PASANTÍA

“DESARROLLO DE UN WEB SERVICE PARA PARA LA EXTRACCIÓN DE INFORMACIÓN DE USUARIOS DEL PROYECTO PILOTO DE SERVICIO DE AGUA PREPAGO DE LA EAAV.”

JHON MARIO OCHOA LEMUS
CÓDIGO 160003131

KAREM DANIELA VELASQUEZ PARDO
CÓDIGO 160003142

UNIVERSIDAD DE LOS LLANOS
FACULTAD DE CIENCIAS BASICAS E INGENIERIA
ESCUELA DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS
VILLAVICENCIO, COLOMBIA
2017

PASANTÍA

“DESARROLLO DE UN WEB SERVICE PARA PARA LA EXTRACCIÓN DE INFORMACIÓN DE USUARIOS DEL PROYECTO PILOTO DE SERVICIO DE AGUA PREPAGO DE LA EAAV.”

JHON MARIO OCHOA LEMUS
CÓDIGO 160003131

KAREM DANIELA VELASQUEZ PARDO
CÓDIGO 160003142

Codirector Académico:
ING. JORGE ANTONIO HERNANDEZ SANCHEZ

UNIVERSIDAD DE LOS LLANOS
FACULTAD DE CIENCIAS BÁSICAS E INGENIERÍA
ESCUELA DE INGENIERÍA
PROGRAMA INGENIERÍA DE SISTEMAS
VILLAVICENCIO
2017

NOTAS DE ACEPTACION

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

_____
Ing. Jorge Antonio Sánchez, Esp.

_____
Ing. Javier Mancera, Jurado.

_____
Fecha.

DEDICATORIAS

Dedicamos este proyecto principalmente a Dios y a nuestros familiares quienes en cada paso que dimos estuvieron presentes para apoyarnos y hacer posible la finalización de nuestra carrera.

AGRADECIMIENTOS

Agradecemos primeramente a Dios, a nuestros padres y familiares, quienes con su apoyo incondicional han hecho posible este logro en nuestras vidas, a nuestros compañeros y profesores de la facultad de ciencias básicas e ingeniería de la universidad de los Llanos puesto que con su compañerismo y experiencia forjamos las bases de nuestro futuro.

TABLA DE CONTENIDO

1. Resumen 7
2. Introducción 8
3. Marco Teórico 9
   3.1. Web Service 9
   3.2. Ventajas de los Web Service frente a otras tecnologías..... 9
   3.3. Organización que desarrollan los estándares del Web Service..... 10
   3.4. Plataformas Web 10
   3.5. Tecnologias y entornos de ejecución 11
      3.5.1. Tecnologías utilizadas 11
         3.5.1.2. Framework de desarrollo web 12
            3.5.1.2.1. Ventajas de framework de desarrollo web .. 13
      3.5.2. Entornos de ejecución 14
   3.6. Extraccion de datos 14
      3.6.1. Extracción de datos del servidor para el cliente 14
      3.6.1.2. Seguimiento 15
4. Metodologia Scrum 15
   4.1. Equipo Scrum 16
      4.1.1. Product Owner 16
      4.1.2. Equipo de desarrollo 16
      4.1.3. The Scrum Master 16
   4.2. Elementos de la metodología Scrum 16
   4.3. Spring Planning 18
   4.4. Scrum Diario 18
   4.5. Sprint Review 18
   4.6. Retrospectiva del sprint 19
5. Desarrollo del proyecto 19
   5.1. Cronograma de actividades 19
      5.1.1. Etapa de inicio 20
      5.1.2. Etapa de planeación 20
      5.1.3. Etapa de desarrollo 20
      5.1.4. Etapa de cierre 21
   5.2. Ejecución 21
   5.3. Sprint 1 21
   5.4. Sprint 2 24
      5.4.1. Planificación del Sprint 24
      5.4.2. Desarrollo del Sprint 27
         5.4.2.1. Adaptación y construcción de información necesaria para el proyecto piloto 27
      5.4.3. Revision del Sprint 27
      5.4.4. Retrospectiva del Sprint 28
   5.5. Sprint 3 28

5.5.1. Planificación del Sprint ……..... 28
5.5.2. Desarrollo del Sprint …… 30
5.5.3. Planeación y desarrollo de módulo “Extracción y envió de información” ……... 30
5.5.4. Revisión del Sprint …… 30
5.5.5. Retrospectiva del Sprint …….. 31
5.6. Sprint 4 …… 31
5.6.1. Planificación del Sprint …… 31
5.6.2. Desarrollo del Sprint …… 32
5.6.2.1. Modelo Relacional …… 32
5.6.2.2. Desarrollo del módulo de envío y recepción de datos en la plataforma …….. 33
5.6.3. Revisión del Sprint …… 33
5.6.4. Retrospectiva del Sprint ……... 33
6. Resultados Obtenidos …… 34
7. Conclusiones ……... 35
8. Referencias …….. 36

1. RESUMEN

Como es sabido los servicios públicos hacen parte fundamental de la vida diaria, dado que son esenciales para subsistir. La carencia de alguno de ellos, desestabilizaría parcial o totalmente una zona urbana y/o rural.

La empresa de acueducto y alcantarillado de Villavicencio (EAAV) cuenta con un sistema postpago para los servicios que ofrece, uno de dichos servicios es el sistema de acueducto que abastece gran parte del casco urbano y rural de la ciudad de Villavicencio[1], siendo cientos de personas las que hacen uso de este líquido vital, que además no cuentan con un control sobre la cantidad de su consumo mensual, esto lleva a un desperdicio en muchas ocasiones; Aparte de ser un agravante para la situación ambiental, también influye en el bolsillo de los consumidores quienes ven reflejado esto en su factura mensual.

En algunas zonas de Villavicencio existen usuarios que no cuentan con una situación económica estable, y en muchos de estos casos van acumulando deudas con la empresa prestadora del servicio, contribuyendo de esta forma al incremento de las cifras correspondientes a deudas para dicha entidad. A pesar de las múltiples estrategias que se han implementado con el fin de dar flexibilidad a los usuarios [2] para amortiguar sus deudas, no se han presentado resultados notables, por este motivo se hace necesario generar una nueva estrategia que permita recuperar parte de este dinero, esta consiste en un plan de medición prepago, el cual permite que el usuario realice recargas del monto que desee y de acuerdo a este dinero se le destina la cantidad de agua en metros cúbicos correspondiente, teniendo su servicio de acuerdo a su capacidad de pago y aportando a disminuir la deuda que tiene con la empresa. Lo anterior se lleva a cabo con ayuda de un equipo de cuatro estudiantes de Ingeniería de sistemas de la Universidad de los llanos, quienes realizarán todo lo concerniente a codificación de dicho proyecto. Este se divide en dos partes:

- Plataforma de integración de todos los procesos correspondientes.
- Web service que provee la información necesaria para la alimentación de la plataforma y la creación de la base de datos propia del sistema de medición prepago.

2. INTRODUCCIÓN

Este proyecto se realizó en la Empresa de Acueducto y Alcantarillado de Villavicencio, consiste en la creación de un web service para el intercambio de información entre la plataforma de recargas y la base de datos empresarial, incluyendo las consultas realizadas a la misma para la extracción de información concerniente a los usuarios, sus deudas y las tarifas aplicadas. Además de la creación de la base de datos del sistema de medición prepago.

