

1. Introducción

ya
no

[Esta primera etapa del trabajo consistió en la planificación, utilizando la metodología ágil Scrum, del desarrollo de un simulador de partidos de básquet de fantasía.

User Stories



Por las características del sistema pedido, donde hay una simulación -que representa una etapa del desarrollo bastante pesada-, quedaron pocas user stories en total, pues gran parte del sistema se concentra específicamente en ese punto. Esto fue algo que se habló con el product owner y estaba dentro de lo esperable.

Justamente por ser algo que llevaba tanto tiempo, se estuvo en la duda de si convenía o no dividir la **User Story** correspondiente a la simulación. Una solución propuesta fue especificar en las user stories que una simulación se componía de turnos, éstos de jugadas, y éstas de acciones de los jugadores. Una división de ese modo resultó exagerada, y además iría en contra del principio de independencia para las user stories de **Scrum**; dado que la simulación debería entrar completa en un único sprint, por lo tanto, se decidió dejarla como una única **User Story**.

Valuación User Stories

Tanto para la sección de *Business Value* como de *Effort* de cada User Story se decidió realizar poker planning entre los 4 integrantes del grupo. Cuando había discrepancias, se esgrimían los argumentos por los que cada uno había puesto el puntaje correspondiente, de manera de intentar convencer a los otros y así converger los criterios.

Roles

Otro punto donde hubo dudas fue en cuanto a los roles. A primera vista, no parecería haber nadie más que participe del sistema más que el usuario final, a quien llamamos un *participante*.

Por cómo se encontraban redactados algunos puntos específicos del enunciado, dejando algunas cosas abiertas con la posibilidad de que sufran modificaciones a futuro, nos pareció propicio considerar un rol de alguien que se encarga de "mantener" y administrar el sistema -que seguramente no sea el dueño, aunque sí puede que esté dirigido por el mismo-. Ese sería el rol del *administrador*.

- **Participante:** es quien se encarga de crear equipos, desafiar a otros participantes y participar de las simulaciones. El *usuario final* del sistema.
- **Administrador:** es aquel que actualiza los datos de los jugadores, define las jugadas de cada técnico y las configuraciones de la simulación, tales como la cantidad de turnos de cada una. Es un supervisor del sistema, quien lo regula, el que realiza las acciones para hacerlo más atractivo para los participantes, y más equilibrado.

1.1. Sprint

La duración del Sprint se decidió que sea de alrededor poco más de 3 semanas, es decir la totalidad del tiempo asignado, y la cantidad de horas hombre que irían en ella de 62 horas. Se llegó a este total dada una estimación de 5 horas semanales por integrante para el desarrollo, teniendo en cuenta horas que se desperdician no cumpliendo el desarrollo principal.

(X) ver lo anterior

2. Planificación

2. Planificación

2.1. Product Backlog

En la siguiente tabla se encuentran todas las user stories definidas para el proyecto.

ID	Descripción	Business Value	Effort
US 1	COMO participante QUIERO tener una cuenta PARA tener asociada mi información	8	3
US 2	COMO participante QUIERO armar un equipo PARA competir contra otros	13	8
US 3	COMO participante QUIERO tener una lista de mis equipos ya armados PARA ahorrar tiempo	1	3
US 4	COMO administrador QUIERO definir cuáles son los jugadores PARA que los participantes armen sus equipos	8	5
US 5	COMO administrador QUIERO poder actualizar las estadísticas y datos de los jugadores PARA ser fiel a la realidad	5	5
US 6	COMO administrador QUIERO poder actualizar los jugadores con datos reales utilizando algún servicio externo PARA que sea automático y no manual	5	13
US 7	COMO participante QUIERO conocer el libro de jugadas del técnico PARA saber cómo va a dirigir	5	3
US 8	COMO administrador QUIERO definir cuáles son los técnicos disponibles PARA que los participantes armen sus equipos	5	3
US 9	COMO administrador QUIERO poder definir las jugadas disponibles de los técnicos PARA enriquecer la simulación	5	5
US 10	COMO participante QUIERO poder crear y aceptar desafíos PARA medirme con otros participantes	13	5
US 11	COMO participante QUIERO apostar fichas PARA subir posiciones en la tabla	8	2
US 12	COMO administrador QUIERO que los participantes puedan simular partidas PARA que jueguen entre sí	21	21
US 13	COMO administrador QUIERO poder ajustar la duración (en turnos) de las simulaciones PARA que el sistema sea flexible	3	3
US 14	COMO administrador QUIERO poder modificar las fórmulas de resolución de acciones PARA ir ajustando el sistema a lo largo del tiempo	8	8
US 15	COMO administrador QUIERO que el primer turno de cada simulación sea al azar PARA hacerlo justo	2	1
US 16	COMO administrador QUIERO que quede un log y toda la información pertinente de cada simulación PARA que cualquier participante pueda consultarla	8	5
US 17	COMO administrador QUIERO que se repartan las fichas adecuadamente después de terminado el partido PARA calcular la nueva tabla de posiciones	13	1
US 18	COMO participante QUIERO ver la tabla de posiciones PARA compararme con los otros participantes	5	5
US 19	COMO administrador QUIERO que el presupuesto de cada equipo no supere el cap del participante PARA equilibrar los valores de los equipos	1	1

