos archivos se guardan en tablas específicas de dos bases de datos SQL server 2008 e INFORMIX donde los datos cargados son procesados por otras aplicaciones y finalmente son alojados en una carpeta de respaldo.

### **3.3.2.6 Situaciones**

En este caso el programa ya estaba creado y había probado, sólo era necesario validar que trabajara de forma correcta. La situación más importante que se presentó fue que al momento de realizar las validaciones el programa sólo fue capaz de leer un XML de los tres tipos que debería leer. Adicionalmente, se detectaron otras situaciones más:

- No existía documentación del programa
- El programa estaba escrito en un solo archivo de más de 2000 líneas de código.
- Los comentarios en el programa no aportaban gran información.
- Los requerimientos no se habían actualizado con los últimos cambios pedidos.
- Tiempo de liberación a producción en contra.
- La persona que desarrolló el programa, por algún motivo, ya no laboraba en la empresa.

### **3.3.2.7 Solución temporal**

Como el proyecto debía ser liberado en un par de semanas, no se contaba con el tiempo suficiente para reorganizar el programa, hacerlo modular y así facilitar su mantenimiento; por lo anterior, para cumplir con el objetivo que me fue encomendado, tomé la decisión de depurar el programa (hacer el seguimiento línea a línea en tiempo de ejecución de las acciones que ejecutaba) y verificando en un servidor de pruebas, para poder entender el comportamiento del servicio al momento de leer cada tipo de archivo, poder localizar los errores lógicos o la falta de implementación del código y forzar a que realiza las acciones necesarias para la continuidad del negocio. Una vez que se logró la lectura de los respectivos archivos, se liberó el programa en producción.

Esta solución fue temporal ya que sería necesario leer nuevos datos de los archivos XML, lo cual, por la forma en que estaba estructurado el programa se complicaría en gran

medida agregar nuevas características y funcionalidades, sin que se modificaran los demás procesos de lectura y se inyectaran errores.

### 3.3.2.8 Solución definitiva

Los requerimientos seguían incrementando y así la complejidad de mantener el programa funcionando. Previa consulta con el líder de desarrollo de la empresa, decidí rediseñar el programa y así para facilitar su mantenimiento; fue necesario convencer al líder de desarrollo de la empresa haciéndole ver que con un nuevo diseño se tendrían ventajas tales como menor tiempo en mantenimiento, optimización del tiempo de proceso de cada archivo, bitácoras más claras. Logré la respectiva autorización con la condición de que no se afectara la continuidad del negocio. Decidí crear un plan el cual me ayudaría a crear e implementar el nuevo servicio de forma exitosa.

El plan consistió en el cálculo de los tiempos del nuevo diseño y la codificación, incluyendo cómo mantener el servicio que estaba en ejecución con los nuevos requerimientos para asegurar la continuidad del negocio, las pruebas y la implementación.

Diseñé el nuevo programa con el paradigma de POO para así ocupar las ventajas y nuevas características que tiene Visual Basic .Net. El patrón de diseño que elegí fue separar el modelo, la lógica del negocio, los datos y las conexiones a las bases de datos para así facilitar su mantenimiento del programa (véase ilustración 8).

