



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA



Departamento de Computación,
Facultad de Ciencias Exactas y Naturales,
Universidad de Buenos Aires

Trabajo Práctico 1

Ingeniería de Software II

Primer Cuatrimestre de 2016

Apellido y Nombre	LU	E-mail
Almansi, Emilio Guido	674/12	ealmansi@gmail.com
Gasperi Jabalera, Fernando	56/09	fgasperijabalera@gmail.com
Christian “El Capo del Sorting” Russo	679/10	christian.russo8@gmail.com
Tagliavini Ponce, Guido	783/11	guido.tag@gmail.com

Índice

1. Introducción	3
1.1. Roles de usuario	3
1.2. Observaciones	3
2. Product Backlog	3
2.1. Spikes	3
2.2. Product backlog	4
2.2.1. Turnos	4
2.2.2. Sobre un Jugador	4
2.2.3. Puntos	4
2.2.4. Rebotes y pelotas divididas	4
2.2.5. Pases y robos de pelota	5
2.2.6. Sobre el equipo (DT y libro de jugadas)	5
2.2.7. Jugadas	5
2.2.8. Fórmulas de resolución de acciones	6
2.2.9. Otras	6
3. Sprint backlog	7
4. Diseño	28
4.1. Descripción informal	28
4.2. Diagramas de objeto	28
4.2.1. Simulador de Partido	28
4.2.2. Simulador de Turno	29
4.2.3. Simulador de Jugada	31
4.2.4. Simulador de Momento	32
4.3. Diagramas de secuencia	33
4.4. Diagramas de clase	33
5. Burndown	35
6. Conclusión	36

1. Introducción

El actual informe presenta el análisis y desarrollo inicial realizado para el producto **The Curry Game**. En el mismo se detallan el conjunto de user stories que abarcan el desarrollo completo de la aplicación, incluyendo tanto las definiciones actuales como los puntos de extensión que se tendrán en cuenta. El objetivo de la aplicación es que los usuarios puedan armar sus equipos de basket, pudiendo participar de desafíos entre los usuarios con el fin de ganar mas fichas que le permiten armar un mejor y más robusto equipo. El sistema cuenta con una simulación del juego, que provee un log para poder ver cómo se desarrolla el juego.

1.1. Roles de usuario

Por brevedad, se utilizan los siguientes roles en las User Stories a lo largo de este trabajo:

Jugador: cualquier usuario del sistema.

Participante: un jugador que postula o acepta un desafío.

Game designer: el encargado de determinar los detalles de la simulación.

1.2. Observaciones

La herramienta de planificación ágil elegida por el equipo es **TargetProcess**. Esta herramienta utiliza un sistema de ranking para el Business Value muy particular, que nosotros traducimos en valores entero del 1 al 4 del siguiente modo:

- Must Have: 4
- Great: 3
- Average: 2
- Nice To Have: 1

Utilizamos Spikes para mostrar las tareas preliminares. Por otro lado, luego de hacer un análisis del sistema se decidió que el lenguaje de programación que utilizaremos será Python, que es de fácil utilización para programación de propósito general (como programar el simulador en cuestión) y provee una sencilla API para realizar solicitudes HTTP.

2. Product Backlog

En esta sección se listan todas las User Stories del Backlog de nuestro sistema.

2.1. Spikes

1. Investigar sobre qué lenguaje es el más adecuado para implementar el sistema. (**Horas Hombres: 2**)
2. Investigar la API de la NBA para evaluar la dificultad de implementar la lectura automática de datos de jugadores. (**Horas Hombres: 2**)
3. Investigar la API de Twitter para evaluar la dificultad de implementar la medición de popularidad de un jugador en Twitter. (**Horas Hombres: 1**)
4. Instalar un entorno de desarrollo (**Horas Hombres: 2**)
5. Instalar un entorno de testing (**Horas Hombres: 3**)

2.2. Product backlog

2.2.1. Turnos

1. Como game designer quiero que el juego tenga 40 turnos.
SP: 1, BV: 4
2. Como game designer quiero que al empezar cada turno, un equipo comience atacando y el otro defendiendo.
SP: 1, BV: 4
3. Como game designer quiero que al empezar cada turno, el equipo que comienza atacando se vaya alternando entre ambos equipos.
SP: 2, BV: 4
4. Como game designer quiero que al empezar el primer turno, el equipo atacante sea elegido al azar.
SP: 2, BV: 4
5. Como game designer quiero que los turnos acaben si la pelota se va de la cancha o si se realiza un tiro exitoso al aro.
SP: 3, BV: 4
6. Como game designer quiero que, en caso de que se agoten los turnos y ambos equipos tengan el mismo puntaje, se jueguen 6 turnos extra de prórroga.
SP: 3, BV: 4

2.2.2. Sobre un Jugador

7. Como game designer quiero que un jugador en ataque pueda lanzar la pelota al aro (por 2 o 3 puntos) o intentar un pase, y sólo eso.
SP: 3, BV: 4
8. Como game designer quiero que un jugador en defensa pueda intentar bloquear tiros al aro de otro jugador.
SP: 3, BV: 4
9. Como game designer quiero que un jugador en defensa pueda interceptar pases.
SP: 8, BV: 4

2.2.3. Puntos

10. Como game designer quiero que cuando un jugador realice un tiro al aro con éxito, su equipo sume 2 o 3 puntos según corresponda.
SP: 2, BV: 4

2.2.4. Rebotes y pelotas divididas

11. Como game designer quiero que cuando un tiro al aro resulte fallido o bloqueado, la pelota quede dividida y ambos equipos tengan la posibilidad de recuperarla.
SP: 2, BV: 4
12. Como game designer quiero que cuando una pelota quede dividida, la misma se dispute en forma intercalada entre los jugadores del equipo defensivo y del equipo ofensivo, en orden decreciente según número de posición.
SP: 5, BV: 4
13. Como game designer quiero que cuando un jugador defensor gane una pelota dividida, se inviertan los roles de atacante y defensor entre ambos equipos.
SP: 3, BV: 4