Dicho proceso se llevó a cabo usando como base la metodología Scrum, se llevaron a cabo reuniones semanales y mensuales, teniendo entregas periódicas a nivel grupal y general, y teniendo en cuenta cada Sprint.

Luego de tener definido el procedimiento a seguir, se procedió a generar un cronograma basado en los Sprint definidos por el Product Owner dependiendo de las expectativas del cliente, dando paso a la etapa de desarrollo.

Gracias a la unión de este proyecto con la plataforma web, la empresa podrá poner en marcha la nueva estrategia de disminución de cartera, con plena confianza de tener un software seguro, rápido y eficaz, siempre y cuando la empresa tenga la información y los implementos necesarios en funcionamiento y al día, ya que esta plataforma depende de ellos para su óptimo funcionamiento.

3. MARCO TEÓRICO

En la actualidad buscando mejorar el control en los servicios públicos como lo son el agua y la energía, se han venido implementando sistemas con nuevas tecnologías, los cuales permiten a los usuarios llevar un control sobre el cupo que deseen adquirir y consumir, esto conlleva a cambios significativos en la medición y facturación de su servicio[3]. Esto hace necesario adaptarse y hacer uso de las nuevas tecnologías que se encuentran hoy en día, tales como plataformas web y/o móviles, contadores prepagos y lo que estos requieren para su manejo, siendo necesario complementar con otras tecnologías para un mejor y óptimo funcionamiento.
A continuación, se muestra un repaso de los conceptos básicos de los temas que están inmersos en el desarrollo del proyecto y que permitirán su desarrollo como lo son los web service, plataformas web, algoritmos de extracción de datos, entre otros.

3.1. Web Service

Cuando algunos sistemas de información, requieren integrarse con otras aplicaciones para el tratamiento, requerimiento y/o utilización de la información, en donde estas aplicaciones están basadas en otros lenguajes y plataformas diferentes, estos paradigmas son los web service [4]. Permitiendo a diversas aplicaciones interactuar para sacar mayor y el mejor provecho entre ellas.

“Según numerosos estudios en el área de T.I, el número de sistemas de software diferentes, que utilizan las tecnologías web, como los servicios web, están en constante crecimiento.”[5]
Permite una infinidad de posibilidades, para el desarrollo de distintas soluciones a problemáticas reales presentadas en ambientes empresariales, institucionales y educativos, permitiendo nuevas y mejores formas de aprendizaje [6].

3.2. Ventajas de los Web Service frente a otras tecnologías.

- Aportan interoperabilidad entre aplicaciones de software, independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Independencia del lenguaje de programación: El servidor y el cliente no necesitan estar escritos en el mismo lenguaje.

- Independencia del modo de transporte: SOAP puede funcionar sobre múltiples protocolos de transporte, como, por ejemplo: http, HTPPS, BEEP, llOP, SMTO o FTP.

3.3. Organizaciones que desarrollan los estándares de los Web Service.

Como toda tecnología, tiene reglas y estándares definidos para el modelo, y referencias de su uso. Tales organizaciones las cuales definen dichos estándares son:

- **W3C**: World Wide Web Consortium. Es la principal organización desarrolladora de estándares para Web Service.
- **OASIS**: Organization of the Advancement Structured information Standards, Avances de estándares de la información estructurada, estos estándares trabajan con tecnologías XML.
- **IETF**: Internet Engineering Task Force. Trabaja sobre las arquitecturas y tecnologías que corren y son ejecutadas por internet.
- **ISO**: International Organization for Standardization. Es la encargada de desarrollar los estándares para mejoramiento del comercio internacional

3.4. Plataformas Web.

Una plataforma web es más que una página web. Una plataforma incluye elementos adicionales a la página web tales como aplicaciones, carritos de compras, formularios, convertidores, instancias de aprobación y alguna otra solución específica para la necesidad del cliente y el mercado. Estos elementos pueden ser públicos o privados, tales como sistemas de comunicación interna o inventarios.

3.5. Tecnologías y entornos de ejecución.

3.5.1. Tecnologías utilizadas.

HTML: Lenguaje de marcado diseñado e implementado por SirTim Berners Lee, a finales de 1989, y presentado a la comunidad en 1990 sobre un ordenador NeXT. Basado en el SGML e HyperText, creando un tipo de HyperText global, utilizando la nomenclatura DNS y con su propio protocolo para la recuperación de archivos mediante enlaces de HyperText (HTTP)[7].

CSS: Hojas de estilo en cascada. Mecanismo para añadir apariencia gráfica a los documentos Web (HTML). Aunque ya Sir Tim BernersLee en 1990 separó la estructura del documento de su representación gráfica (mediante una hoja de estilos muy simple), no fue presentado a la comunidad hasta abril de 1995 en la conferencia WWW por Håkon Wium Lie y Bert Bos[8].

PHP: Lenguaje de scripting ejecutado en el servidor web, es el sucesor de un producto llamado PHP/FI. Creado en 1994 por Rasmus Lerdorf, la primera versión fue un conjunto simple de archivos binarios CGI escritos en C. Con el tiempo llamó al conjunto de scripts “Personal Home Page Tools”, más conocido como “PHP Tools”. En julio de 2004 fue lanzado PHP 5, con un núcleo Zend Engine 2.0 que contiene un nuevo modelo de objetos y docenas de nuevas opciones [9].

SQL: En la década de los setenta, después de que E.F. Codd presentará su propuesta sobre el Modelo Relacional, aparecieron varios productos comerciales que seguían esta propuesta [...]. En la década de los 80 el organismo ANSI publicó la primera versión estándar del SQL (Structured Query Language). A partir de ese momento, este organismo publicó varias versiones que mejoran las deficiencias encontradas en las versiones previas e introducían las mejoras que las investigaciones en el campo de las bases de datos iban descubriendo[10].

JavaScript: Lenguaje de programación interpretado, del que surgió el estándar ECMAScript, se utiliza, usualmente, embebido en documentos HTML o referenciado como archivos .js dentro del código HTML y ejecutado, en un alto porcentaje de los casos, en el navegador del usuario, no en el servidor. Sirve para añadir interacción con el usuario a los documentos estáticos HTML. Fue creado en 10 días en mayo de 1995 por Brendan Eich, trabajador de Netscape, actualmente en Mozilla. Entre 1996 y 1997 fue llevado ante la European Computer Manufacturers Association (ECMA) (Ecma International) del que surgió el estándar ECMAScript, que, actualmente va por la edición 5.1[11].

JQuery: Es una biblioteca que añade funcionalidad a JavaScript, creada por John Resig, permite interactuar con los documentos HTML de una manera sencilla: manipular el árbol DOM del HTML, manejar eventos y agregar interacción AJAX[12].

3.5.1.2. Framework de desarrollo web.

Un framework de desarrollo se puede considerar como una aplicación genérica incompleta y configurable a la que se le puede añadir las últimas piezas para construir una aplicación concreta[13].

Un framework agrega funcionalidad extendida a un lenguaje de programación, automatiza muchos de los patrones de programación para orientarlos a un determinado propósito, proporcionando una estructura al código, mejorándolo y haciéndolo más entendible y sostenible, y permite separar en capas la aplicación[14].