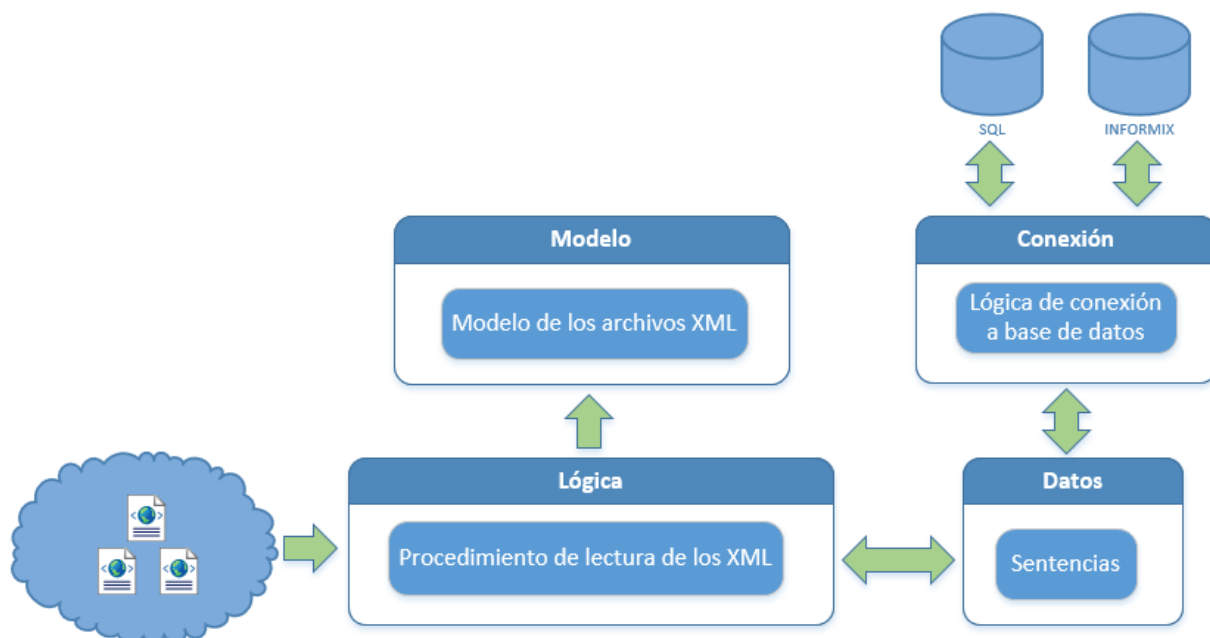


Ilustración 8 Diseño interfaz XML

Después de diseñar y codificar el nuevo servicio, fue necesario realizar un gran número de pruebas para asegurarme que el nuevo servicio estuviera trabajando de forma correcta, al concluir las pruebas pedí la autorización para liberar la nueva versión, la cual ha trabajado sin contratiempos.

### **3.3.2.9 Mantenimiento**

Con el paso del tiempo el servicio se ha ido manteniendo con los nuevos requerimientos y gracias al patrón elegido permite agregar nuevas características, y no es necesario que sea yo quien se las agregue, otros desarrolladores lo han podido mantener sin complicaciones.

### **3.3.3 Proyecto de Servicios Web (*Web Services*) para ver el desempeño de vendedores**

Este proyecto también lo realicé como *outsourcing*.

#### **3.3.3.1 ¿Quién participó en el proyecto?**

Consultor Jr. (yo).

#### **3.3.3.2 Roles de los involucrados**

Consultor Jr.: Desarrollar el servicio Web

#### **3.3.3.3 Necesidad del cliente**

El cliente compró una aplicación que estaría en la nube, la cual necesitaba obtener los datos alojados en una base de datos en INFORMIX; esta base de datos cuenta con más de 2 mil tablas alguna de ellas con más de veinte millones de registros. Era necesario crear un servicio Web por cada tabla solicitada para enviar la información.

#### **3.3.3.4 Análisis de la situación**

El lenguaje con el que trabaja la empresa es Visual Basic .Net por lo tanto decidí crear una aplicación donde estarían los servicios Web requeridos para su consumo.

#### **3.3.3.5 Desarrollo**

La aplicación donde estaban los servicios Web la estructuré de la siguiente manera: modelo, conexión a la base de datos, sentencias y los servicios Web. La estructure así para un fácil mantenimiento y mayor seguridad.

Debido a la gran cantidad de registros que devolvían algunas tablas, agregué filtros a los servicios Web y así reducir la cantidad de registros devueltos.



