



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 1

5 de Octubre de 2015

Ingeniería de Software II

Integrante	LU	Correo electrónico
Lambrisca, Santiago	274/10	santiagolambrisca@gmail.com
Mancuso, Emiliano	597/07	emiliano.mancuso@gmail.com
Mataloni, Alejandro	706/07	amataloni@gmail.com
Reartes, Marisol	422/10	mreartes5@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Contents

1	Introducción	3
2	Product Backlog	4
3	Sprint Backlog	5
4	Documentación	8
5	Diseño	9
6	Retrospectiva	13
7	Aclaraciones	14
8	Conclusiones	15

1 Introducción

Con el objetivo de lograr una mayor comprensión de la metodología Agile, utilizando en particular el metodo Scrum, nos dispusimos a realizar el presente trabajo práctico, el cual consiste en realizar una iteración en el proceso de desarrollo de un sistema de puntuación para una aseguradora.

El trabajo requiere pasar por distintas situaciones a enfrentar a la hora de comenzar a desarrollar un proyecto utilizando el método Scrum, ya sean la definicion de las user stories necesarias para obtener el sistema deseado, como la discusion e intercambio de opiniones con el product owner acerca de las mismas. Tambien debimos estimar el esfuerzo necesario y el valor de negocio para cada una de ellas. Definimos las tareas necesarias para completar cada user story, estimando las horas necesarias para cada tarea, y marcando los tiempos trabajados sobre cada una de ellas.

El trabajo termina luego de realizado el primer sprint, con el cual se llega a presentar una demostración del funcionamiento obtenido hasta el momento.

2 Product Backlog

Las user stories del Backlog son:

1. Como dueño quiero que la aplicacion consulte un servicio de mapas para obtener información.
Story points: 2
Criterio de aceptación: Dado un par de coordenadas se prueba si la información obtenida es la correcta.
Business Value: 5
Tareas:
 - Elegir un servicio de mapas para obtener la información necesaria.
Tiempo estimado: 8 horas.
2. Como dueño quiero que la aplicación consulte un servicio de mapas para obtener información de velocidad máxima permitida del lugar transitado
Story points: 5
Criterio de aceptación: Dado un par de coordenadas a las que le corresponda una velocidad máxima de 50 km/h, se consulta al servicio de mapas por la velocidad máxima y se obtiene 50 km/h.
Business Value: 3
Tareas:
 - Conectarse con el servicio de mapas elegido para conocer la velocidad máxima permitida del lugar que se está transitando.
Tiempo estimado: 12 horas.
3. Como dueño quiero que la aplicacion consulte un servicio de mapas para obtener información de Zona insegura del lugar transitado
Story points: 5
Criterio de aceptación:
 - Dado un par de coordenadas correspondientes a una zona segura, se consulta al servicio de mapas y la respuesta es que es zona segura.
 - Dado un par de coordenadas correspondientes a una zona insegura, se consulta al servicio de mapas y la respuesta es que es zona insegura.**Business Value:** 3
Tareas:
 - Conectarse con el servicio de mapas elegido para conocer si la zona transitada es insegura o no.
Tiempo estimado: 12 horas.
4. Como dueño quiero que el celular del asegurado se comunique por bluetooth con el dispositivo del auto para que la aplicacion reconozca que esta en el auto asegurado
Story points: 1
Criterio de aceptación: Si me acerco con el celular al dispositivo colocado en el auto, se establece la conexión y se muestra una notificación de éxito. En caso de que falle se intentará cada 10 segundos.
Business Value: 2
Tareas:
 - Desarrollar el sistema de reconocimiento del vehiculo via bluetooth.
Tiempo estimado: 4 horas.

5. Como dueño quiero que la aplicación pueda conectarse en todo momento al auto del asegurado

Story points: 1

Criterio de aceptación: Cuando se acerca el teléfono al dispositivo bluetooth se establece la conexión directamente, sin que el usuario tenga que abrir la aplicación

Business Value: 3

Tareas:

- Hacer que la aplicación corra en background y se conecte con el dispositivo en el auto asegurado.

Tiempo estimado: 3 horas.

6. Como dueño quiero que la aplicación registre la información de la posición actual y velocidad para guardar los datos que servirán para los cálculos del scoring

Story points: 2

Criterio de aceptación: Una vez establecida la conexión, la aplicación debe registrar cada 10 segundos la información.