Actualmente, con el auge de las tecnologías, han surgido una gran cantidad de grandes framework que lograron impactar hasta el punto de ser reconocidos gracias al uso efectivo de sus diferentes herramientas que poseen, tales como *Angular.js, ionic, Ruby on Rails, Django* entre otros. Algunas características de estos son:

- **Abstracción de URLs y sesiones**: No es necesario manipular directamente las URLs ni las sesiones, el framework ya se encarga de hacerlo.

- **Acceso a datos**: Incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en BD, XML, etc. Además de ello se encargará de generar la infraestructura esquemática de formularios, campos, relaciones, y operaciones CRUD, a partir del modelo de base de datos de un sistema previamente contemplado.

- **Controladores**: implementa una serie de controladores para gestionar eventos como el acceso a una página. Estos controladores en los framework ofrece la propiedad de ser fácilmente adaptables a las distintas necesidades de un proyecto concreto.

- **Autentificación y control**: Incluyen mecanismos para la identificación de usuarios, mediante el uso de login que permiten el acceso y restricción en determinadas páginas creadas.

- **Separación entre diseño y contenido**: Esta característica es una de las principales ventajas que puede ofrecer un framework, dado que permite separar de todo el diseño, visualización con toda la lógica del negocio, lógica de aplicación y demás elementos pertinentes tanto del front-end como del back-end.

3.5.1.2.1. Ventajas de framework de desarrollo web:

A la hora de desarrollar aplicaciones web, con base en un determinado framework de desarrollo se nos presenta una serie de ventajas al facilitarnos la creación de aplicaciones, el mantenimiento del código, realizar migraciones entre otras. Principales ventajas:

- **Uso de patrones de diseño:** una de las razones principales por las cuales se pueden usar herramientas de desarrollo como esta, es la facilidad con la que se pueden implementar patrones de diseño para el desarrollo de una aplicación. El patrón más utilizado es el conocido Modelo-Vista-Controlador (MVC). Aunque este puede ser usado sin la necesidad de usar un framework:

  Modelo: Representa los datos de la aplicación y sus reglas de negocio.

  Vista: Representa la capa presentación, como representamos los datos a los usuarios.

  Controlador: Es el encargado de procesar las peticiones de los usuarios y controla el flujo de ejecución del sistema.

Fig. 1

[15]

- **Estructura predefinida:** la distribución de la aplicación ya no queda a cargo de los desarrolladores pues esta viene predefinida en el framework. Esto permite ubicar rápidamente archivos, código y demás. Puesto que estos tienen su ubicación fija.

- **Código altamente testeado:** muchos de los framework están en constante desarrollo y tienen soporte en sus respectivas versiones. Por ende las diferentes entregas garantizan un buen funcionamiento.

- **Comunidad:** La gran mayoría de los frameworks tienen detrás a una amplia comunidad de usuarios, de los cuales muchos ayudan en su desarrollo o creando extensiones con funcionalidades extra que se puede utilizar de forma sencilla sin tener que ser desarrolladas por cuenta propia.

3.5.2. Entornos de ejecución.

Sistema Operativo: Windows, Linux y MacOS.

Sistema Gestor de Base de Datos: Mysql, PostgreSQL, SQL Server y Oracle.

Lenguaje de Desarrollo en Servidor: PHP, Python, ASP, Perl, Ruby.

Servidor Web: Apache, Microsoft iis, Sun java, Ngnix entre otros.

3.6. Extracción de datos.

3.6.1. Extracción de datos del servidor para el cliente.

El acceso a datos remotos se inicia a menudo extrayendo datos de un servidor SQL Server, MySql server u otro alguno el cual proporcione ese tipo de servicio, después se insertan en una base de datos cliente. Relativamente se puede ver como un proceso fácil, sencillo y dinámico, pero todo dependerá del alcance que se decida, en el producto de software y/o solución dispuesta para una problemática.

Para extraer los datos de parte del servidor, se debe configurar mediante la programación de la aplicación. Cuando se configure e inicie una operación de extracción con cualquiera técnica, tendrá que especificar lo siguiente:

- La base de datos de la que desea extraer los datos.
- La autenticación del servidor web necesario para conectarse a un servidor web.
- La autenticación de SQL utilizada.
- Las tablas que se copiaran y propagaron al cliente.
- Las opciones de extracción, incluidos el seguimiento y una tabla de errores.

3.6.1.2. Seguimiento.

Una de las opciones las cuales puede ofrecer una aplicación, y quizás una de las más fundamentales, es la realización de un seguimiento de todos los cambios realizados en una tabla de la base de datos. Las operaciones de extracción e inserción cuyo seguimiento se podrá realizar mediante el acceso a datos remotos, utilizan el control de simultaneidad optimista.

4. METODOLOGÍA SCRUM

Teniendo en cuenta que la ingeniería de software y de sistemas ha evolucionado, se ha generado siempre la búsqueda de metodologías adecuadas para la buena producción de software y de alta calidad. Cuando se habla de metodología, se refiere a cómo va ser la planeación de un proyecto, el orden de las actividades, etapas y el cómo va a ser su transición. Es ahí donde se va a escoger el que se ajusta al plan de desarrollo del proyecto.

Con el fin de cumplir con todos los objetivos definidos se eligió la metodología Scrum, que es la base sobre la cual se desarrollará el proyecto ya que describe los procesos que involucran la administración y desarrollo del mismo.

“Scrum no es un proceso o una técnica para construir productos; Más bien, es un Marco en el que puede emplear diversos procesos y técnicas. Scrum deja claro la eficacia relativa de sus prácticas de gestión y desarrollo de productos para que pueda mejorar… las reglas de Scrum enlazan los eventos, roles y artefactos, que rigen las relaciones e Interacción entre ellos” [16].

Scrum es un proceso empírico el cual se base en el resultado previo de las experiencias. Los tres pilares fundamentales de este proceso empírico son:

- **Transparencia:** aquellos aspectos importantes y significativos del proyecto y/o proceso deben ser conocidos por todos los participantes. Esto exige que estos aspectos se definan a través de una norma en común.
- **Inspección:** las inspecciones son fundamentales en el proceso, por lo cual se debe realizar con frecuencia para la revisión de los distintos artefactos para las metas de cada sprint, para detectar errores o posibles fallas. Las inspecciones son más beneficiosas si son realizadas diligentemente, por inspectores.
- **Adaptación:** la detección de aspectos los cuales se están desviando o están fuera del alcance del proceso y/o proyecto, por lo cual deben ser ajustados para evitar que esa desviación crezca.

4.1. EL EQUIPO SCRUM

Los equipos Scrum están conformados por el Product Owner, el equipo de desarrollo y un Scrum Master. “Los Equipos Scrum son auto-organizadores y multifuncionales. Los equipos auto-organizados eligen la mejor forma de realizar su trabajo, en lugar de ser dirigido por otros fuera del equipo” [16].
Los Equipos Scrum entregan los productos de forma iterativa e incremental, maximizando la realimentación. Las entregas incrementales de productos "Done" garantizan una versión potencialmente del producto.

4.1.1. Product Owner

El dueño del producto, es el máximo responsable de todo el proceso es quien el que va poniendo el valor al producto, es la persona más involucrada en el proyecto, la cual va decidiendo los cambios, y teniendo una perspectiva sobre todo los requerimientos ajustados a los avances previos los cuales se le presentan, al dueño del producto. Las decisiones del dueño del producto son fundamentales y se deben respetar.