### 3.3.3.6 Primeras pruebas

Los encargados de desarrollar la aplicación alojada en la nube realizaron las primeras pruebas para verificar la conectividad con el servidor y asegurarse que se pudieran obtener los datos sin problemas. Estas pruebas se realizaron con un software llamado *SoapUI* el cual fue creado específicamente para probar los servicios Web. En la realización de las pruebas no hubo inconvenientes.

### 3.3.3.7 Pruebas con la aplicación en la nube

Posteriormente, se prosiguió a realizar las pruebas con la aplicación alojada en la nube. En éstas comenzaron a salir algunos inconvenientes de *timeout* (la aplicación sólo tenía 120 segundos para recibir una respuesta) y duplicidad de datos (véase ilustración 9).

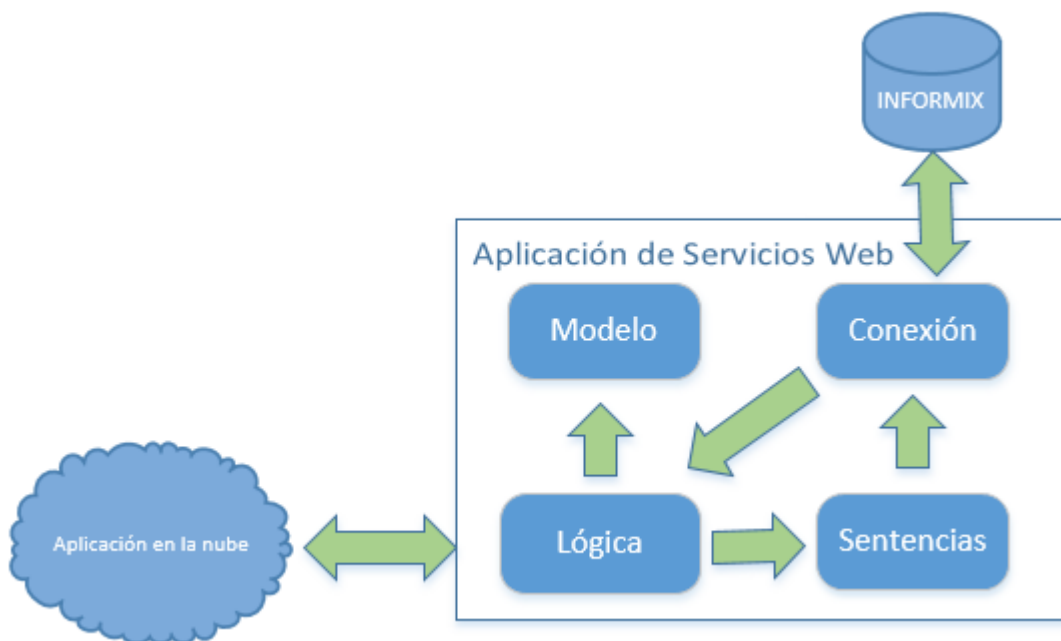


Ilustración 9 Diseño servicios web primeras pruebas

### 3.3.3.8 Cambio en los requerimientos

Debido a esas complicaciones los encargados de la aplicación alojada en la nube decidieron modificar los requerimientos iniciales; los cambios principales fueron:

- Ya no querían un servicio Web por cada tabla de la base de datos, sino un servicio Web que enviara los datos combinando las tablas. Por ejemplo, en la base de datos existen dos tablas una con el id del cliente, nombre del cliente y otra tabla donde se encuentra el id del cliente y las direcciones que éste posee, ya no

requerían dos servicios Web para cada una de las tablas, sólo requerían uno con el id del cliente, nombre y dirección.

