**CAPITULO 4- METODOLOGIA DE DESARROLLO.**

La metodología de desarrollo elegida para el presente proyecto es SCRUM. Este es un marco de trabajo para la gestión y [desarrollo de software](http://es.wikipedia.org/wiki/Desarrollo_de_software) basada en un proceso [iterativo e incremental](http://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente) utilizado comúnmente en entornos basados en el [desarrollo ágil de software](http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software) [Schwaber, 2013].

En SCRUM hay tres aspectos fundamentales a diferenciar, los actores, los eventos o acciones y los artefactos. Los actores son los que ejecutarán las acciones para alcanzar los objetivos del proyecto y con ello se genera los artefactos que servirán para planificar o organizar el desarrollo del proyecto.

**Los actores**

**Product Owner:** representa a los *stakeholders* y conoce los requerimientos de alto nivel del producto marcando las prioridades del mismo.

**Scrum Master:** es la persona que asegura el seguimiento de la metodología guiando las reuniones y ayudando al equipo ante cualquier problema que pueda aparecer. Su responsabilidad es entre otras, la de hacer de paraguas ante las presiones externas.

**Scrum Team:** son las personas responsables de implementar la funcionalidad o funcionalidades elegidas por el Product Owner.

**Usuarios o Cliente:** son los beneficiarios finales del producto, y son quienes viendo los progresos, pueden aportar ideas, sugerencias o necesidades [Schwaber, 2013].

**Los eventos o acciones:**

Daily Scrum Meeting: Es una tarea iterativa que se realiza todos los días que dure el Sprint Backlog con el equipo de desarrollo o de trabajo. Es una reunión operativa, informal y ágil, de un máximo de 30 minutos, en la que se hace 3 preguntas a cada integrante del equipo:

* Que tareas se ha realizado desde la última reunión.
* Sobre qué se va a trabajar en el día actual.
* Identificación de obstáculos o riesgos que impiden o pueden impedir el normal avance del proyecto. El Scrum Master debe eliminar cualquier obstáculo que encuentre.

**Sprint Planning Meeting:** es una reunión que tiene por objetivo planificar el Sprint a partir del Product Backlog. El objetivo de esta reunión es la de mover las tareas del Product Backlog al Sprint Backlog. En esta reunión, suelen participar el Product Owner que es quien prioriza las tareas.

Sprint Goal: es un pequeño documento o una breve descripción que indica lo que el Sprint Backlog intentará alcanzar.

***Sprint Review o Demo*:** se revisa en unas 2 horas como máximo el Sprint finalizado. Al llegar a este punto, se debe tener "algo" que el Cliente o el Usuario pueda ver y tocar. En esta reunión, suelen asistir el Product Owner, el Scrum Master, el Scrum Team y personas que podrían estar involucradas en el proyecto. El Scrum Team es quién muestra los avances realizados en el Sprint.

**Sprint Retrospective**: Al finalizar un Sprint Backlog y el Sprint Review, se inicia el **Sprint Retrospective.** El Product Owner revisará con el equipo los objetivos marcados inicialmente en el Sprint Backlog concluido, se aplicarán los cambios y ajustes si son necesarios, y se marcarán los aspectos positivos (para repetirlos) y los aspectos negativos (para evitar que se repitan) del Sprint. [Schwaber, 2013].

**Los artefactos**:

Product Backlog: Corresponde con todas las tareas, funcionalidades y requerimientos a realizar por el equipo de desarrollo. El Product Owner es la persona que se encarga de marcar las prioridades y de mantener y actualizar la lista de tareas.

Sprint Backlog: Corresponde con una o más tareas que provienen del Product Backlog. El Sprint Backlog tiene una duración de 2 a 4 semanas que al momento de iniciar, este no puede ser alterado o modificado en su transcurso. Se tiene que esperar a que concluya el Spring Backlog para realizar la correspondiente modificación o alteración cuyas tareas formaría para del siguiente Spring Backlog. Cuando se ha finalizado un Sprint Backlog, se debe tener un entregable o algo que se pueda mostrar y que enseñe los avances acometidos en el Sprint.

Incremento:

El Incremento es la suma de todos los elementos del Product Backlog completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint, el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado por el dueño del producto. [Schwaber, 2013].