4.1.2. Equipo de Desarrollo

El equipo de trabajo está conformado, por profesionales los cuales realizan entregables incrementales periódicos, los cuales son funcionales terminados. Estos son presentados en periodos definidos de tiempo, y son dados a ver al product owner.

4.1.3. The Scrum Master

*“*Es el encargado de comprobar que el modelo y la metodología funcionan. Eliminará todos los inconvenientes que hagan que el proceso no fluya e interactúa con el cliente y con los gestores*.”*[17].
Es considerado como el rol más importante, ya que, acata como intermediario entre el dueño del producto y el equipo de trabajo, filtrando todas las actividades pertinentes al alcance de cada entregable y reorganizando los requerimientos adicionales, evitando de esta manera la desorientación del equipo de desarrollo.

4.2. Elementos de la metodología Scrum.

Scrum, propone tres herramientas o "artefactos" para mantener organizados nuestros proyectos. Estos artefactos, ayudan a planificar y revisar cada uno de los Sprints.

- **Product Backlog:** lista de necesidades del cliente. Estos requisitos serán los que tendrá el producto o los que adquirirá.
  La lista será gestionada y creada por el cliente con la ayuda del Scrum Master, quien indicará el coste estimado para completar un requisito, y además contendrá todo lo que aporte un valor final al producto.

Figura 2. Formato de Product Backlog.

Identificador (ID) de la historia	Enunciado de la historia	Alias	Estado	Dimensión / Esfuerzo	Iteración (Sprint)	Prioridad

Fuente: [18]

- **Historias de Usuario**:

Son las descripciones de las funcionalidades que va a tener el software. Estas historias de usuario, serán el resultado de la colaboración entre el cliente y el equipo, e irán evolucionando durante toda la vida del proyecto.

Figura 3. Formato de las Historias de Usuario.

Título:			**ID:**	
Funcionario:			**Dependencia:**	Historia Clínica
Tipo:	Épico	Historia	**No. Requisito:**	
Descripción:				
Criterios de aceptación:				

Fuente: Propia

Título: Nombre de la historia de usuario
ID: Identificación obtenida en el Product Backlog
Funcionario: Cargo de la persona entrevistada
Descripción: Descripción de la historia
No. Requisito:
Criterios de aceptación:
Dependencia: Dependencia a la que pertenece la persona entrevistada
Tipo: El tipo de historia (Épica o de usuario)

- **Sprint Backlog:** lista de tareas que hace un Sprint.La realiza el equipo durante la reunión de planificación del sprint, auto asignando cada tarea a un miembro del equipo, e indicando en la misma lista cuánto tiempo o esfuerzo prevé.

Historia de Usuario	Tarea	Responsable	Estado	Días / Estimación (Hras)									
				Total	1	2	3	4	5	6	7	8	9	10

Figura 4. Formato de las Sprint Backlog.
Fuente: [19]

Historia de Usuario: Historia de usuario del Product Backlog.
Tarea: Tarea del Product Backlog.
Responsable: Persona responsable a realizar la tarea.
Estado: Estado de la historia de usuario (Pendiente, En proceso, Completa)
Días / Estimación (Hras): Lista de días que tiene el Sprint con su respectivo campo que representa el tiempo gastado en horas.

- **Incremento:** El incremento es la parte de producto producida en un sprint, y tiene como característica el estar completamente terminada y operativa, en condiciones de ser entregada al cliente.

4.3. Sprint Planning

El trabajo que se llevará a cabo en el Sprint está previsto en la planificación de Sprint. Este plan es creado por el trabajo colaborativo de todo el Equipo Scrum. Esta planeación es clave pues responderá duda sobre que se puede entregar al final del Sprint, como se logra y qué tareas se incluirán para la realización de cada Sprint.

4.4. Scrum Diario.

Reunión diaria breve, de no más de 15 minutos, en la que cada miembro del equipo dice las tareas en las que está trabajando, si se ha encontrado o prevé encontrarse con algún impedimento.

4.5. Sprint Review.

Es una revisión al Sprint en una fecha establecida para inspeccionar el incremento del producto, y si es necesario adaptar el backlog del producto. En esta reunión el equipo Scrum y las partes interesadas interactúan sobre lo que se hizo en el sprint. Lo que caracteriza a esta reunión es que no tiene ningún status, es más informal y su principal función es retroalimentar y fomentar la colaboración.

4.6. Retrospectiva del sprint.

Reunión que se realiza tras la revisión de cada sprint, y antes de la reunión de planificación del siguiente, con una duración recomendada de una a tres horas, según la duración del sprint terminado.
En ella el equipo realiza autoanálisis de su forma de trabajar, e identifica fortalezas y puntos débiles. El objetivo es consolidar y afianzar las primeras, y planificar acciones de mejora sobre los segundos.

5. DESARROLLO DEL PROYECTO

5.1. CRONOGRAMA DE ACTIVIDADES:

Figura 5. Cronograma de Actividades.

ACTIVIDADES	MES 1				MES 2				MES 3				MES 4			
	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
ETAPA DE INICIO																
Definir título y objetivos del proyecto																
Definir metodología de trabajo																
Etapa de planeación																
Definir actividades																
Estimar duración de las actividades																
Definir la arquitectura del software																
Fijar alcance del proyecto																
Adecuar el servidor de pruebas																
Identificación de la base de datos (de la empresa)																
Definición Modelo entidad relación																
Etapa de desarrollo																
Selección de herramientas de desarrollo																
Desglose de la recarga																
Reunión Mensual																
creación de la base de datos (propia)																
Desarrollo de consultas para la base de datos																
Desarrollo plataforma web (backend)																
Formato de envío y recepción de peticiones																
Creación y almacenamiento de usuarios																
Conexión con la base de datos																
Creación del web Service																
Desarrollo de las interfaces																
Conexión de la tarjeta con el sistema																
Reunión Mensual																
Integración servidor/web/medidor																

Realización de pruebas																
Seguimiento y control																
Planificación del Sprint 1																
Desarrollo del Sprint 1																
Sprint review																
Planificación del Sprint 2																
Desarrollo del Sprint 2																
Sprint review																
Planificación del Sprint 3																
Desarrollo del Sprint 3																
Sprint review																
Planificación del Sprint 4																
Desarrollo del Sprint 4																
Sprint review																
CIERRE																
Entrega Oficial del proyecto																
Documento de cierre de proyecto																

Fuente: Propia

5.1.1. ETAPA DE INICIO:

Para la realización del proyecto, se tomaron en cuenta las diferentes metodologías de desarrollo de software, con el fin de cumplir con todos los objetivos definidos se decidió trabajar con la metodología Scrum, que es la base sobre la cual se desarrolló el proyecto ya que describe los procesos que involucran la administración y desarrollo del mismo.

5.1.2. ETAPA DE PLANEACIÓN:

En esta etapa se tomaron en cuenta todos los aspectos necesarios para llevar a cabo un desarrollo basado en buenas prácticas, se tuvieron en cuenta las tecnologías a usar, para así mismo adecuarlas en el servidor empresarial donde corre el aplicativo, además del tiempo estimado para cada actividad.