- Sólo requerían que los datos se enviaran una vez; por ejemplo, si ellos realizaban una consulta al servicio Web y les llegaban originalmente 100 registros, en la siguiente consulta sólo les tenían que llegar o los nuevos registros o, si algún registro de los originalmente enviados que hubiera sido modificado.
- Minimizar el tiempo de respuesta en el servicio Web; sin importar el número de registros, era necesario que no tardara más de 2 min.
- Los servicios Web ya no requerirían filtros, como el tiempo de respuesta se reduciría éstos ya no serían necesarios.
- Si la consulta inicial arrojaba más de 10000 registros era necesario realizar una primera carga a partir de un archivo .csv.

### **3.3.3.9 Desarrollando nuevos requerimientos**

Comencé a desarrollar los nuevos requerimientos, lo primero que realicé fue verificar cuáles serían las tablas que estarían involucradas para generar los reportes requeridos; a partir de esto, creé las sentencias y después comparé los resultados obtenidos con los reportes oficiales de la empresa (no provienen de la misma base de datos) para asegurar que las sentencias estuvieran correctas.

Para controlar el envío de los registros decidí ocupar tablas intermedias para llevar el control de los datos enviados y verificar si alguno tenía algún cambio, estas tablas las creé en SQL server ya que las restricciones de la empresa prohíben crear tablas en la base de datos de INFORMIX.

Para reducir el tiempo de consulta en los registros creé una tarea en Windows que se ejecutara cada 10 min y fuera actualizando la tabla en SQL para que el servicio Web consultara la tabla de SQL y ésta fuera rápida. Esta implementación también funcionó para quitar los filtros (véase ilustración 10).

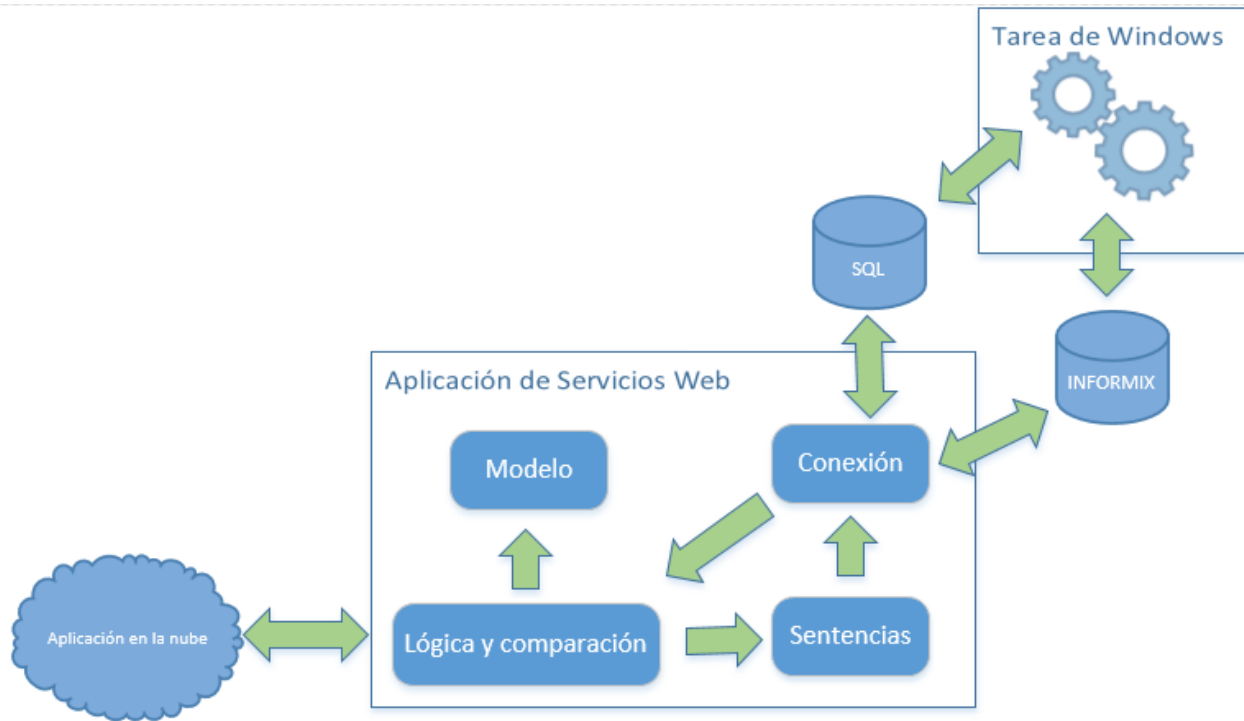


Ilustración 10 Diseño servicios web liberación final

Para los reportes que generaban más de 10,000 registros ocupé *Power Pivot* de Excel para facilitar la creación del archivo csv.