Business Value: 5

Tareas:

- Obtener del gps la posición actual y velocidad

Tiempo estimado: 4 horas.

7. Como dueño quiero que la aplicación registre el total de kilómetros recorridos (en 1 mes) para brindar más información al algoritmo que calcula el scoring

Story points: 2

Criterio de aceptación: Dada una lista de mediciones, que representan 10 kilómetros recorridos, la aplicación debe devolver 10 como cantidad de km recorridos, el cual es el valor esperado.

Business Value: 1

Tareas:

- Registrar los kilómetros recorridos en 1 mes.

Tiempo estimado: 4 horas.

8. Como dueño quiero que la aplicación envíe información al servidor una vez al mes para hacer los cálculos del scoring.

Story points: 2

Criterio de aceptación: Se pueden enviar desde la aplicación los datos obtenidos y poder recibirlos desde el servidor.

Business Value: 3

Tareas:

- Enviar la información al servidor una vez al mes.

Tiempo estimado: 4 horas.

3 Sprint Backlog

Las user stories del Sprint son:

1. Como dueño quiero que la aplicación consulte un servicio de mapas ficticio para la demo para obtener información de velocidad máxima permitida del lugar transitado y si la zona es insegura o no.

Story points: 1

Criterio de aceptación: Dado un par de coordenadas se prueba si la información obtenida es la correcta.

Business Value: 5

Tareas:

- Generar un servicio de mapas de prueba para la demo.

Tiempo estimado: 2 horas.

2. Como dueño quiero registrar distintos tipos de eventos con sus costos asociados para sumar puntos al scoring

Story points: 3

Criterio de aceptación: Los eventos solicitados hasta el momento son tenidos en cuenta por la aplicación

Business Value: 8

Tareas:

- Hacer el diseño para soportar distintos tipos de eventos.

Tiempo estimado: 4 horas.

3. Como dueño quiero registrar eventos de cambio brusco

Story points: 5

Criterio de aceptación: Dado un evento que suma 20 puntos de scoring por cambios bruscos en la velocidad:

- Si no se registra ningún cambio brusco, no se sumaran puntos al scoring.
- Si se registra 1 cambio brusco, se sumaran 20 puntos al scoring.
- Si se registran 3 cambios bruscos, se sumaran 60 puntos al scoring.

Business Value: 5

Tareas:

- Agregar evento de cambio brusco de velocidad.

Tiempo estimado: 8 horas.

4. Como dueño quiero registrar eventos de zona insegura

Story points: 3

Criterio de aceptación: Dado un evento que suma 15 puntos de scoring si se transita una zona insegura:

- Si ninguna medición fue registrada en zona insegura, no se sumaran puntos al scoring.
- Si 1 medición fue registrada en zona insegura, se sumaran 15 puntos al scoring.
- Si 3 mediciones fueron registradas en zonas inseguras, se sumaran 45 puntos al scoring.

Business Value: 5

Tareas:

- Agregar evento de zona insegura.

Tiempo estimado: 4 horas.

5. Como dueño quiero registrar eventos de velocidad máxima

Story points: 3

Criterio de aceptación: Si tengo un evento que superando el 10

- Si ninguna medición supera por más del 10
- Si 1 medición supera por más del 10
- Si 3 mediciones superan por más del 10

Business Value: 5

Tareas:

- Agregar eventos de velocidad máxima.
Tiempo estimado: 4 horas.

6. Como dueño quiero poder agregar con facilidad nuevos tipos de eventos para hacer más flexible la aplicación

Story points: 3

Criterio de aceptación: Poder agregar un nuevo tipo evento y se refleje en los cálculos del scoring

Business Value: 3

Tareas:

- Extender el modelo para que soporte nuevos tipos de eventos.
Tiempo estimado: 5 horas.

7. Como dueño quiero que se procesen los datos recibidos para generar los scorings.

Story points: 8

Criterio de aceptación: Dado un conjunto de datos, se calcula el scoring a partir de los mismos.

- Con una medición que exceda el límite de velocidad (con un evento con 10 puntos de scoring), debe devolver un scoring de 10.
- Con 2 mediciones que excedan el límite de velocidad (con un evento con 10 puntos de scoring), debe devolver un scoring de 20.
- Con 1 medición que se encuentre en una zona insegura (con un evento con 30 puntos de scoring), debe devolver un scoring de 30.