14. Como game designer quiero que cuando un jugador atacante gane una pelota dividida, continúe atacando su equipo.
SP: 3, BV: 4
15. Como game designer quiero que cuando ante una pelota dividida ninguno de los 10 jugadores gana el rebote, entonces la pelota salga de la cancha.
SP: 3, BV: 4

2.2.5. Pases y robos de pelota

16. Como game designer quiero que cuando un jugador realice un pase de pelota exitoso y sin intercepción, aumente en un pequeño porcentaje la chance de éxito de tiro al aro del jugador que realiza el tiro posteriormente en la jugada.
SP: 1, BV: 4
17. Como game designer quiero que cuando un jugador defensor tenga éxito interceptando un pase, se cambien los roles de atacante y defensor entre los equipos y continúe como atacante el equipo del defensor que robó la pelota.
SP: 3, BV: 4
18. Como game designer quiero que cuando un pase sea fallido y además no sea interceptado por un defensor contrario, la pelota salga de la cancha.
SP: 5, BV: 4

2.2.6. Sobre el equipo (DT y libro de jugadas)

19. Como game designer quiero que el técnico tenga un libro de jugadas para jugar adecuadamente.
SP: 3, BV: 4
20. Como game designer quiero que el técnico tenga una frecuencia para cada jugada del libro para poder ejecutarlas como si él lo hiciera.
SP: 2, BV: 4
21. Como game designer quiero que al comenzar el ataque de un equipo (ya sea al iniciar un turno, recuperar una pelota dividida, o interceptar un pase), se elija una jugada ofensiva del libro de jugadas del director técnico del equipo atacante, teniendo en cuenta la frecuencia asignada a las mismas.
SP: 2, BV: 4
22. Como game designer quiero que cuando el equipo atacante elija una jugada ofensiva, el equipo defensor escoja una jugada defensiva de su libro de jugadas, teniendo en cuenta la frecuencia asignada a las mismas.
SP: 2, BV: 4

2.2.7. Jugadas

23. Como game designer quiero que se pueda usar la jugada ofensiva colectiva externa.
SP: 3, BV: 4
24. Como game designer quiero que se pueda usar la jugada ofensiva colectiva interna.
SP: 3, BV: 4
25. Como game designer quiero que se pueda usar la jugada ofensiva MVP.
SP: 3, BV: 4
26. Como game designer quiero que se pueda usar la jugada ofensiva Contra Ataque.
SP: 3, BV: 4
27. Como game designer quiero que se pueda usar la jugada defensiva Hombre a Hombre.
SP: 3, BV: 4

- 28. Como game designer quiero que se pueda usar la Dos Cada.
SP: 5, BV: 4
- 29. Como game designer quiero que se pueda usar la FiboJugada.
SP: 5, BV: 4

2.2.8. Fórmulas de resolución de acciones

- 30. Como game designer quiero que cada UE de una acción X sea la probabilidad de éxito al realizar la acción X.
SP: 3, BV: 4
- 31. Como game designer quiero que el Umbral de Exito (UE) de tiro por 2 puntos sea FG para hacer el juego más realista.
SP: 2, BV: 4
- 32. Como game designer quiero que el Umbral de Exito (UE) de tiro por 3 puntos sea 3P para hacer el juego más realista.
SP: 2, BV: 4
- 33. Como game designer quiero que el Umbral de Exito (UE) de bloqueo sea $BPG * 0.2$ para hacer el juego más realista.
SP: 2, BV: 4
- 34. Como game designer quiero que el Umbral de Exito (UE) de pase de pelota sea $1 - TO * 0.1$ para hacer el juego más realista.
SP: 2, BV: 4
- 35. Como game designer quiero que el Umbral de Exito (UE) de robo de pelota sea $SPG * 0.2$ para hacer el juego más realista.
SP: 2, BV: 4
- 36. Como game designer quiero que el Umbral de Exito (UE) de reboteo de pelota sea $RPG * 0.05$ para hacer el juego más realista.
SP: 2, BV: 4
- 37. Como game designer quiero tener las estadísticas originales de los jugadores para que la simulación sea más realista.
SP: 13, BV: 4
- 38. Como game designer quiero que la popularidad en Twitter de los jugadores para afecte mejorar su performance.
SP: 13, BV: 4

2.2.9. Otras

- 39. Como jugador quiero armar un equipo para jugar en un desafío.
SP: 5, BV: 3
- 40. Como jugador quiero aceptar un desafío para ganar ranking, fichas y cap.
SP: 8, BV: 3
- 41. Como jugador quiero postular un desafío para ganar ranking, fichas y cap.
SP: 8, BV: 3
- 42. Como participante de desafío quiero en caso de ganar quiero que incremente mi número de fichas para tener más fichas que otros jugadores y así estar mejor posicionado en el ranking.
SP: 1, BV: 3
- 43. Como participante de desafío quiero en caso de ganar quiero que incremente mi cap para poder formar mejores equipos.
SP: 1, BV: 3