Los criterios de aceptación de las user stories que entran en el Sprint (definido en la sección 2.2), se encuentran a continuación.

US11

US11.TA1

Descripción: Discusión y modelado sobre el mecanismo de apuestas. Decidir cuestiones implementativas.

Duración estimada: 1.0hs

US11.TA2

Descripción: Implementación del mecanismo de apuesta de fichas

Duración estimada: 1.0hs

US11.TA3

Descripción: Realizar casos de pruebas con valores de apuestas válidos e inválidos. Verificar que la apuesta quede asociada al participante y simulación correctos.

Duración estimada: 1.0hs

US1

US1.TA1

Descripción: Discutir distintos tipos posibles de cuentas de usuario y definir el modelo que representará a los participantes en el sistema. Determinar la interfaz que permitirá ingresar datos.

Duración estimada: 1.0hs

US1.TA2

Descripción: Implementación de la abstracción que representa al usuario y de la pantalla de introducción de datos.

Duración estimada: 1.0hs

US1.TA3

Descripción: Implementación del mecanismo de log-in del sistema.

Duración estimada: 1.0hs

US1.TA4

Descripción: Creación y ejecución de casos de prueba tanto para registro como para login. Verificar que no se puedan crear usuarios inválidos.

Duración estimada: 1.0hs

Mucho mejor aunq' todavía desperdicia espacio

US19

US19.TA1

Descripción: Utilizar la inecuación (sumatoriaValores < capParticipante), definir los mensajes de error y cómo mostrárselos al usuario.

Duración estimada: 0.5hs

US19.TA2

Descripción: Asignarle los valores a la inecuación cada vez que estén los 5 jugadores elegidos.

Duración estimada: 1.0hs

US19.TA3

Descripción: Probar todos los casos de la inecuación (casos: menor, mayor, igual) y realizar una selección de jugadores que de como resultado cada uno de ellos. Verificar que los mensajes de error sean correctos, y que no se deje formar un equipo que no cumpla la condición.

Duración estimada: 1.0hs

US12

US12.TA1

Descripción: Involucra el análisis y el modelado de los distintos aspectos de la simulación, así como de las diferentes colaboraciones que se llevan a cabo entre los componentes del sistema para que la simulación se realice correctamente.

Duración estimada: 10 hs

US12.TA2

Descripción: Supone la implementación del diseño de la simulación.

Duración estimada: 10 hs

US12.TA3

Descripción: Definir y ejecutar casos de prueba que permitan verificar el correcto funcionamiento de cada una de las componentes de la simulación, tales como la resolución de jugadas, cantidad de turnos de desempate, equipos que participan y el uso correcto de sus estadísticas, y que el resultado de la simulación sea el deducido de los resultados de cada jugada.