5.1.3. ETAPA DE DESARROLLO:

Teniendo en cuenta el alcance, objetivos y las adecuaciones pertinentes del proyecto, se procedió a escoger aquellas herramientas tecnológicas que sean acorde con los requerimientos exigidos por la empresa, permitiendo flexibilidad, seguridad y robustez al producto final de software.

El tiempo que se designó para cada una de las tareas fue acorde con la experiencia académica en la realización de proyectos, en este tiempo se diseñaron las estrategias pertinentes y se dividieron las diferentes tareas de acuerdo a las fortalezas del equipo. Aparte de ello se dieron avances significativos para las respectivas revisiones, como lo demanda la metodología Scrum.

5.1.4. ETAPA DE CIERRE:

Es la etapa se realizó una entrega formal del aplicativo con sus respectivos manuales ante la Empresa de Acueducto y Alcantarillado de Villavicencio, se redacta el documento de cierre y se recibe el aval como soporte de que el aplicativo se entregó en correcto funcionamiento y en los plazos definidos inicialmente.

5.2. EJECUCIÓN.

Siguiendo la metodología de desarrollo ágil Scrum, se realizaron mejoras conforme se avanzaba durante el proceso y se obtuvo la retroalimentación del cliente. Se utilizaron las historias de usuario que sirven de soporte para el desarrollo y los procedimientos asociadas a este.

Inicialmente se tomaron las especificaciones exigidas por la empresa (EAAV) en cuanto a tecnologías y a requerimientos para iniciar el desarrollo del proyecto.

En el transcurso del desarrollo del proyecto, se realizaron 4 Sprints, con una duración promedio de 3 semanas y media cada uno. En cada Sprint se realizó la planeación correspondiente y un desarrollo funcional del web service acorde con los requerimientos base, junto con los que surgieron con el transcurso del tiempo.

Al finalizar cada iteración los resultados obtenidos eran expuestos y explicados a las dependencias correspondientes con el fin de adaptar y de ser necesario replantear las siguientes fases del proyecto, esto permite reprogramar las iteraciones posteriores en base a las opiniones, sugerencias y nuevos requerimientos que no se tuvieron en cuenta al momento de la planeación del alcance del proyecto.

5.3. SPRINT 1

Para la realización del proyecto piloto, el cual pretende ser una estrategia para la amortiguación de la cartera de la empresa, fue necesario una explicación y exposición a los diferentes procesos los cuales involucran este

proyecto, las herramientas tecnológicas a disposición y adaptación a ellas, la respectiva explicación de la finalidad e impacto del proyecto.

Posterior a ello se procedió al desarrollo del web service para para la extracción de información de usuarios del proyecto piloto de servicio de agua prepago.

Tabla 1. Listas de tareas del Proyecto

Historia de Usuario	Tarea	Responsable	Estado	Días / Estimación (Horas)									
				Total	1	2	3	4	5	6	7	8	9	10
SP2-01	Hablar con la DBA a cargo para explicación del cobro de Cartera.	Daniela	Planificada											
SP2-02	Verificar toda la información de cobro de una determinada factura se encuentre en la base de datos.	Jhon	Planificada											
SP2-03	Hablar con los implicados en el desarrollo de la plataforma para conocer los datos que han de requerir los roles de la plataforma.	Jhon	Planificada											
SP2-04	Solicitar creación de usuario, en el motor de base de datos, con permisos de lectura.	Jhon	Planificada											
SP2-04	Realizar la consulta pertinente que satisfaga los datos solicitados.	Daniela	Planificada											
SP2-04	Verificar que los datos de la cita sean coherentes.	Daniela	Planificada											
SP2-04	Documentar la consulta resultante.	Daniela	Planificada											
SP2-05	Realizar la consulta pertinente que satisfaga los datos solicitados.	Jhon	Planificada											
SP2-05	Empalmar datos con una determinada factura.	Jhon	Planificada											
SP2-05	Documentar la consulta resultante.	Jhon	Planificada											

SP2-06	Realizar la consulta pertinente que satisfaga los datos solicitados.	Daniela	Planificada										
SP2-06	Documentar la consulta resultante.	Daniela	Planificada										
SP3-01	Adecuar el servidor de pruebas con el framework de desarrollo Django vs 1.8.	Jhon	Planificada										
SP3-01	Instalar las dependencias requeridas para la versión del SGB (controladores)	Jhon	Planificada										
SP3-01	Realizar algoritmo en el framework que permita sacar la información requerida y realizar él envió necesario.	Jhon	Planificada										
SP3-01	Validar la información correspondiente concuerda.	Jhon	Planificada										
SP3-01	Documentar el algoritmo correspondiente.	Jhon	Planificada										
SP3-02	Realizar algoritmo en el framework que permita sacar la información requerida y realizar él envió necesario.	Daniela	Planificada										
SP3-02	Validar la información correspondiente concuerda.	Daniela	Planificada										
SP3-02	Validar la información correspondiente concuerda.	Daniela	Planificada										
SP4-01	Recoger los diferentes modelos relacionales acordados para la plataforma.	Jhon	Planificada										
SP4-01	Definir los datos que se deben almacenar	Jhon	Planificada										
SP4-01	Realizar modelo entidad relación de la base de datos propia para el proyecto.	Daniela											
SP4-01	Socializar modelo	Daniela	Planificada										

	entidad relacional con el equipo de desarrollo												
SP4-02	Realizar los Scripts correspondientes para el ingreso de información recogida por la plataforma.	Jhon	Planificada										
SP4-02	Validar los campos a almacenar enviados por la plataforma.	Daniela	Planificada										
SP4-02	Realizar las consultas requeridas para la información requerida por el equipo de desarrollo	Daniela	Planificada										
SP4-02	Validar la información correspondiente	Daniela	Planificada										

5.4. Sprint 2

5.4.1. Planificación del Sprint

Para este Sprint se realizaron las historias de usuario correspondiente, tomando las de mayor prioridad y de acuerdo con lo requerido para la respectiva entrega funcional y con las adecuaciones de la plataforma pertinentes. A Continuación, se muestran las historias de usuario a desarrollar en este Sprint Ver tablas 2, 3, 4, 5, 6 y 7. A partir de estas se realizaron la lista de tareas de dicho Sprint que se pueden ver en la tabla 8.

Tabla 2. Historia de usuario para Reconocimiento de información relacionada al proceso de cartera de la empresa.

Título:	Reconocimiento de información relacionada al proceso de cartera de la empresa.	**ID:**	SP2-01
Funcionario:		**Dependencia:**	Oficina de sistemas.
Tipo:	Épico Historia X	**No. Requisito:**	
Descripción:			
Para poder designar los diferentes rubros a cobrar al respectivo cliente, se requiere conocer cómo se cobra y que se le cobra			
Criterios de aceptación:			
N. A.			

Fuente: Propia

ºTabla 3. Historia de usuario para Desglose de información en el proceso de facturación.

Título:	Desglose de información en el proceso de facturación.	**ID:**	SP2-02
Funcionario:		**Dependencia:**	Oficina de sistemas.

Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Para hacer un desglose adecuado de la recarga del usuario, en cuanto a tarifas fijas, aportes subsidios y demás.					
Criterios de aceptación:					
N. A.					