44. Como participante de desafío quiero apostar fichas.
SP: 3, BV: 3
45. Como participante de desafío quiero que me resten las fichas que aposté.
SP: 2, BV: 3
46. Como participante de desafío quiero elegir mi equipo para jugar con los jugadores que considere más adecuados.
SP: 5, BV: 3
47. Como participante de desafío quiero elegir mi DT para ejecutar las jugadas deseadas.
SP: 3, BV: 3
48. Como jugador quiero registrarme con mis datos para darme de alta en el sistema.
SP: 8, BV: 3
49. Como jugador quiero acceder a una tabla de rankings en base a cantidad de desafíos ganados para informarme.
SP: 8, BV: 2
50. Como jugador quiero acceder a una tabla de rankings en base a cantidad de fichas total para informarme.
SP: 8, BV: 2
51. Como jugador quiero poder guardar cada equipo que arme para usarlo luego en un desafío.
SP: 5, BV: 2
52. Como jugador quiero autenticar mi email para activar mi cuenta.
SP: 5, BV: 2
53. Como jugador quiero loguearme con mi email (o username) para ingresar al sistema.
SP: 8, BV: 2
54. Como jugador quiero obtener una cantidad inicial de fichas para apuestas para poder apostar.
SP: 1, BV: 2
55. Como jugador quiero obtener una cantidad de presupuesto (cap) inicial para poder armar mi equipo.
SP: 1, BV: 2

3. Sprint backlog

En esta sección listamos un subconjunto de las User Stories del Backlog que decidimos meter dentro del Sprint teniendo en cuenta el Business Value. Para cada User Story detallamos descripción, tareas, criterios de aceptación, etc.

US1: *Como game designer quiero que el juego tenga 40 turnos.*

Descripción: Cada juego tendrá un total de 40 turnos.

Criterios de aceptación:

1. La cantidad de turnos que se muestran en el log es igual a 40.

Story Points: 1.0

Tarea1A: *Setear la cantidad inicial de iteraciones a 40.*

Horas hombre: 0.25

Tarea1B: *Testear que la cantidad de iteraciones este seteada en 40.*

Horas hombre: 0.25

US2: *Como game designer quiero que al empezar cada turno, un equipo comience atacando y el otro defendiendo.*

Descripción: En cada turno uno de los equipos comienza atacando y el otro defendiendo.

Criterios de aceptación:

1. En el log se notará que al comienzo de un turno con dos equipos, uno de los equipos es atacante y el otro es defensor.

Story Points: 1.0

Tarea2A: *Diseñar un modelo donde se cuente con el equipo atacante y el equipo defensor*

Horas hombre: 1.0

Tarea2B: *Al comenzar un turno setear el equipo atacante y el defensor.*

Horas hombre: 0.5

Tarea2C: *Testear que en cada turno existe un equipo atacante y otro equipo defensor.*

Horas hombre: 0.25

US3: *Como game designer quiero que al empezar cada turno, el equipo que comienza atacando se vaya alternando entre ambos equipos.*

Descripción: Luego de cada turno el equipo que comenzó atacando en el turno anterior empezará defendiendo en el próximo.

Criterios de aceptación:

1. En el log se notará que en dos turnos consecutivos el equipo atacante no es el mismo.
2. En el log se notará que si en un turno un equipo atacó, en el próximo aparece defendiendo.

Story Points: 2.0

Tarea3A: *Guardar el equipo que comienza atacando antes de comenzar un turno.*

Horas hombre: 0.25

Tarea3B: *Cuando termina un turno se setea al equipo que empezará atacando en el próximo turno, como el equipo contrario al que empezó en el anterior.*

Horas hombre: 0.5

Tarea3C: *Diseñar un modelo en el cual el jugador atacante se valla alternando en cada turno.*

Horas hombre: 0.5

Tarea3D: *Testear que cuando pasa un turno el jugador atacante sea el otro equipo*

Horas hombre: 0.25

US4: *Como game designer quiero que al empezar el primer turno, el equipo atacante sea elegido al azar.*

Descripción: Se elegirá un equipo al azar entre los dos equipos que participan y este comienza atacando en este turno.

Criterios de aceptación:

1. Al comienzo del log, es decir al momento del primer turno, aparecerá cualquiera de los dos equipos como atacante.

Story Points: 2.0

Tarea4A: *Al comenzar el primer turno setear el equipo que ataca en él de forma aleatoria.*

Horas hombre: 0.25

Tarea4A: *Diseñar un modelo donde se pueda seleccionar un equipo random para el comienzo de la simulacion*

Horas hombre: 0.25

Tarea4A: *Testear que el equipo atacante en el primer turno esta elegido aleatoriamente.*

Horas hombre: 0.25

US5: *Como game designer quiero que los turnos acaben si la pelota se va de la cancha o si se realiza un tiro exitoso al aro.*

Descripción: Los turnos durarán hasta que uno de los dos equipos convierte un punto o la pelota sale del campo de juego.

Criterios de aceptación:

1. En el log se mostrará que cuando la pelota se va fuera de la cancha el turno termina.
2. En el log se mostrará que cuando un tiro al aro fue exitoso, el turno termina.

Story Points: 3.0

Tarea5A: *Si la pelota se va o un equipo convierte, dar por finalizado el turno, setear el nuevo atacante, aumentar el contador de turnos y comenzar el nuevo turno.*

Horas hombre: 0.75

Tarea5A: *Diseñar un modelo donde la única forma de terminar el turno es que se valla la pelota o que un turno sea exitoso.*

Horas hombre: 0.75

Tarea5A: *Testear que si la pelota sale de cancha termina el turno*

Horas hombre: 0.25

Tarea5A: *Testear que si un tiro fue exitoso termina el turno.*

Horas hombre: 0.25

US6: *Como game designer quiero que, en caso de que se agoten los turnos y ambos equipos tengan el mismo puntaje, se jueguen 6 turnos extra de prórroga.*

Descripción: Si se terminan los turnos restantes y el puntaje de los equipos es el mismo se jugarán 6 turnos extra.

Criterios de aceptación:

1. En caso de que al finalizar los 40 turnos, los puntajes de los equipos sean los mismos, en el log aparecerán 6 turnos extras.