Duración estimada: 5hs



3. Diseño

[proyecto?] A continuación se presentan y explican algunas de las decisiones de diseño elegidas durante el desarrollo del trabajo práctico. Se decidió dividir la presentación en subsecciones para facilitar la comprensión de la misma, haciendo especial énfasis en aquellas que sentíamos eran de importancia para la comprensión del mismo. [repiten muchas veces]

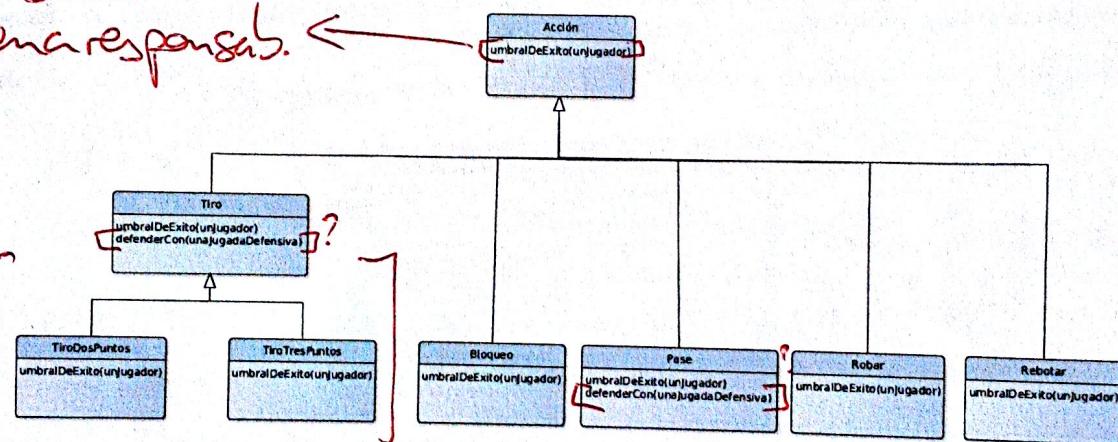
Vale mencionar que si bien es cierto que se generó una idea general en el grupo antes de comenzar a reflejar las decisiones tomadas en código, el diseño en profundidad se terminó de desarrollar y afinar en paralelo, mientras surgían cuestiones mismas relacionadas al propio desarrollo que no habían sido tenidos en cuenta en papel.

3.1. Acciones

Estaría bien si empieza contando algo del proyecto, an vez de arrancar los detalles: enfoque top-down

Entre las decisiones de diseño que consideramos de mayor relevancia, se encuentra el modelado de las jugadas como una sucesión de acciones que ocurren hasta que la pelota se vaya fuera del campo de juego. A su vez, [modelados] cada una de las distintas posibilidades de acción mediante una jerarquía. El mayor beneficio de esta decisión es el de poder modificar los umbrales de éxito de cada una de las acciones de manera sencilla y global para cualquier jugada, afectando lo mínimo indispensable al resto del modelo del sistema. [modelamos?]

No es una buena respuesta.



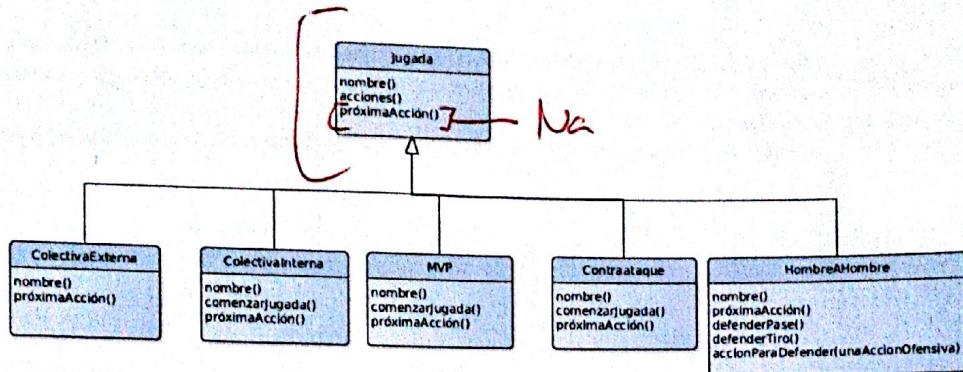
3.2. Jugadas

Como dijimos anteriormente, se decidió modelar a las jugadas como una lista de acciones a ser ejecutadas. Es por esto que el concepto abstracto de jugada requiere poder conocer su lista de acciones y su próxima acción a ejecutar. Una vez más, esta jerarquización, a demás de dejarnos representar el dominio de nuestro problema de forma correcta (MVP, ColectivaExterna, ColectivaInternacional no son más que distintas versiones de la idea de jugada)

Abstracto o concreto?