Fuente: Propia

Tabla 4. Historia de usuario para Adaptación de roles correspondientes de la plataforma.

Título:	Adaptación de roles correspondientes de la plataforma.			**ID:**	SP2-03
Funcionario:				**Dependencia:**	Oficina de sistemas.
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Los roles definidos para la plataforma del proyecto piloto deben de tener una alimentación de datos provenientes de la información de los diferentes usuarios de la empresa, para ello se debe definir tales datos.					
Criterios de aceptación:					
N. A.					

Fuente: Propia

Tabla 5. Historia de usuario para Creación de las consultas SQL para la información correspondiente a la cartera de un determinado cliente

Título:	Creación de las consultas SQL para la información correspondiente a la cartera de un determinado cliente.			**ID:**	SP2-04
Funcionario:				**Dependencia:**	Oficina de sistemas.
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Se requiere que con el código de usuario se pueda visualizar el monto total de la deuda que tiene ese usuario con la empresa, los conceptos a los cuales pertenece, las deudas que acumula y los intereses pertinentes según sea la estructura de cobro en cartera.					
Criterios de aceptación:					
N. A.					

Fuente: Propia

Tabla 6. Historia de usuario para Creación de las consultas SQL que extraigan la información concerniente al cobro de metros cúbicos de agua.

Título:	Creación de las consultas SQL que extraigan la información concerniente al cobro de metros cúbicos de agua.			**ID:**	SP2-05
Funcionario:				**Dependencia:**	Oficina de sistemas.
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Mostrar la información exclusiva de cobro de metros cúbicos de agua, donde se pueda visualizar el monto por metro cubico de agua y alcantarillado, los subsidios o aportes correspondientes.					

Criterios de aceptación:
N. A.

Fuente: Propia

Tabla 7. Historia de usuario para Creación de las consultas SQL que muestren información concerniente de un determinado usuario de la empresa.

Título:	Creación de las consultas SQL que muestren información concerniente de un determinado usuario de la empresa.	**ID:**	SP2-06
Funcionario:		**Dependencia:**	Oficina de sistemas.
Tipo:	Épico Historia X	**No. Requisito:**	
Descripción:			
Se requiere mostrar toda la información básica de los usuarios, esta información debe ser: tipo de usuario, código, documento de identidad, nombres, apellidos y estrato.			
Criterios de aceptación:			
N. A.			

Fuente: Propia

Tabla 8. Listas de tareas del Sprint dos.

Historia de ºsuario	Tarea	Responsable	Estado	Días / Estimación (Horas)									
				Total	1	2	3	4	5	6	7	8	9	10
SP2-01	Hablar con la DBA a cargo para explicación del cobro de Cartera.	Daniela	Planificada											
SP2-02	Verificar toda la información de cobro de una determinada factura se encuentre en la base de datos.	Jhon	Planificada											
SP2-03	Hablar con los implicados en el desarrollo de la plataforma para conocer los datos que han de requerir los roles de la plataforma.	Jhon	Planificada											
SP2-04	Solicitar creación de usuario, en el motor de base de datos, con permisos de lectura.	Jhon	Planificada											
SP2-04	Realizar la consulta pertinente que satisfaga los datos solicitados.	Daniela	Planificada											
SP2-04	Verificar que los datos de la cita sean	Daniela	Planificada											

	coherentes.												
SP2-04	Documentar la consulta resultante.	Daniela	Planificada										
SP2-05	Realizar la consulta pertinente que satisfaga los datos solicitados.	Jhon	Planificada										
SP2-05	Empalmar datos con una determinada factura.	Jhon	Planificada										
SP2-05	Documentar la consulta resultante.	Jhon	Planificada										
SP2-06	Realizar la consulta pertinente que satisfaga los datos solicitados.	Daniela	Planificada										
SP2-06	Documentar la consulta resultante.	Daniela	Planificada										

Fuente: Propia

5.4.2. Desarrollo del Sprint.

5.4.2.1. Adaptación y construcción de información necesaria para el proyecto piloto.

Para la correcta funcionalidad de la plataforma, es necesario hacer uso de datos reales que pertenezcan a la empresa (EAAV). Para esto se hizo una integración previa con el sistema de gestión de base de datos (SGBD) que la empresa designo para el proyecto y donde se encuentran todos los datos de los usuarios, teniendo siempre cuidado con la integridad de la información, sabiendo la importancia de ella.

Teniendo la adaptación y el reconocimiento de la base de datos realizado, se procedió al levantamiento de requerimientos para el Sprint 1, los cuales inicialmente consistieron en la recolección de datos básicos de usuarios, tarifas y cartera. Se adaptaron dichos datos a la plataforma.

Después de obtener la información necesaria se procedió a la realización del empalme correspondiente a esta, corroborando con facturas de prueba y usuarios con deuda en cartera teniendo total veracidad de la información obtenida por las consultas desarrolladas para obtener dichos datos.

5.4.3. Revisión del Sprint.

Se realizó una reunión entre el Scrum Master, el Product Owner, equipo de desarrollo y diferentes dependencias de la empresa donde se presentó los avances correspondientes del proyecto.

Las personas que asistieron a la reunión aceptaron las funcionalidades desarrolladas en este sprint, dando como resultado una serie recomendaciones para la mejora de dicha funcionalidad y dudas con respecto a algunos aspectos del uso y del manejo del sistema a futuro que no se tuvieron en cuenta, junto con algunas obviedades que algunos asistentes no tenían bien definidas.

Al finalizar la reunión se da el aval para continuar y empezar a planificar el siguiente sprint

5.4.4. Retrospectiva del Sprint.

Se realizó una reunión con el Scrum Master y el equipo de desarrollo donde se presentaron las dificultades y fortalezas que se tuvieron en el desarrollo del sprint.

En el transcurso del sprint se pudo evidenciar algunas dificultades con el manejo de las tecnologías dispuestas para él proyecto, pero la curva de aprendizaje pudo ser superada gracias al asesoramiento de algunos funcionarios de la empresa.

5.5. Sprint 3.

5.5.1. Planificación del Sprint.

En este sprint se asignaron historias de usuario de los nuevos requerimientos y modificaciones las cuales surgieron como resultado del sprint 2, junto con las historias de usuario las cuales daban la continuidad del proyecto, en donde se tienen en cuenta las interfaces y procesos a manejar por la plataforma. A continuación, se muestran las historias de usuario a desarrollar en el sprint, ver Tablas 9 y 10. A partir de estas historias de usuario se hizo la respectiva lista de tareas como se muestra en la Tabla 11.

Tabla 9. Historia de usuario Envió información de un determinado usuario.

Título:	Envió información básica de un determinado usuario		**ID:**	SP3-01
Funcionario:			**Dependencia:**	Oficina de sistemas
Tipo:	Épico	Historia X	**No. Requisito:**	
Descripción:				
La plataforma debe poder visualizar campos con la información de un usuario cualquiera, para ello se deberá enviar la información requerida y correspondiente recibiendo el código pertinente de dicho usuario				
Criterios de aceptación:				
El sistema debe permitir ver el detalle de la información requerida para un determinado usuario, esa información únicamente debe ser el código, nombre completo y barrio				

Fuente: Propia

Tabla 10. Historia de usuario Envío información de cartera un determinado usuario.