Story Points: 3.0

Tarea6A: *Cuando se terminan los turnos y los equipos están empatados el contador de turnos restantes se seteará en 6 y el juego se reanudará.*

Horas hombre: 0.75

Tarea6B: *Diseñar un modelo donde se acaben los turnos*

Horas hombre: 0.5

Tarea6C: *Diseñar un modelo que exista prorroga del partido.*

Horas hombre: 1.0

Tarea6D: *Testear que se juega prorroga en caso de terminar la simulacion empatados.*

Horas hombre: 0.5

US7: *Como game designer quiero que un jugador en ataque pueda lanzar la pelota al aro (por 2 o 3 puntos) o intentar un pase, y sólo eso.*

Descripción: Durante un turno de ataque, el jugador con la pelota deberá realizar siempre una de dos acciones: lanzar al aro o realizar un intento de pase.

Criterios de aceptación:

1. Las jugadas ofensivas sólo contienen acciones de pase o tiro al aro.
2. En el log las únicas acciones de ataque que se observan son del tipo tiro al aro (por dos o tres puntos), o intento de pase.

Story Points: 3.0

Tarea7A: Durante el ataque, el jugador que tiene la pelota, en función de lo que dicte la jugada, intentará pasar la pelota o tirar al aro.

Horas hombre: 1.0

Tarea7A: Diseñar un modelo donde se puedan realizar tiros al aro

Horas hombre: 1.5

Tarea7A: Diseñar un modelo donde se puedan realizar pases.

Horas hombre: 1.5

Tarea7A: Testear que se realicen correctamente tiros al aro de 2 puntos

Horas hombre: 0.5

Tarea7A: Testear que se realicen correctamente tiros al aro de 3 puntos

Horas hombre: 0.5

Tarea7A: Testear que un jugador pueda realizar un pase correctamente.

Horas hombre: 0.5

US8: Como game designer quiero que un jugador en defensa pueda intentar bloquear tiros al aro de otro jugador

Descripción: En el turno que un equipo está defendiendo, sus jugadores pueden tratar de bloquear los tiros del contrincante

Criterios de aceptación:

1. El participante en el log ve que cuando su equipo está atacando sólo pasan la pelota o lanzan al aro.
2. Las jugadas ofensivas sólo contienen acciones de pase o tiro al aro.

Story Points: 8.0

Tarea8A: Diseñar un modelo donde se puedan bloquear tiros al aro.

Horas hombre: 1.0

Tarea8B: Testear que un jugador puede bloquear tiros al aro de otro jugador

Horas hombre: 0.5

Tarea8C: Cuando un jugador atacante tira al aro, un jugador del equipo defensivo intentará bloquearlo. Si lo logra simula el rebote.

Horas hombre: 1.0

US9: *Como game designer quiero que un jugador pueda intentar interceptar pases*

Descripción: En el turno que un equipo está defendiendo, sus jugadores pueden tratar de interceptar los pases.

Criterios de aceptación:

1. El participante en el log ve que cuando su equipo está atacando sólo pasan la pelota o lanzan al aro.
2. Las jugadas ofensivas sólo contienen acciones de pase o tiro al aro.

Story Points: 2.0

Tarea9A: *Cuando un jugador atacante realiza un pase, un jugador del equipo defensivo intentará interceptarlo. Si lo logra, el equipo defensor debe pasar a ser atacante y el jugador que interceptó el pase es el nuevo dueño de la pelota.*

Horas hombre: 1.5

Tarea9B: *Testear que un jugador puede tratar de interceptar pases*

Horas hombre: 1.0

US10: *Como game designer quiero que cuando un jugador realice un tiro al aro con éxito, su equipo sume 2 o 3 puntos según corresponda.*

Descripción: Cuando un jugador hace un tiro exitoso, ese equipo va a sumar 2 o 3 puntos dependiendo del tiro.

Criterios de aceptación:

1. Como participante veo que el puntaje de mi equipo es igual a la suma de tiros exitosos de 2 o 3 puntos que aparecen en el log.
2. Cuando un jugador hace un tiro exitoso de dos o tres puntos, el puntaje de su equipo se incrementa en dos o tres puntos.

Story Points: 2.0

Tarea10A: *Cuando un jugador realice un tiro exitoso, sumarle al equipo del jugador 2 o 3 puntos dependiendo del tiro realizado.*

Horas hombre: 0.5

Tarea10B: *Diseñar un modelo que cuando un tiro sea exitoso el puntaje sume 2 o 3 puntos.*

Horas hombre: 0.5

Tarea10C: *Testear que cuando un tiro es exitoso incremente 2 o 3 puntos el puntaje.*

Horas hombre: 0.25

US11: *Como game designer quiero que cuando cuando un tiro al aro resulte fallido pero no bloqueado, la pelota quede dividida y ambos equipos tengan la posibilidad de recuperarla.*

Descripción: Cuando un tiro no es enceestado, se disputa la pelota entre ambos equipos.

Criterios de aceptación:

1. En el log se mostrará que cuando un tiro al aro fue fallido, la pelota queda dividida.

Story Points: 2.0

Tarea11A: *Cuando un tiro es fallado, llamar a la rutina de reboteo de pelota.*

Horas hombre: 0.2

Tarea11B: *Diseñar un modelo donde en caso de tiros fallidos pero no bloqueado, ambos equipos puedan recuperar la pelota*

Horas hombre: 0.75

Tarea11A: *Testear que en caso que el tiro sea fallido pero no bloqueado ambos equipos puedan recuperar la pelota.*

Horas hombre: 0.5

US12: *Como game designer quiero que cuando una pelota quede dividida, la misma se dispute en forma intercalada entre los jugadores del equipo defensivo y del equipo ofensivo, en orden decreciente según número de posición.*

Descripción: En caso de haber un rebote, la pelota se disputa de forma intercalada entre el defensor y el atacante en orden decreciente según su posición.