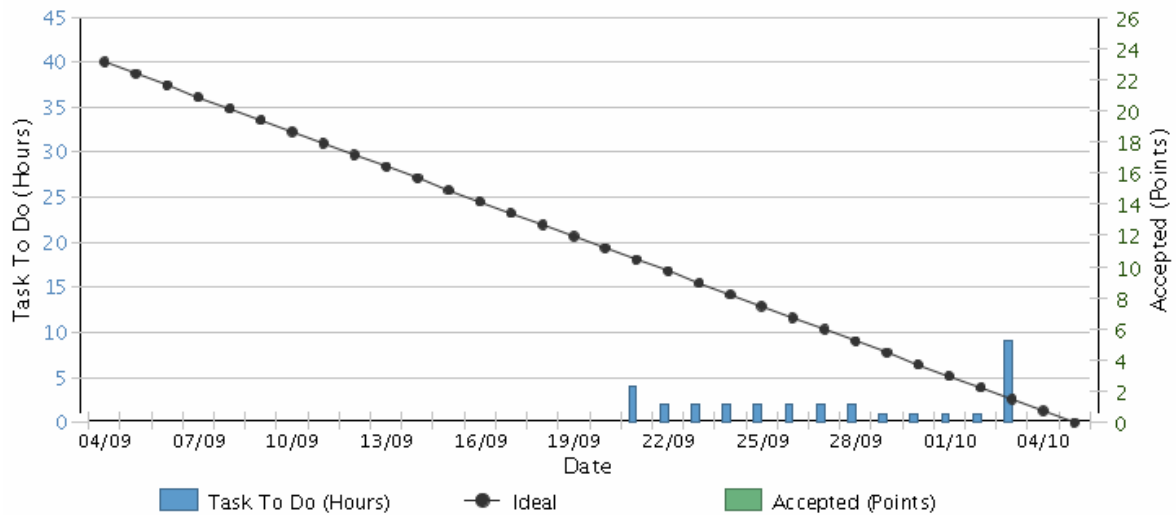
Business Value: 13

Tareas:

- Desarrollar el generador de scoring, que dadas unas mediciones, calcule el scoring correspondiente a las mismas.
Tiempo estimado: 16 horas.

4 Documentación

Este trabajo práctico se realizó siguiendo la metodología ágil Scrum. A continuación, se puede observar el gráfico Burndown correspondiente al Sprint desarrollado para concluir las tareas necesarias para la demo final. El mismo fue obtenido automáticamente a partir de la herramienta utilizada RallyDev.