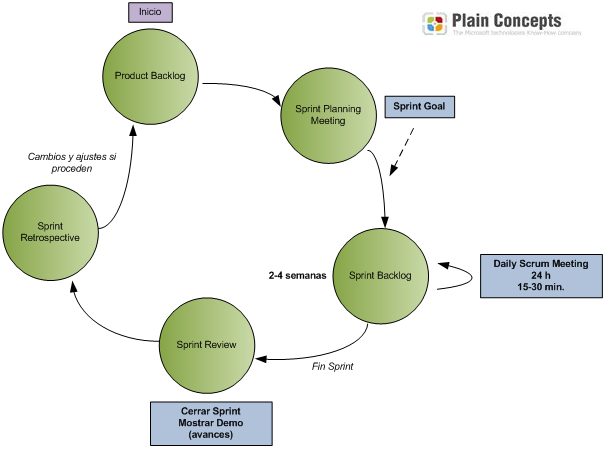
****

Figura 4.1: Flujo de Acciones en un proyecto de Scrum [Schwaber, 2013].

**Planning Poker**

 Es una técnica para calcular una estimación basada en el consenso, en su mayoría utilizada para estimar el esfuerzo o el tamaño relativo de las tareas de [desarrollo de software](http://es.wikipedia.org/wiki/Desarrollo_de_software).

El Planning Poker se basa en una lista de tareas a ser estimado y una baraja de cartas. Un mazo típico contiene tarjetas mostrando la secuencia de Fibonacci incluyendo un cero: 0, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89. Otros mazos utilizan progresiones similares. La razón de utilizar la secuencia de Fibonacci es reflejar la incertidumbre inherente en la estimación. Un mazo que se encuentra en el mercado utiliza la siguiente secuencia: 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, y, opcionalmente, una tarjeta, una con signo de interrogación (?) y otra con signo de infinito (∞), que pueden ser usadas para declarar completa incertidumbre o desconocimiento de la característica. [Piedrasanta, 2011]

Los recursos para trabajar con la técnica Planning Poker son:

* Cartas Planning Poker
* Dos a cuatro participantes
* Un moderador o líder del proyecto.
* Un reloj o algún elemento que determine el tiempo.

**Procedimiento de uso**

En la reunión de la estimación a cada participante se le da un conjunto completo de tarjetas.

La reunión prosigue de la siguiente manera:

* Un moderador, que no jugará, preside la reunión. Puede ser el perfil del Líder de Equipo o Líder de Proyecto.
* El desarrollador con más conocimiento de una determinada característica proporciona una breve introducción sobre la misma. El equipo tiene la oportunidad de hacer preguntas y discutir para aclarar los supuestos y riesgos. Un resumen de la discusión es registrado por el moderador.
* Cada persona coloca una tarjeta boca abajo que representa su estimación. Las unidades utilizadas pueden ser variadas y definidas previamente. Pueden ser días de duración, días ideales o puntos de la historia. Durante el debate, los números no debe ser mencionados en absoluto.
* Todo el mundo muestra sus tarjetas de forma simultánea.
* A las personas con estimaciones altas y bajas se les da un tiempo para ofrecer su justificación para la estimación y la discusión continúa. Como máximo 15 minutos.
* Se repita el proceso de cálculo hasta que se alcance un consenso. El programador que probablemente tenga el entregable tiene una gran parte del voto de consenso, aunque el moderador puede negociar el consenso.
* Se puede utilizar un reloj de arena para asegurar que el debate sea estructurado. Una ronda de Poker finaliza cuando se llego a un acuerdo sobre la estimación o se termino el tiempo del reloj. Se puede dejar una funcionalidad sin consenso y volver después de terminar con las demás tareas. [Piedrasanta, 2011]

**Aplicación de la Metodología de Desarrollo**

**Actores:**

De definió los roles de Scrum de la siguiente manera

|  |  |
| --- | --- |
| **SCRUM** | |
| **Product Owner** | Pablo Chávez(Dueño del Restaurante). |
| **Scrum Master** | Elizabet Reinoso. |
| **Scrum Team** | Martin Carrizo y Walter Vale. |
| **Usuarios o Cliente** | Personas con discapacidad auditiva. |

**Product Backlog:**

Las tareas del Product Backlog se definieron en una reunión con el Product Owner, Scrum Master y Scrum Team. El Product Owner es quién dio a conocer los requerimientos de alto nivel y priorizó las tareas a desarrollarse. Todas las dudas del Scrum Team acerca de la ambigüedad de los requerimientos fueron disipadas por Product Owner. La reunión fue precedida y coordinada por el Scrum Master. La Tabla 4.1 fue el resultado de la reunión.