Criterios de aceptación:

1. Cuando en el log se informa que la pelota queda dividida, se debe ver, a continuación, la secuencia de disputas entre jugadores de ambos equipos.

Story Points: 8.0

Tarea12A: *Enfrentar a los jugadores, en orden.*

Horas hombre: 2.0

US13: *Como game designer quiero que cuando un jugador defensor gane una pelota dividida, se inviertan los roles de atacante y defensor entre ambos equipos.*

Descripción: En caso de que un jugador defensor gane un rebote, entonces este se queda con la pelota y el rol del atacante pasa a defender y el defensor pasa a atacar. (Invierten roles).

Criterios de aceptación:

1. Luego de un rebote, en el log puedo ver que si el jugador defensor lo ganó, la próxima acción será que él realice un pase o tire al aro.

Story Points: 3.0

Tarea13A: *Setear a ese jugador defensor como nuevo dueño de la pelota.*

Horas hombre: 0.25

Tarea13B: *Cambiar el equipo atacante.*

Horas hombre: 0.25

Tarea13C: *Diseñar un modelo donde se inviertan los roles de atacante y defensor en caso de pelota dividida*

Horas hombre: 0.5

Tarea13D: *Testear que en caso de pelota dividida se invierten los roles de los equipos.*

Horas hombre: 0.25

US14: *Como game designer quiero que cuando un jugador atacante gane una pelota dividida, continúe atacando su equipo.*

Descripción: En caso de que un jugador atacante gane un rebote, entonces este se queda con la pelota y continua el ataque.

Criterios de aceptación:

1. Luego de un rebote, en el log puedo ver que si el jugador atacante lo ganó, la próxima acción será que él realice un pase o tire al aro.

Story Points: 3.0

Tarea14A: *Cuando un jugador atacante gana un rebote setear a ese jugador como nuevo dueño de la pelota.*

Horas hombre: 0.75

Tarea14B: *Diseñar un modelo donde si el jugador que atacante gana una pelota dividida no cambie de rol*

Horas hombre: 0.25

Tarea14C: *Testear que en caso de que un jugador de un equipo atacante gana la pelota, ese equipo sigue atacando.*

Horas hombre: 0.25

US15: *Como game designer quiero que cuando ante una pelota dividida ninguno de los 10 jugadores gana el rebote, entonces la pelota salga de la cancha.*

Descripción: Las pelotas divididas, aquellas resultantes de un tiro fallido, serán disputadas por un jugador defensor y un atacante. Si ninguno la gana, la disputa pasará a otros dos. Si ningún jugador gana la pelota, ésta saldrá de la cancha.

Criterios de aceptación:

1. Después de un rebote en el que ningún jugador ganó la pelota el log indica que la misma se fue de la cancha.

Story Points: 3.0

Tarea15A: *Cuando un rebote es perdido por los dos equipos se escribirá en el log que la pelota salió de la cancha.*

Horas hombre: 0.15

Tarea15B: *Se restará uno de la cantidad de turnos restantes.*

Horas hombre: 0.15

Tarea15C: *Diseñar un modelo donde el turno termina si ninguno de los jugadores gana la pelota dividida.*

Horas hombre: 0.5

Tarea15D:

Horas hombre: 0.5

US16: *Como game designer quiero que cuando un jugador realice un pase de pelota exitoso y sin intercepción, aumente en un pequeño porcentaje la chance de éxito de tiro al aro del jugador que realiza el tiro posteriormente en la jugada.*

Descripción: Por cada pase exitoso de un jugador durante una jugada, se aumenta el umbral de éxito de tiro al aro del jugador que efectivamente realiza el tiro al final de la jugada. El aumento es de mientras no se supere un máximo de 0.3.

Criterios de aceptación:

1. Si el UE de un jugador es menor a 0.3 un pase exitoso incrementa su UE en $APG * 0.025$ pero el UE final no supera 0.3.
2. Si el UE de un jugador es mayor o igual a 0.3 un pase exitoso no lo incrementa.

Story Points: 1.0

Tarea16A: *Luego de un pase exitoso aumentar la cantidad de pases exitosos de la jugada en uno.*

Horas hombre: 0.15

Tarea16B: *Diseñar un modelo donde se pueda aumentar la chance de éxito de tiro al aro de un jugador.*

Horas hombre: 0.75

Tarea16C: *Testear que si un jugador realiza un pase exitoso aumenta su chance de éxito de tiro al aro.*

Horas hombre: 0.5

US17: *Como game designer quiero que cuando un jugador defensor tenga éxito interceptando un pase, se cambien los roles de atacante y defensor entre los equipos y continúe como atacante el equipo del defensor que robó la pelota.*

Descripción: Durante un turno, si un jugador defensor roba la pelota con éxito, entonces él se adueña de ella y el turno continúa con los roles de atacante y defensor invertidos.

Criterios de aceptación:

1. Cuando luego de que en el log aparece que un jugador de mi equipo roba la pelota, mi equipo comienza a atacar.
2. El jugador que obtiene la pelota luego de un robo exitoso es el mismo que intentó robarla.
3. En el log el jugador que realiza el primer pase o tiro luego de un robo exitoso es el mismo que la robó.
4. La cantidad de turnos restantes no se modifica luego de un robo exitoso.