¿Qué?

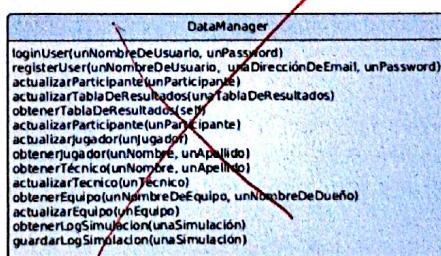
(X) hay poca cohesión



3.3. DataManager Mal nombre (X) no mala idea

Esta es, quizás, la decisión más conflictiva que tuvo el diseño planteado. Mientras una mitad del grupo sostenía que las distintas acciones que componen al DataManager presentado debían estar separadas en distintas clases, la otra se posicionaba firme en la idea de concentrar todo dentro de un 'Manager' simplificando el diseño de una forma que no impactaba en detrimento de la calidad. Debido a un problema de tiempo del grupo, la opción 'simplificadora' terminó ganando impulso y se impuso. Esta clase representa entonces todo el manejo de información que se realiza por fuera del juego propiamente dicho (actualizar jugadores, técnicos, etc). En particular nosotros jerarquizamos esta idea abstracta con una clase de manejo de datos por archivos (FileDataManager) porque es la que usamos para nuestra implementación, pero gracias a esta jerarquía nada impediría poder extender el diseño con, por ejemplo, un DBDataManager que implemente la carga de la información mediante una base de datos.

les habría advertido q' venían tomando un enfoque



Este se rompe
constantemente
y q' tiene alto acoplamiento

& muy baja cohesión;
lo contrario de lo
q' se busca

demasiado separado
hacia la persistencia
(& xtanto muy limitado)

3.4. Nuevo técnico

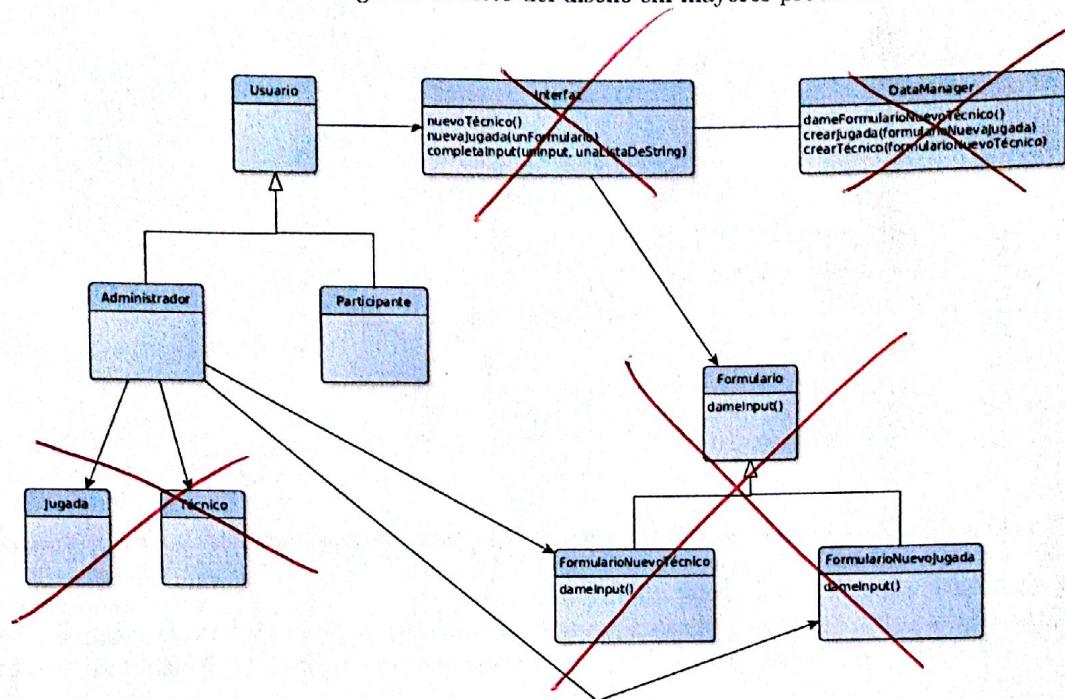
A continuación, se ofrece el caso particular del alta de un nuevo técnico, no porque sea una acción que consideremos relevante, sino como una forma de introducir el contexto en el que nuestro diseño se compone con el 'exterior'. Planteamos entonces una jerarquización entre los usuarios, dando la



15 ¿Porq' habla de "jerarquización"
q' vez q' usa herencia?