Título:	Envío información de cartera un determinado usuario			**ID:**	SP3-02
Funcionario:				**Dependencia:**	Oficina de sistemas
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
La plataforma debe poder visualizar algo concreto y resumido de la deuda en cartera que posea un cliente determinado. Para ello se recibirá el código de dicho usuario.					
Criterios de aceptación:					
El sistema debe permitir ver el monto total de la cartera que posea un cliente.					

Fuente: Propia

Tabla 11. Tabla de tareas para el sprint 3

Historia de Usuario	Tarea	Responsable	Estado	Días / Estimación (Horas)									
				Total	1	2	3	4	5	6	7	8	9	10
SP3-01	Adecuar el servidor de pruebas con el framework de desarrollo Django vs 1.8.	Jhon	Planificada											
SP3-01	Instalar las dependencias requeridas para la versión del SGB (controladores)	Jhon	Planificada											
SP3-01	Realizar algoritmo en el framework que permita sacar la información requerida y realizar él envió necesario.	Jhon	Planificada											
SP3-01	Validar que la información correspondiente concuerde.	Jhon	Planificada											
SP3-01	Documentar el algoritmo correspondiente.	Jhon	Planificada											
SP3-02	Realizar algoritmo en el framework que permita sacar la información requerida y realizar él envió necesario.	Daniela	Planificada											
SP3-02	Validar que la información correspondiente concuerde	Daniela	Planificada											
SP3-02	Validar que la	Daniela	Planificada											

	información correspondiente concuerde.													

Fuente: Propia

5.5.2. Desarrollo del Sprint

5.5.3. Planeación y desarrollo de módulo “Extracción y envió de información”

Para la realización del módulo requerido por la plataforma, se adecuo el servidor de pruebas de la empresa (EAAV) con el framework de desarrollo Django versión 1.8. Este fue elegido dado que es una herramienta en auge y ha dado gran acogida en el entorno de desarrollo puesto que ofrece seguridad, facilidad de navegación, manejo de usuarios e interacción con la base de datos. Aparte de ello posee una gran integración con otras herramientas de desarrollo dejando abierta la posibilidad de realizar una solución de software completa y variada acorde con los requerimientos exigidos.

La plataforma en su diseño preliminar requiere ciertos tipos de datos, dependiendo del módulo, estos pueden variar. En el caso de módulos correspondientes al usuario se debe tomar obligatoriamente el código del usuario para el tratamiento, petición y modificación de datos. La plataforma debe garantizar él envió del código pertinente del usuario a tratar.

Teniendo lo anterior y con la correcta adecuación de las herramientas necesarias se procede a la creación de las Apps en Django las cuales fueron previamente habladas y discutidas según su funcionalidad. Al crear la(s) App se asocian ciertos módulos en python los cuales permiten toda la creación de proyecto en sí.

En dichos módulos de python se procede a la realización del algoritmo de consulta, donde se usa el código del cliente para la búsqueda de información y se trae lo necesario para responder. La petición es enviada en el formato pertinente donde la plataforma lo interpreta y realiza la visualización correspondiente de los datos enviados. Esto se realiza tanto para los datos de información de código, nombre completo y barrio del usuario a consultar y la información de cartera del mismo.

5.5.4. Revisión del Sprint.

Se realizó una reunión con el Scrum Master, el Product Owner, equipo de desarrollo. Donde se presentó el proyecto integrado y funcional de las interfaces de la plataforma junto con aquello datos básicos de consulta requeridos para la respectiva entrega.

Las personas que asistieron a la reunión aceptaron las funcionalidades y reportes desarrollados en este sprint, donde no hubo recomendaciones para dichas funcionalidades y quedaron satisfechos con el trabajo realizado.

5.5.5. Retrospectiva del Sprint.

Se realizó una reunión con el Scrum Master y el equipo de desarrollo donde se presentaron las dificultades, fortalezas presentadas en el desarrollo del sprint.

Para este sprint hubo mejoras en cuanto a las dificultades presentadas en el sprint anterior. Solo un poco de inconvenientes técnicos en cuanto al uso del framework de desarrollo dado que fue una nueva herramienta tecnológica la cual todo el equipo se tuvo que acoplar. Se realizó un buen trabajo cumpliendo con las funcionalidades planeadas.

5.6. Sprint 4.

5.6.1. Planificación del Sprint

Para este Sprint se realizaron las historias de usuario correspondiente, tomando las de mayor prioridad y de acuerdo con lo requerido para la respectiva entrega funcional y con las adecuaciones de la plataforma pertinentes. A continuación, se muestran las historias de usuario a desarrollar en el sprint, ver Tablas 12 y 13. A partir de estas historias de usuario se hizo la respectiva lista de tareas como se muestra en la Tabla 14.

Tabla 12. Historia de usuario Almacenamiento de información recolectada de la plataforma.

Título:	Almacenamiento de información recolectada de la plataforma.			**ID:**	SP4-01
Funcionario:				**Dependencia:**	Oficina de sistemas
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Dado que la plataforma debe procesar y recolectar cierta información concerniente a el monto de recarga para la compra de agua, los diferentes puntos de recaudadores y demás datos de la plataforma, se debe registrar toda esa información.					
Criterios de aceptación:					

Fuente: Propia

Tabla 13. Historia de usuario Consulta de información pertinente al proyecto piloto. .

Título:	Consulta de información pertinente al proyecto piloto.			**ID:**	SP4-02
Funcionario:				**Dependencia:**	Oficina de sistemas
Tipo:	Épico	Historia	X	**No. Requisito:**	
Descripción:					
Como una de las funcionalidades que exige el proyecto es poder visualizar todo los consumos, los					

montos invertidos tanto de clientes como de recaudadores, la información aportada a la deuda de cartera. Se deben realizar las consultas pertinentes para la visualización de toda esta información.
Criterios de aceptación:

Fuente: Propia

Tabla 14. Tabla de tareas para el sprint 4

Historia de Usuario	Tarea	Responsable	Estado	Días / Estimación (Horas)									
				Total	1	2	3	4	5	6	7	8	9	10
SP4-01	Recoger los diferentes modelos relacionales acordados para la plataforma.	Jhon	Planificada											
SP4-01	Definir los datos que se deben almacenar	Jhon	Planificada											
SP4-01	Realizar modelo entidad relación de la base de datos propia para el proyecto.	Daniela	Planificada											
SP4-01	Socializar modelo entidad relacional con el equipo de desarrollo	Daniela	Planificada											
SP4-02	Realizar los Scripts correspondientes para el ingreso de información recogida por la plataforma.	Jhon	Planificada											
SP4-02	Validar los campos a almacenar enviados por la plataforma.	Daniela	Planificada											
SP4-02	Realizar las consultas requeridas para la información requerida por el equipo de desarrollo	Daniela	Planificada											
SP4-02	Validar la información correspondiente	Daniela	Planificada											

Fuente: Propia.

5.6.2. Desarrollo de Sprint

5.6.2.1. Modelo relacional.

ANEXO A: Modelo Relacional

5.6.2.2. Desarrollo del módulo de envío y recepción de datos en la plataforma.

Según lo acordado con el equipo de desarrollo de la plataforma web para el proyecto piloto de agua prepago, se debía registrar ciertos datos y sobre ellos realizar una visualización en la plataforma. Esto con el fin de que los diferentes roles puedan observar estadísticas, ver montos de cartera aportados, montos de cartera a deber, metros de agua consumidos. Etc. Todos aquellos datos los cuales permitan dar toda una información completa, resumida y veras en la plataforma.