Story Points: 3.0

Tarea17A: *Cuando un jugador roba la pelota exitosamente setearlo como nuevo dueño de la pelota.*

Horas hombre: 0.25

Tarea17B: *Setear al equipo atacante como el equipo del jugador que robó la pelota.*

Horas hombre: 0.25

Tarea17C: *Diseñar un modelo donde los roles de los equipos cambie si un jugador defensor intercepta un pase.*

Horas hombre: 0.75

Tarea17D: *Testear que si un jugador defensor intercepta un pase cambie de rol los equipos y siga atacando el equipo del jugador defensor.*

Horas hombre: 0.25

US18: *Como game designer quiero que cuando un pase sea fallido y además no se intercepte por un defensor contrario, la pelota salga de la cancha.*

Descripción: Los pases que no son interceptados por un jugador defensor ni recibidos por el destinatario del mismo causan que la pelota salga de la cancha.

Criterios de aceptación:

1. En el log aparecerá que la pelota sale de la cancha cuando el pase es fallido y ningún defensor contrario la intercepta.

Story Points: 5.0

Tarea18A: *Informar en el log que la pelota salió de la cancha.*

Horas hombre: 0.25

Tarea18B: *Reducir en uno la cantidad de turnos restantes.*

Horas hombre: 0.1

Tarea18C: *Diseñar un modelo que permita que si el pase es fallido y no es interceptado se termine el turno.*

Horas hombre: 0.50

Tarea18D: *Testear que si un pase es fallido y nadie lo intercepta se termine el turno*

Horas hombre: 0.25

US19: *Como game designer quiero que el técnico tenga un libro de jugadas para jugar adecuadamente.*

Descripción: Cada director técnico representa a un libro de jugadas. El libro de jugadas se lee de la entrada.

Criterios de aceptación:

1. En el log se podrá reconocer a partir del número de pases y quién realiza los tiros si la jugada se corresponde con una del libro de jugadas del director técnico elegido para ese equipo.

Story Points: 5.0

Tarea19A: *Leer los libros de jugadas y el técnico al que pertenecen de la entrada.*

Horas hombre: 1.0

Tarea19B: *Guardar las jugadas y vincularlas al técnico al que le pertenecen.*

Horas hombre: 1.0

Tarea19C: *Diseñar un modelo donde el tecnico tenga un libro de jugadas*

Horas hombre: 0.75

Tarea19D: *Testear que el tecnico tenga un libro de jugadas.*

Horas hombre: 0.25

US20: *Como game designer quiero que el técnico tenga una frecuencia para cada jugada del libro*

Descripción: Cada jugada perteneciente al libro de jugadas del técnico de un equipo tiene configurada una frecuencia. Esta frecuencia es la que se tiene en cuenta a la hora de elegir cuál jugada usar.

Criterios de aceptación:

1. La entrada cuenta con una frecuencia para cada jugada.
2. Luego de varios juegos todas las jugadas ocurren con una frecuencia cercana a la recibida en la entrada.

Story Points: 2.0

Tarea20A: *Al elegir una jugada del libro de jugadas, tener en cuenta la frecuencia con la que debe elegirse cada jugada.*

Horas hombre: 0.15

Tarea20B: *Diseñar un modelo donde el tecnico tenga una frecuencia de cada jugada.*

Horas hombre: 0.25

Tarea20C: *Testear que el tecnico elija la jugada con una frecuencia.*

Horas hombre: 0.25

US21: *Como game designer quiero que al comenzar el ataque de un equipo (ya sea al iniciar un turno, recuperar una pelota dividida, o interceptar un pase), se elija una jugada ofensiva del libro de jugadas del director técnico del equipo atacante, teniendo en cuenta la frecuencia asignada a las mismas.*

Descripción: La jugada se elige del libro de jugadas del DT. Cada jugada tiene una frecuencia dada para ser elegida.

Criterios de aceptación:

1. La jugada ofensiva elegida tiene que pertenecer al libro de jugadas del director técnico del equipo.
2. Si el game designer elige sucesivas jugadas ofensivas la frecuencia relativa de cada una se acerca a la frecuencia deseada por el director técnico.

Story Points: 2.0

Tarea21A: *Obtener el libro de jugadas del director técnico del equipo atacante.*

Horas hombre: 0.15

Tarea21B: *Tirar un random para elegir una de las jugadas ofensivas.*

Horas hombre: 0.15

Tarea21C: *Asignarle la jugada al equipo atacante.*

Horas hombre: 0.15

Tarea21D: *Testear que la jugada elegida sea del libro del tecnico.*

Horas hombre: 0.25

US22: *Como game designer quiero que cuando el equipo atacante elija una jugada ofensiva, el equipo defensor escoja una jugada defensiva de su libro de jugadas, teniendo en cuenta la frecuencia asignada a las mismas.*

Descripción: La jugada se elige del libro de jugadas del DT. Cada jugada tiene una frecuencia dada para ser elegida.

Criterios de aceptación:

1. La jugada defensiva elegida tiene que pertenecer al libro de jugadas del director técnico del equipo.
2. Si el game designer elige sucesivas jugadas defensivas la frecuencia relativa de cada una se acerca a la frecuencia deseada por el director técnico.

Story Points: 2.0

Tarea22A: *Obtener el libro de jugadas del director técnico del equipo defensor.*

Horas hombre: 0.15

Tarea22B: *Tirar un random para elegir una de las jugadas defensivas.*

Horas hombre: 0.15

Tarea22C: *Asignarle la jugada al equipo defensor.*

Horas hombre: 0.15

US23: *Como game designer quiero que se pueda usar la jugada ofensiva colectiva externa.*

Descripción: Una posible jugada es la Jugada Ofensiva Colectiva Externa, en la cual la pelota comienza en la manos del base, deben hacerse k-1 pases con éxito y llegar en el pase k a las manos de un jugador que juegue en las posiciones 1, 2 o 3, el cual tira al aro por 3 puntos.