La tabla 4.1 está compuesta por la columna ID que funciona para identificar unívocamente las tareas del Product Backlog y la del Spring Backlog. El ESFUERZO ESTIMADO para finalizar una tarea fue calculado utilizando Planning Poker. El CRITERIO DE ACEPTACIÓN es una condición que debe lograrse para que la tarea adopte la categoría de finalizada o terminada.

|  |  |  |  |
| --- | --- | --- | --- |
| **PRODUCT BACKLOG ITEMS** | | | |
| **ID** | **TAREA** | **ESFUERZO ESTIMADO** | **CRITERIO DE ACEPTACIÓN** |
| 0 | Anteproyecto | 72 | Se debe formular un documento según los estándares definidos por el establecimiento. |
| 1 | Investigación y elección de las herramientas para el desarrollo del prototipo | 12 | Investigar las herramientas, Framework y librerías de realidad aumentada más óptimas para el desarrollo del prototipo. Realizar una comparación de las tecnologías y justificar la elección en un documento. |
| 3 | Instalación de las Herramientas para el desarrollo del prototipo. | 5 | Realizar la Instalación de las herramientas necesarias y documentar los pasos de instalación. |
| 4 | Uso y manejo de la interfaz de Blender. | 21 | Se debe alcanzar los conocimientos en el manejo de la interfaz de la herramienta de modelado 3D con el objetivo de modificar los modelos ya existentes en internet. |
| 5 | Uso y manejo de la interfaz de Unity 3D. | 20 | Se debe lograr los conocimientos mínimos en el manejo de la interfaz con el fin de crear una aplicación para Android. |
| 6 | Aprendizaje del SDK de Vuforia | 16 | Se debe estudiar la arquitectura del SDK de Vuforia y la programación básica en el uso de sus funciones. También la creación de una base de datos de imágenes para el reconocimiento de marcadores. |
| 7 | Introducción al lenguaje de programación C# y la plataforma .net de Visual Studio 2010. | 12 | Se debe alcanzar los conocimientos básicos sobre el lenguaje de programación C# y del manejo de interfaz de la IDE de Visual Studio 2010. Manejo en la conexión con la base de datos Mysql, NHibernate para la persistencia de datos, uso de formularios y patrones de diseño. |
| 8 | Selección del lugar para el despliegue del prototipo. | 2 | Se debe contar con un lugar físico para la instalación del sistema y posteriormente realizar las prueba de aceptación con personas con capacidades diferentes. |
| 9 | Obtención y especificación de requerimiento funcional y no funcional. | 20 | Obtener una descripción completa del comportamiento del sistema a través del Product Owner en lenguaje natural y realizar una especificación de los requerimientos obteniendo así los siguientes diagramas: Diagrama de caso de uso, Diagrama de contexto, Diagrama de base de datos y diagrama de clases. |
| 10 | Diseño de la arquitectura de software. | 6 | Se debe crear un diseño de alto nivel de la estructura, funcionamiento e interacción entre las partes del software. |
| 13 | Diseño gráfico de la carta de gourmet. | 12 | Se debe realizar el diseño de la carta gourmet en formato digital. Debe ser atractiva visualmente y contar con la combinación equilibrada de textos e imágenes. El diseño debe ser aprobado por el Product Owner. |
| **16** | **Diseño de la interfaz del sistema en papel.** | **5** | **Diseñar todas las pantallas del sistema.** |
|  |  |  |  |
| **17** | **Investigar Conexión a base de datos** | **10** | **Investigar conexión del dispositivo android con una base de datos local y realizar la conexión.** |
| **18** | **Crear Base de datos para el sistema** | **3** | **crear la base de datos en el servidor para guardar los pedidos.** |
| **19** | **Investigacion del servidor que se va a utilizar.** | **10** |  |
| **20** | **Configuracion del servidor** | **13** | **Se debe configurar el servidor y dejarlo listo para peticiones de base de datos mediate el celular.** |
| **21** | Alta de Pedido | 13 | El cliente podrá realizar un pedido mediante un teléfono celular sin ayuda de un intermediario. El mismo será almacenado en el servidor del sistema. |