Se realizaron las diferentes aplicaciones las cuales contenían los algoritmos, y sentencias SQL. Esto gracias a los diferentes métodos Get y Post web correspondientes para realizar todo lo requerido para el ultimo requerimiento del proyecto.

5.6.3. Revisión del Sprint.

Se realizó una reunión con el Scrum Master, el Product Owner, equipo de desarrollo. Donde se presentó el proyecto integrado y funcional de las interfaces de la plataforma junto con los requerimientos finales para la entrega final.

Las personas que asistieron a la reunión aceptaron las funcionalidades y reportes desarrollados en este sprint, donde hubo ciertas recomendaciones en cuanto entorno visual por el lado de la plataforma. Estos cambios eran por menores dado que solo diferían al lugar donde debería ubicarse el logo de la empresa y usar los coles correspondientes de la empresa a partir de la paleta de colores que se proporcionó en esa reunión.

5.6.4. Retrospectiva del Sprint.

Se realizó una reunión con el Scrum Master y el equipo de desarrollo donde se presentaron las dificultades, fortalezas presentadas en el desarrollo del Sprint final.

Para este sprint hubo muchos inconvenientes dados a la mala planeación de tiempo a las diferentes actividades requeridas para este. Aparte de ello se atrasó un poco el desarrollo de los módulos correspondientes dado que los permisos necesarios para la creación, escritura y lectura de datos en el motor de base de datos, fueron un poco lentos. Las demás actividades se llevaron a cabo con éxito y se logró respetar los tiempos establecidos.

6. RESULTADOS OBTENIDOS.

- Se creó un Script que lleva inmersas las consultas realizadas a la base de datos empresarial que se encarga de extraer la información de suscriptor y valores de cartera solicitados por la plataforma.

- La creación de la base datos de la plataforma se realizó con SQL Server, ciñéndonos al SGBD utilizado por la entidad, se realizó el modelo entidad de relación y el modelo relacional como guía para el desarrollo del Script, teniendo en cuenta las variables involucradas en todo el proceso de la plataforma.

- Teniendo en cuenta que la inserción de información concerniente al proyecto piloto del servicio de agua prepago se llevara a cabo directamente desde la plataforma según el proceso que se realizó, se codifico un script en Python que permite la interacción con la base de datos propia de la plataforma, tomando las credenciales del sistema y permitiendo la inserción de los históricos de cada rol.

7. CONCLUSIONES

En esta pasantía se logró adquirir conocimiento sobre el manejo de tecnologías web, que sobresalen en la actualidad, además se aprendió a conocer el ámbito laboral y a seguir un proceso riguroso en cuanto a la documentación y metodología del desarrollo de software.

Como enseñanza quedo el llevar a cabo un proceso estricto en cuanto al levantamiento de requerimientos solicitados por parte del contratante y a tener constancia escrita de los acuerdos realizados en dicho proceso.

Se vivió la experiencia de llevar un conducto regular en cuanto al uso de metodologías de desarrollo ágil de software, llevando a cabo las reuniones periódicas y teniendo un cliente que solicite cambios y modificaciones sobre lo que ya se desarrolló, precisamente por esto se insiste en tener documentos por escrito con los acuerdos entre las partes.

Para la correcta realización del proceso de pasantía es necesario llegar a un mutuo acuerdo con los términos de confidencialidad, dado que puede presentar inconvenientes para las partes participantes en dicho proceso.

8. REFERENCIAS.

[1] EAAV, “EAAV: Cobertura.” [Online]. Available: http://www.eaav.gov.co/index.php?id=149. [Accessed: 19-Nov-2016].

[2] EAAV, “EAAV: EAAV sigue trabajando para reducir los índices de cartera.” [Online]. Available: http://www.eaav.gov.co/index.php?id=28&tx_ttnews%5Btt_news%5D=1280&cHash=908ba36c5f49b9a9e560c8ee2aa7ef34. [Accessed: 19-Nov-2016].

[3] L. Franek, L. Šťastný, and P. Fiedler, “Prepaid energy in time of Smart Metering,” pp. 428–433, 2013.

[4] B. J. Kao, “Developer â€TM s Guide to Building XML-based Web Services with the Java 2 Platform , Enterprise Edition (J2EE),” no. June, pp. 1–21, 2001.

[5] A. Chernyshov, A. Balandina, and A. Kostkina, “Intelligence Search Engine and Automatic Integration System for Web-Services and Cloud-Based Data Pro- viders Based on Semantics,” vol. 88, pp. 272–276, 2016.

[6] I. Golitsyna, “Application of Web Services in Teaching of IT-Discipline,” vol. 214, no. June, pp. 578–585, 2015.

[7] Longman Wesley Addison, “Chapter 2-A history of HTML,” 1998. [Online]. Available: https://www.w3.org/People/Raggett/book4/ch02.html. [Accessed: 15-Nov-2016].

[8] H. W. Lie and B. Bos, *Cascading style sheets : designing for the Web*. Addison-Wesley, 1999.

[9] Php, “PHP: Historia de PHP - Manual,” 2004. [Online]. Available: http://php.net/manual/es/history.php.php. [Accessed: 15-Nov-2016].

[10] Departamento de Lenguajes y Sistemas Informáticos, “Bases de Datos Modelo relacional de Codd Estructuras y restricciones,” pp. 1–14, 2004.

[11] W3C, “A Short History of JavaScript - Web Education Community Group,” 2012. [Online]. Available: https://www.w3.org/community/webed/wiki/A_Short_History_of_JavaScript. [Accessed: 15-Nov-2016].

[12] “jQuery.” [Online]. Available: https://jquery.com/. [Accessed: 15-Nov-2016].

[13] J. Gutiérrez, “¿ Qué es un framework web?,” pp. 1–4, 2006.

[14] G. Martínez Villalobos, G. D. Camacho Sánchez, and D. A. Biancha Gutiérrez, “Diseño de Framework web para el desarrollo dinámico de aplicaciones,” *Sci. Tech.*, vol. XVI, no. 44, pp. 178–183, 2010.

[15] L. Delía, G. Cáseres, H. Ramón, P. Thomas, and R. Bertone, “Framework para el Desarrollo Ágil de Aplicaciones Web,” pp. 289–299, 2015.

[16] J. Sutherland, “The Scrum Guide TM,” no. July, 2013.

[17] M. Trigas Gallego and A. C. Domingo Troncho, “Gestión de Proyectos Informáticos. Metodología Scrum.”

[18] PmoInformatica, “Plantillas Scrum: Pila de producto (Product Backlog) - La Oficina de Proyectos de Informática.” [Online]. Available: http://www.pmoinformatica.com/2013/11/plantillas-scrum-pila-producto-product.html. [Accessed: 07-Jul-2017].

[19] PmoInformatica, “Plantillas Scrum: Lista de tareas de la iteración (sprint backlog) - La Oficina de Proyectos de Informática,” 2016. [Online]. Available: http://www.pmoinformatica.com/2016/11/plantillas-scrum-sprint-backlog.html#more. [Accessed: 07-Jul-2017].