Criterios de aceptación:

1. A partir del log puedo ver la utilización de la jugada ofensiva externa cada vez que es realizada con éxito si ésta pertenece al libro de jugadas del director técnico de alguno de los equipos que está jugando.

Story Points: 3.0

Tarea23A: *Diseñar un modelo donde se pueda usar la jugada colectiva externa.*

Horas hombre: 1.0

Tarea23B: *Testear el uso de la jugada colectiva externa.*

Horas hombre: 0.50

US24: *Como game designer quiero que se pueda usar la jugada ofensiva colectiva interna.*

Descripción: Una posible jugada es la Jugada Ofensiva Colectiva Interna, en la cual la jugada comienza con la pelota en las manos del base, deben hacerse k-1 pases con éxito y llegar en el pase numero k a las manos de un jugador que juegue en las posiciones 4 o 5 quien tira al aro por 2 puntos.

Criterios de aceptación:

1. A partir del log puedo ver la utilización de la jugada ofensiva interna cada vez que es realizada con éxito si ésta pertenece al libro de jugadas del director técnico de alguno de los equipos que está jugando.

Story Points: 3.0

Tarea24A: *Diseñar un modelo donde se pueda utilizar la jugada ofensiva colectiva interna.*

Horas hombre: 1.0

Tarea24B: *Testear el uso de la jugada colectiva interna.*

Horas hombre: 0.5

US25: *Como game designer quiero que se pueda usar la jugada ofensiva MVP.*

Descripción: Una posible jugada es la Jugada MVP, en la cual la pelota comienza en manos del base, quien se la pasa al jugador franquicia del equipo (estrella) para que este tire al aro con el tiro que quiera. Cada vez que se usa esta jugada el jugador estrella pierde 0.01 de su Umbral de tiro de 2 y 3 puntos.

Criterios de aceptación:

1. A partir del log puedo ver la utilización de la jugada MVP cada vez que es realizada con éxito si ésta pertenece al libro de jugadas del director técnico de alguno de los equipos que está jugando.

Story Points: 3.0

Tarea25A: *La jugada ofensiva MVP debe poder ser recibida en la entrada.*

Horas hombre: 0.15

Tarea25B: *Cuando la jugada es seleccionada se debe setear como jugador con pelota al base. Éste debe intentar realizar un pase al MVP. Si el pase es exitoso el MVP debe tirar al aro por 2 o 3 puntos.*

Horas hombre: 0.15

Tarea25C: *Elegir el tipo del tiro al azar.*

Horas hombre: 0.15

Tarea25D: *Se informará en el log el tiro al aro.*

Horas hombre: 0.15

Tarea25E: *Reducir el UE de tiro del MVP en 0.01.*

Horas hombre: 0.15

Tarea25F: *Diseñar un modelo que utilice la jugada ofensiva MVP.*

Horas hombre: 1.0

Tarea25G: *Testear la jugada ofensiva MVP.*

Horas hombre: 0.5

US26: *Como game designer quiero que se pueda usar la jugada ofensiva Contra Ataque.*

Descripción: Una posible jugada es la Jugada Ofensiva Contra Ataque, en la cual el jugador que robó la pelota en una jugada defensiva inmediatamente anterior a la ofensiva actual, tira al aro con el tiro que quiere y sin oposición alguna de la defensa.

Criterios de aceptación:

1. Si la jugada elegida fue la ofensiva contra ataque en el log se deberá ver que el jugador que robó la pelota en la jugada defensiva anterior a la ofensiva actual tira al aro.

Story Points: 3.0

Tarea26A: *La jugada ofensiva contra ataque debe poder ser recibida en la entrada.*

Horas hombre: 0.15

Tarea26B: *La jugada sólo puede ser seleccionada luego de un haber interceptado un pase exitosamente.*

Horas hombre: 0.15

Tarea26C: *El jugador que interceptó el pase tirará al aro por 2 o 3 puntos.*

Horas hombre: 0.15

Tarea26D: *Ningún jugador del equipo contrario intentará bloquear el tiro.*

Horas hombre: 0.15

Tarea26E: *El tipo de tiro se elegirá al azar.*

Horas hombre: 0.15

Tarea26F: *Se informará en el log el tiro al aro.*

Horas hombre: 0.15

Tarea26G: *Diseñar un modelo donde se pueda usar la jugada ofensiva Contra Ataque.*

Horas hombre: 1.0

Tarea26H: *Testear la jugada ofensiva Contra Ataque.*

Horas hombre: 0.5

US27: *Como game designer quiero que se pueda usar la jugada defensiva Hombre a Hombre.*

Descripción: Una posible jugada es la Jugada Hombre a Hombre, en la cual cada jugador defensor sigue al jugador del equipo contrario que ocupa su misma posición. Cuando este jugador tira al aro, el jugador defensor intenta bloquearlo, y cuando intenta pasar la pelota este jugador intenta interceptar el pase.

Criterios de aceptación:

1. Si la jugada elegida fue la defensiva Hombre a Hombre en el log se deberá ver que el jugador que intenta bloquear es un jugador de la misma posición.

Story Points: 3.0

Tarea27A: *Cuando la jugada defensiva seleccionada es la Hombre a Hombre cuando un jugador atacante realiza un pase el jugador del equipo defensivo que intentará interceptarlo será el que ocupe su misma posición.*

Horas hombre: 0.5

Tarea27A: *Diseñar un modelo donde se pueda usar la jugada defensiva Hombre a Hombre.*

Horas hombre: 1.0

Tarea27A: *Testear el uso de la jugada defensiva Hombre a Hombre.*

Horas hombre: 0.50

US28: *Como game designer quiero que se pueda usar la Dos Cada.*

Descripción: Una posible jugada ofensiva es la jugada Dos Cada, en donde cada jugador del equipo atacante toca la pelota dos veces, el último jugador en tocar la pelota 2 veces lanza la pelota al aro.