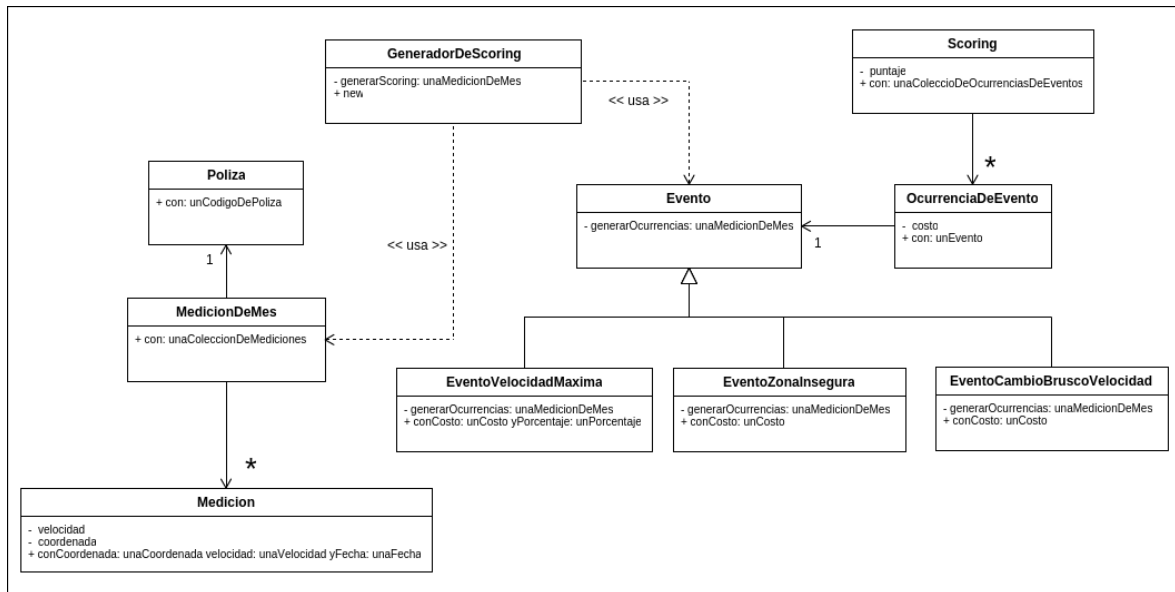


A partir del gráfico, se puede notar que la mayor parte de las tareas fueron progresando desde aproximadamente la mitad del sprint. También se puede observar que sobre el final del sprint se completaron las tareas.

Se puede observar en el gráfico la inconsistencia del desarrollo del proyecto. Esto ocurrió principalmente por haber agregado y eliminado tanto stories como tareas ya estando en la etapa de desarrollo. Se debería haber definido inicialmente, sin lugar a modificaciones, todas las especificaciones por parte del cliente y las tareas que son requeridas para cada una. Luego de la reunión intermedia realizada con el Product Owner se agregaron nuevas tareas y se eliminaron otras, generando así un desorden en la organización.

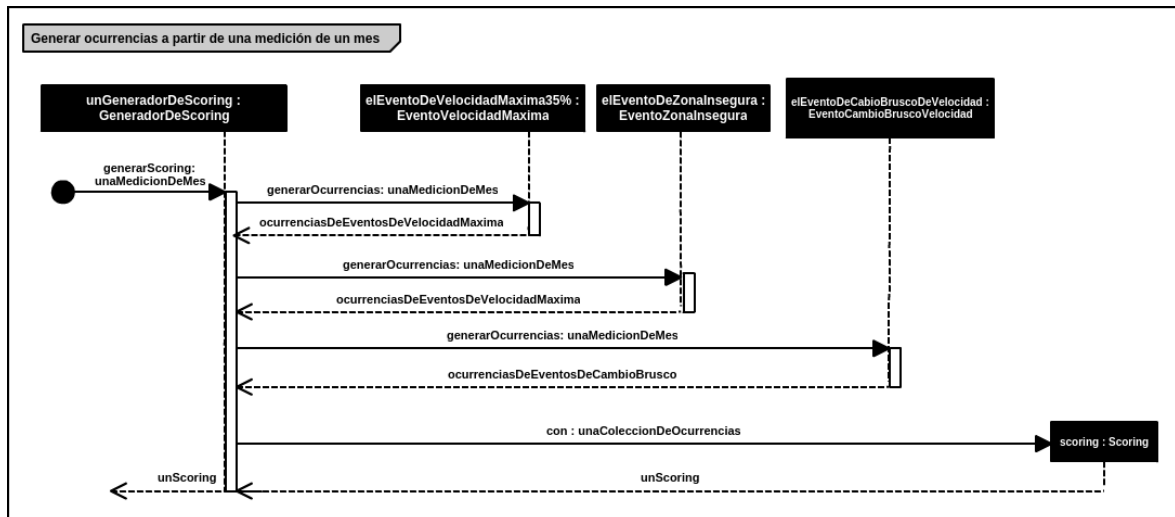
5 Diseño

A continuación se muestra el diagrama de clases correspondiente al sistema generador de scorings.