| **22** | imprimir Pedido | 8 | El administrador podrá imprimir el pedido para que el chef pueda elaborar el plato. |
| **23** | Modificacion de Pedido | 7 | El cliente podra modificar el pedido antes de ser almacenado en el sistema |
| **24** | Consulta de Pedido por Cliente | 8 | Una ves almacenado el pedido en el sistema, el cliente podrá consultar el pedido |
| **25** | Consulta de Diarios Digitales | 11 | El cliente tendra acceso a los diarios digitales mas leidos de jujuy |
| **26** | Reproducir video de recomendación de chef | 15 | El cliente podra reproducir un video de las recomendaciones del chef o bien de las promociones del restaurante |
| **28** | Navegar por la carta gourmet | 13 | El cliente podra navegar por la carta gourmet apreciando los diferentes platos. Debe incorporarse botones para la navegacion. |
| **29** | Prueba de aceptacion del diseño de la carta gourmet | 2 | El dueño del restaurante debe aceptar el diseño de la carta gourmet. |
| **30** | Login Administrador | 8 | El administrador del sistema podrá logearse para ingresar al sistema. |
| **31** | Consulta de Pedido | 13 | El administrador del sistema podra realizar una consulta de los pedidos realizados por sus clientes. La consulta se realizará entre un intervalo de fechas dado por el administrador. |
| **32** | Imprimir consulta de Pedido | 10 | El administrador del sistema podrá generar un reporte de la consulta de pedido. Este reporte podrá ser impreso. |
| **33** | Agregar video | 7 | El administrador del sistema podrá agregar un nuevo video sobre las recomendaciones del chef. El alta de estos video será realizada del lado del servidor. |
| **35** | Eliminar video | 4 | El administrador del sistema podrá eliminar un video de la base de datos si lo requiere. |
| **36** | Modificar el precio del plato | 6 | El administrador del sistema podrá modificar el precio del plato de la carta gourmet en caso que se lo requiera. Únicamente modificar el campo precio del formulario. |
| **37** | **Creación de la portada para el prototipo al iniciar el sistema e icono.** | **7** | **Seleccionar y diseñar la imagen de inicio del sistema e icono. Incorporacion al prototipo del sistema. Debe quedar visiblemente bien.** |
| **38** | **Documentación del avance del proyecto.** | **6** | **Obtener un listado digital para imprimir y presentar. Refinar la documentacion de las tareas anteriores.** |
| **39** | **Corrección de errores y depuración sistema** | **6** | **Se debe corregir los errrores del sistema.** |
| **40** | **Manual de usuario** | **10** | **Se debe obtener un digital con las instrucciones de instalacion y menejo del sistema.** |
| **41** | **Instalación de sistema en celular y server.** | **2** | **El sistema debe estar en funcionamiento en el celular y en el server. Realiza las comprobaciones de funcionamiento.** |
| **42** | **Resolver el numero el pedido por orden de mesa** | **5** | **Obtener la solucion al problema y programarlo.** |
| **43** | **Generacion del apk y .exe para el sistema.** | **4** | **generar los archivos apk y .exe del sistema** |
| **44** | **Prueba de Aceptación del sistema.** | **3** | **El sistema debe ser aceptado por los usuarios finales.** |
| **45** | **Correcciones para mejorar el sistema (Prueba de Aceptacion)** | **6** | **Se realizara las correcciones necesarios que propongan los usuarios finales** |
| **46** | **Capacitacion del Sistema al Product Owner** | **3** | **Se instruira en el manejo del sistema tanto para el server, como para la tablet.** |
| **47** | **Documentacion de la metodologia de desarrollo (Srum)** | **3** | **Generar un documento sobre la metodologia Scrum y sus procedimiento de ejecucion para el desarrollo de software agil.** |
| **48** | Agregar simbologia de lengua de señas | 13 | Agregar simbolos y imágenes a la aplicación de lengua de señas. |
| **49** | Total | 435 |  |

Tabla 4.1: Listado de tareas del Product Backlog