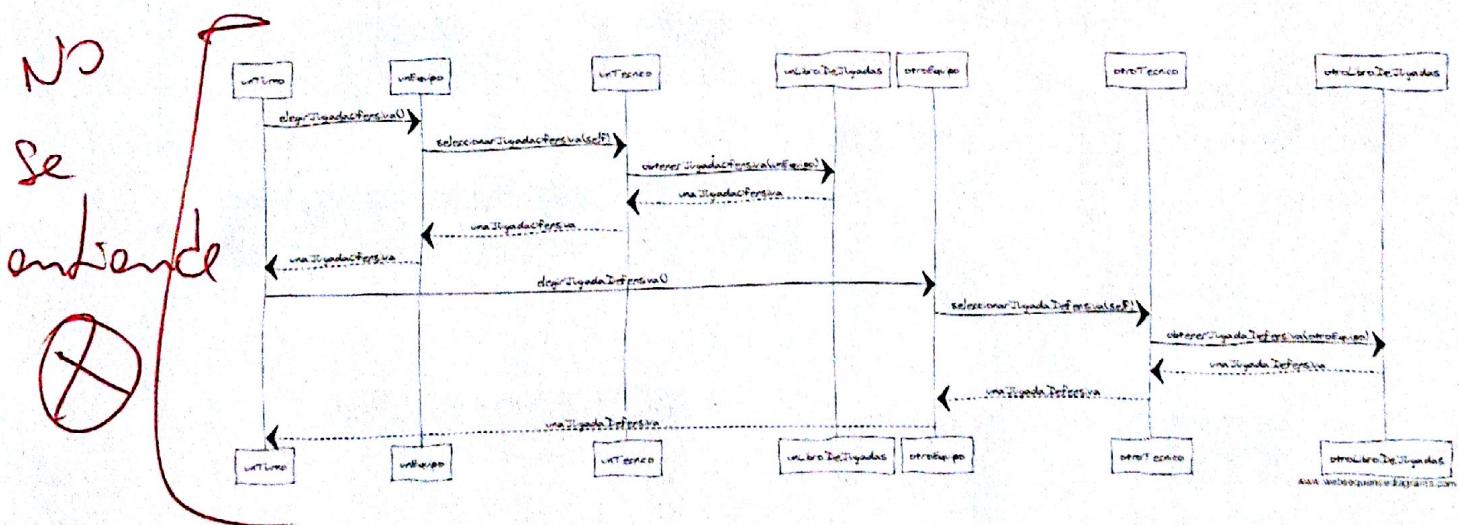
posibilidad a estos de ser tanto participantes del juego como administradores del mismo. Además se provee de un grupo de interfaces encargadas de 'representar' el accionar de cada uno de los formularios. Por consiguiente, la implementación de las distintas interfaces por los formularios, permiten modelar el comportamiento correcto e integrarlo al resto del diseño sin mayores problemas.