### 3.3.3.10 Liberación

Se volvieron a realizar las pruebas y se pusieron en marcha los servicios Web para su consumo, la aplicación consumió los datos de forma exitosa. Conforme ha sido necesario se han ido agregando nuevos reportes sin ningún problema, esta aplicación sigue en crecimiento.

# Conclusiones

En este informe he plasmado mi evolución dentro de ATL: comencé a laborar como becario y gracias a mi desempeño se me han incrementado las responsabilidades hasta las que corresponden a un consultor sr.

He detallado esta evolución en los proyectos descritos:

- En el proyecto de mantenimiento y soporte me enfrenté, más que a un reto intelectual, un reto de resistencia física y de disciplina, ya que consistió en seguir las instrucciones que se me indicaron de forma ordenada para lograr el armado de los equipos.
- El proyecto de capacitación representó ya un reto intelectual y de organización debido a que fue necesario dominar los temas que debí impartir y coordinar tiempos con los otros instructores; esto lo logré aplicando, además de mi formación profesional, habilidades tales como el trabajo en equipo y la comunicación asertiva.
- Definitivamente los proyectos de desarrollo han sido lo que me implican los retos más destacados, ya que debo crear soluciones tecnológicas y aplicar habilidades profesionales para trabajar en equipo con los demás miembros de la empresa, interactuar con los clientes y sobre todo cumplir con las expectativas de los directivos han puesto en mí.

Gracias a la capacitación, supervisión y trabajo en equipo, en los casi dos años de laborar en ATL, he alcanzado la madurez necesaria para obtener un ascenso a consultor sr y así seguir en mi desarrollo profesional.

# Bibliografía

*Agile Academy (2015). Personal Software Process Workbook Student*

*Agile Academy (2015). Scrum Master Certified Workbook Student. SCRUMstudy SMC.*

Claudio V. (07 junio, 2016). Lenguajes de programación más usados, más demandados y con más futuro. Recuperado de <https://www.adslzone.net/2016/06/07/lenguajes-programacion-mas-usados-mas-demandados-mas-futuro/>

Claudio V. (07 junio, 2016). Lenguajes de programación más usados, más demandados y con más futuro. Recuperado de <https://www.adslzone.net/2016/06/07/lenguajes-programacion-mas-usados-mas-demandados-mas-futuro/>

Durán, F., Gutiérrez, F. y Pimentel E. (2007) Programación Orientada a Objetos con Java. España: Thomson.

*Microsoft (2017). C#: Developer Network.* Estados Unidos. Recuperado de <https://msdn.microsoft.com/es-mx/library/kx37x362.aspx>

*Microsoft (2017). The Model-View-Presenter (MVP) Pattern.* Recuperado de <https://msdn.microsoft.com/en-us/library/ff649571.aspx>

*Microsoft (2017). SQL Server 2016.* Recuperado de <https://www.microsoft.com/es-es/sql-server/sql-server-2016>

*Oracle (2017). Conozca más sobre la tecnología Java.* Recuperado de <https://www.java.com/es/about/>

*Oracle (2017). Data Base 12c Oracle.* Recuperado de <https://www.oracle.com/lad/database/index.html>

Rumbaugh, J., Jacobson, I. y Booch, G. (2000). El lenguaje unificado de modelado manual de referencia. España: Pearson Educación S.A.