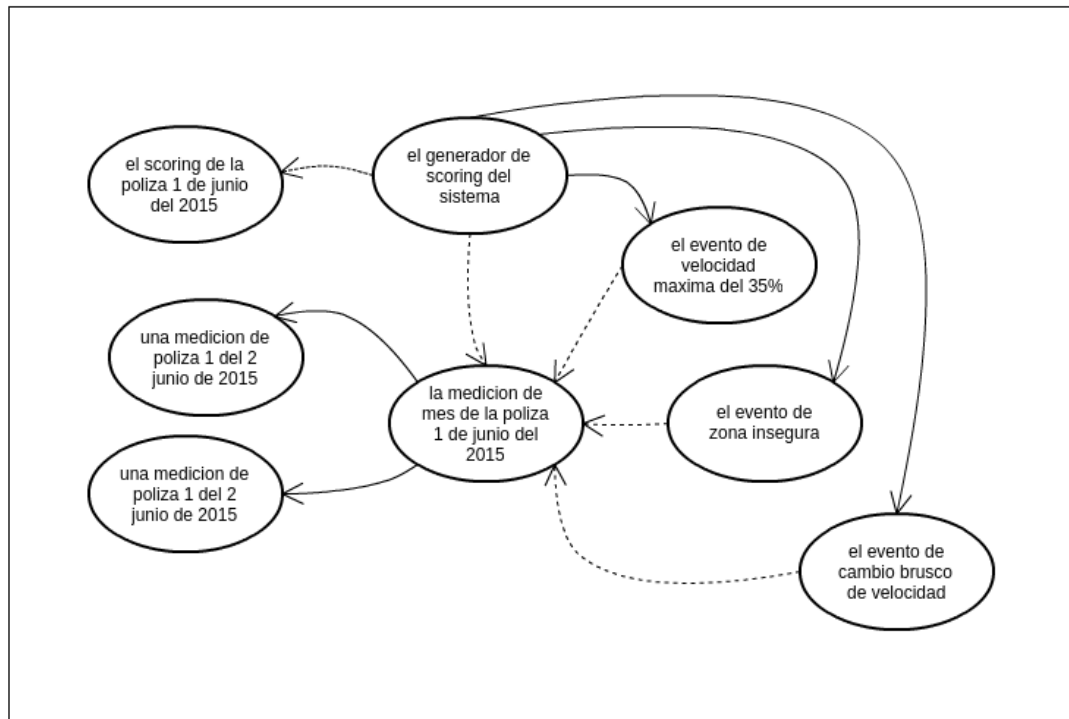


Como se ve en el diagrama, este diseño soporta el registro de distintos tipos de eventos. Se puede observar que en un futuro se podrán agregar nuevos tipos de eventos sin hacer grandes modificaciones al modelo.

Debajo, se muestra el diagrama de secuencias del método *generarScoring*. La idea es mostrar como interactúa el *Generador de Scoring* con los distintos *Eventos* para obtener el conjunto de *Ocurrencias de Eventos*, utilizados para calcular el scoring dada una *Medición de Mes*. Esta última contiene las *Mediciones* obtenidas durante cierto mes.

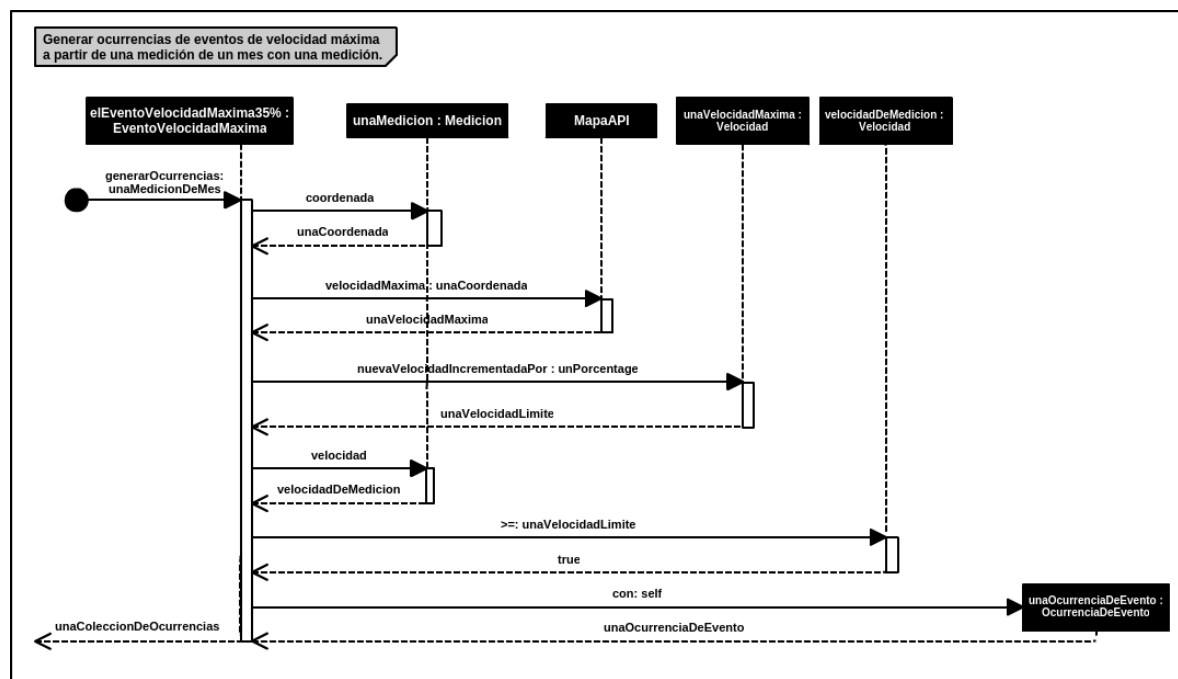


Por otro lado, para complementar la idea explicada anteriormente, se presenta el diagrama de objetos relacionado.