3.5. Secuencia de turnos

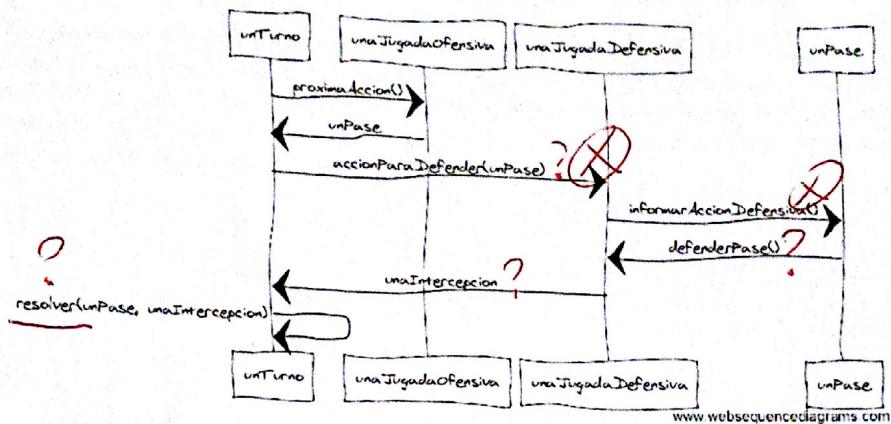
objetos

En este apartado analizaremos el intercambio de mensajes entre nuestras entidades con el fin de clarificar una de las situaciones más repetidas dentro de cualquier simulación: La selección de jugadas. Para ello, presentamos el siguiente diagrama de secuencias en donde ambos equipos necesitan obtener sus respectivas jugadas (primero quien ataca y luego quien esta a la defensiva).



Como podemos observar, al comenzar el turno se le pide al equipo que debe realizar la jugada ofensiva que elija una de su libro de jugadas. Como no es el equipo quien conoce sus jugadas, sino más bien el técnico, se le informa a este quien es el encargado de obtener una de su libro de jugadas y darla a conocer a los demás. Cuando el turno en ejecución conoce la jugada que el equipo ofensivo tiene

pensado realizar, le informa al otro equipo, quien a su vez realiza la misma operación, pero esta vez para obtener una jugada defensiva. Una vez que ambas jugadas estan definidas, es el turno el encargado de manejar la selección de las diferentes acciones a realizar, como veremos a continuación:

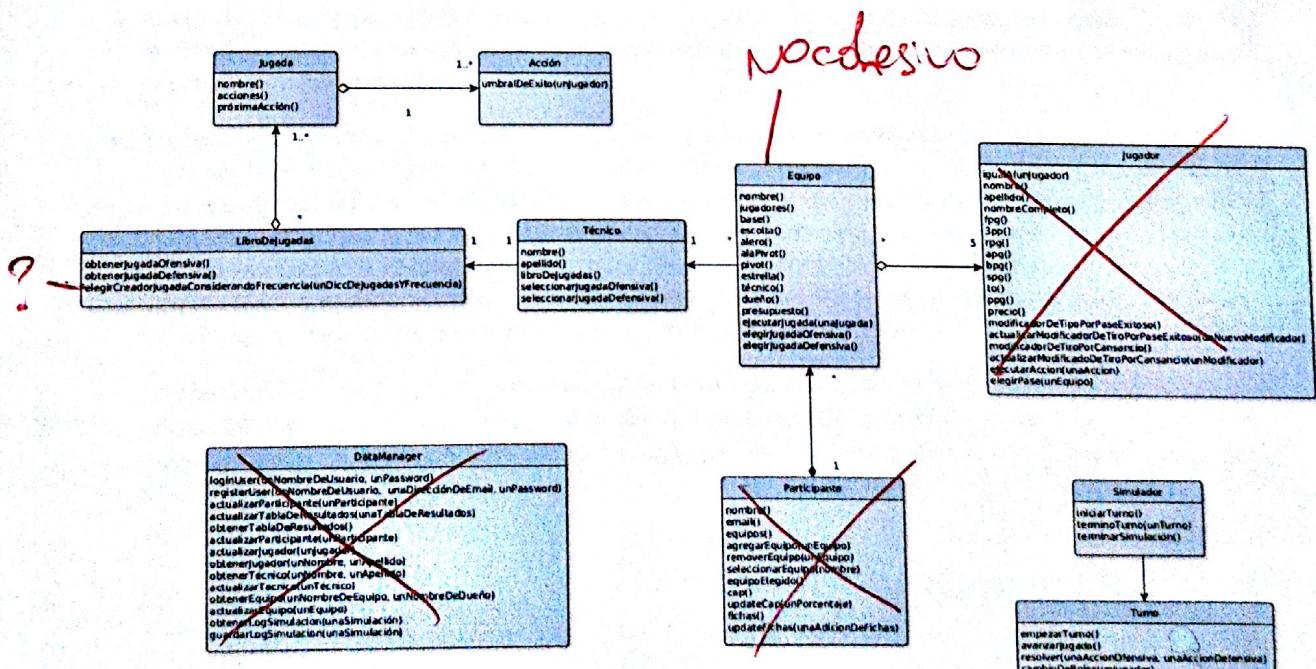


Como bien dijimos anteriormente, es el turno quien pide las respectivas acciones a las jugadas provistas en el paso anterior. En este caso, la jugada ofensiva consta de realizar un pase, por lo que se le informa a la defensiva para tomar una acción acorde a esta decisión. Esta pide la acción a realizar para defender un pase y cuando la obtiene se la notifica también al turno quien será ahora el encargado de resolver (mediante las ecuaciones provistas o por cualquier otro método desarrollado a posterioridad).

3.6. Vision general

[Resumen al ppdo]

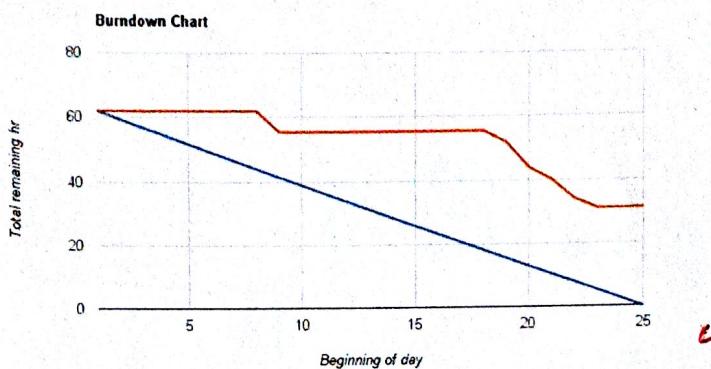
El objetivo de este apartado es el de presentar un panóptico general de la interrelación de algunas de las clases más importantes del diseño. Si bien algunas ya fueron mencionadas con anterioridad (jugadas, acciones, etc), decidimos repetirlas en pos de poder ofrecer una idea, a grandes rasgos, de su integración dentro del sistema.



4. Sprint Retrospective

El estado del TP entregado no es lo esperado ni lo solicitado, por lo que se torna difícil realizar conclusiones de cierre sobre algo que no cumple con esa condición, estar terminado.

No se pudieron apreciar por completo las bondades ni dificultades de la metodología scrum porque decir que se siguió a rajatabla (dentro de la flexibilidad que ofrece) la metodología es una mentira. La dinámica del grupo no fue la esperada, con algunos desencuentros entre los integrantes, lo que dificultó considerablemente la tarea de llevar a cabo un desarrollo exitoso. Esta situación se refleja claramente en el resultado final de nuestro burndown chart:



Algunos comentarios concernientes a la metodología que se podrían hacer son los siguientes:

- Se sobreestimó la cantidad de horas que podría dedicarle cada integrante al grupo; si bien creímos que pusimos un número bastante conservador, ni siquiera se llegó a cumplir esa dedicación para el Trabajo Práctico.
- Se subestimó la cantidad de horas para algunas tareas. No hay tareas que puedan llevar 0.5hrs, y muy pocas tareas que puedan llevar 1 hora. ✓
- Entre los dos items anteriores está claro cuán alejada estuvo finalmente la estimación de la realidad, y además, que en ambos puntos nos equivocamos en direcciones opuestas, aumentando aún más la famosa brecha.
- En base a lo vivido en el sprint, Scrum parecía tener más sentido para un grupo reducido de personas que está cotidianamente en el mismo lugar, dedicándole una cantidad de horas similar a las tareas, y con una comunicación fluida y constante. La flexibilidad es una ventaja cuando se es dinámico y se tiene una adaptabilidad rápida a las distintas situaciones y problemáticas que van surgiendo a lo largo del desarrollo. De ser esta conjectura cierta, el grupo no respetó bien ninguno de los puntos, y la falta de dinamismo e ida y vuelta ocasionó que no se pudieran tomar las decisiones necesarias en el momento indicado, retrasando todo.
- Es bastante difícil entrar en contacto con una metodología completamente desconocida para los integrantes en apenas unas semanas, y en el marco de una materia, donde además hay otras responsabilidades (relacionadas con la materia, con la Facultad, y las peores de todas, las ajenas a todo ello).