Luego, para entrar más en detalle de cómo se crean las *Ocurrencias de Eventos*, se realizó el diagrama de secuencias para el siguiente escenario:

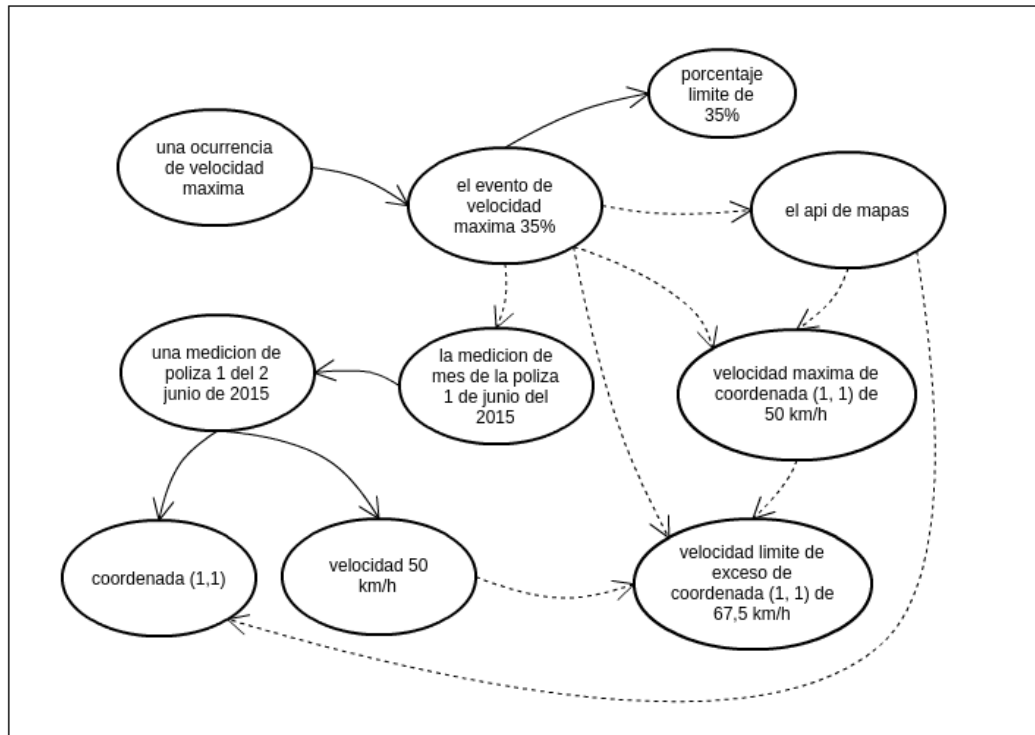
- Dados una medición de mes con sólo una medición, la cual tiene una velocidad 50 km/h y una coordenada en la cual la velocidad máxima es de 30 km/h, y un evento de velocidad máxima con un porcentaje de 10%, se genera una ocurrencia de evento.



Aclaraciones:

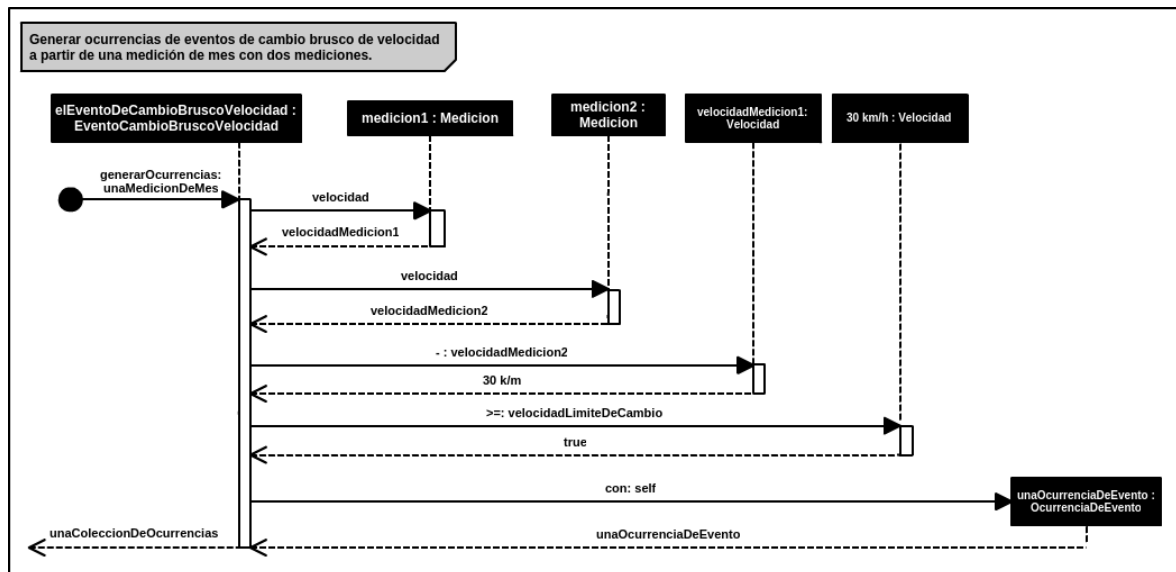
- *unPorcentaje* es el colaborador interno de *elEventoDeVelocidadMaxima35%*.
- En este diagrama, sólo se muestra el intercambio de mensajes con una medición pero esto se repetirá para cada medición del mes.

Para este escenario, también se realizó el diagrama de objetos:



A continuación, se puede observar el diagrama de secuencias del siguiente escenario:

- Dados una medición de mes con dos mediciones, con velocidades 60 km/h y 90 km/h, y un evento de cambio brusco de velocidad con una velocidad límite de cambio de 20 km/h, se genera una ocurrencia de evento.



Aclaraciones:

- La *velocidadLimiteDeCambio* es colaborador interno de *elEventoDeCambioBruscoVelocidad*
- En este diagrama, sólo se muestra el intercambio de mensajes con un par de mediciones pero esto se repetirá para cada par de mediciones del mes.

6 Retrospectiva

Uno de los cambios a realizar para la próxima iteración es una mejor utilización de la herramienta para el seguimiento del sprint ya que muchas veces nos juntábamos y no se actualizaban las user stories o tareas una vez que se progresaba en las mismas o se finalizaban.

Otro cambio sería ir avanzando en el desarrollo gradualmente y no todo sobre el final. En este trabajo no se prosiguió de esa manera debido a los horarios de los integrantes.

7 Aclaraciones

- La aplicación de celular registrará datos de la fecha/hora, posición y velocidad. Esta información se enviará al servidor y este se encargará de realizar los cálculos correspondientes para obtener el scoring.
- El servidor será el encargado de conectarse con la aplicación de mapas para obtener la información de velocidades máximas y zonas inseguras.
- Se asume que el celular del asegurado tiene prendido el bluetooth.
- En el diagrama de clases no se muestran clases como Velocidad y Coordenada, ya que creemos que no son representativos al problema.
- Asumimos que la unidad de la velocidad es siempre km/h.
- La velocidad límite del evento de cambio brusco de velocidad no se define cuando se crea el mismo, sino que ya está predefinida.
- Los eventos que conoce el Generador de Scoring están fijos a nivel código ya que para la demo no nos centramos en ese problema.

8 Conclusiones

La metodología Scrum permite trabajar de manera eficiente sobre el desarrollo de un proyecto. La misma requiere de varias iteraciones y experiencia de quienes la utilizan para aprovecharla.

En este trabajo práctico se hizo una única iteración por lo que las estimaciones de las user stories no fueron las mejores ya que no había parámetros anteriores con los cuales comparar y así asignar story points razonables a las mismas. Debido a la inexperiencia los user stories y tareas fueron cambiando a lo largo del sprint, junto con sus estimaciones, lo cual se refleja en el burndown chart explicado anteriormente. Lamentablemente no pudo ser aprovechada de manera óptima en este proyecto y se cree que es por falta de experiencia en el uso de la misma y, sobre todo, por estar en la etapa inicial.

Si bien no pudo ser utilizada de manera eficiente, creemos que Scrum es una buena metodología para el desarrollo de software.