Criterios de aceptación:

1. En el log se notará que al elegir la jugada ofensiva Dos Cada, cada jugador tocará la pelota exactamente dos veces.
2. En el log se notará que al elegir la jugada ofensiva Dos Cada, ningún jugador puede tocar la pelota más o menos de dos veces.

Story Points: 5.0

Tarea28A: *Se iniciará un contador por cada jugador en 0 y uno global en 5. Cuando un jugador obtiene la pelota se aumentará su contador. Si el contador del jugador es igual a 2 el contador global se decrementará. Cuando el contador global llegue a 0 el jugador que posee la pelota tirará al aro.*

Horas hombre: 0.75

Tarea28B: *El jugador al que pasar la pelota se elegirá al azar entre los que menos veces la tocaron.*

Horas hombre: 0.15

Tarea28C: *El tipo de tiro se elegirá al azar.*

Horas hombre: 0.15

Tarea28D: *Se informará en el log el tiro al aro.*

Horas hombre: 0.15

Tarea28E: *Diseñar un modelo donde se pueda usar la jugada Dos Cada.*

Horas hombre: 1.0

Tarea28F: *Testear el uso de la jugada Dos Cada.*

Horas hombre: 0.50

US29: *Como game designer quiero que se pueda usar la FiboJugada.*

Descripción: Una posible jugada ofensiva es la jugada FiboJugada, en donde cada jugador, posicionados en la posición i (de a a 5), toca la pelota una cantidad de veces igual al i -ésimo número de Fibonacci.

Criterios de aceptación:

1. En el log se notará que al elegir la jugada ofensiva FiboJugada, cada jugador tocara la pelota al menos una cantidad de veces igual al número de Fibonacci que le corresponde.
2. En el log se notará que al elegir la jugada ofensiva FiboJugada, ningún jugador puede tocar la pelota mas o menos que su correspondiente número de Fibonacci.

Story Points: 5.0

Tarea29A: *Se iniciará un contador por cada jugador en 0 y uno global en 5. Cuando un jugador obtiene la pelota se aumentará su contador. Si el contador del jugador es igual a su correspondiente número de Fibonacci el contador global se decrementará. Cuando el contador global llegue a 0 el jugador que posee la pelota tirará al aro.*

Horas hombre: 0.75

Tarea29B: *El jugador al que pasar la pelota se elegirá al azar entre los que menos veces la tocaron*

Horas hombre: 0.15

Tarea29C: *El tipo de tiro se elegirá al azar.*

Horas hombre: 0.15

Tarea29D: *Se informará en el log el tiro al aro.*

Horas hombre: 0.15

Tarea29F: *Diseñar un modelo donde se pueda usar la jugada FiboJugada.*

Horas hombre: 1.0

Tarea29G: *Testear el uso de la jugada FiboJugada.*

Horas hombre: 0.50

US30: *Como game designer quiero que cada UE de una acción X sea la probabilidad de éxito al realizar la acción X.*

Descripción: Las acciones se resuelven obteniendo un número aleatorio de 0 a 1 con precisión de al menos 3 decimales. Si el valor obtenido es menor o igual al umbral de éxito del jugador, el mismo realiza la acción. Si es superior, falla en la misma.

Criterios de aceptación:

1. Se pasará un jugador con UE de 1 y en el log se verá que todas sus acciones son exitosas.
2. Se pasará un jugador con UE de 0 y en el log se verá que todas sus acciones fallan.

Story Points: 3.0

Tarea30A: *Setear el UE de cada jugador.*

Horas hombre: 0.25

Tarea30B: *Conseguir una forma de obtener un número aleatorio entre 0 y 1 con una precisión de la menos 3 decimales.*

Horas hombre: 0.25

Tarea30C: *Cuando un jugador realiza una acción se elegirá un número al azar y se comparará con el UE seteado para decidir si el resultado es exitoso.*

Horas hombre: 0.25

Tarea30D: *Diseñar un modelo donde el Umbral de Exito sea probabilístico.*

Horas hombre: 0.25

Tarea30E: *Testear la probabilidad de exito de un Umbra de Exito*

Horas hombre: 0.15

US31: *Como game designer quiero que el UE de tiro por 2 puntos sea FG para hacer el juego más realista.*

Descripción: En el caso de un tiro doble, el umbral de éxito de un jugador se calcula como FG.

Criterios de aceptación:

1. La frecuencia relativa de dobles exitosos realizados por un jugador se acerca a FG.

Story Points: 2.0

Tarea31A: *Setear el FG para todos los jugadores.*

Horas hombre: 0.25

Tarea31B: *Cuando un jugador tira por 2 puntos utilizar como UE su FG.*

Horas hombre: 0.15

Tarea31B: *Diseñar un modelo donde el UE de tiro por 2 puntos sea FG.*

Horas hombre: 0.50

Tarea31B: *Testear que el UE de tiro por 2 puntos sea igual a FG.*

Horas hombre: 0.15

US32: *Como game designer quiero que el UE de tiro por 3 puntos sea 3P para hacer el juego más realista.*

Descripción: En el caso de un tiro triple, el umbral de éxito de un jugador se calcula como 3P.

Criterios de aceptación:

1. La frecuencia relativa de triples exitosos realizados por un jugador se acerca a 3P.

Story Points: 2.0

Tarea32A: *Setear P para todos los jugadores.*

Horas hombre: 0.25

Tarea32B: *Cuando un jugador tira por 3 puntos utilizar como UE 3P.*

Horas hombre: 0.15

Tarea32C: *Diseñar un modelo donde el UE de tiro por 3 puntos sea 3P.*

Horas hombre: 0.50

Tarea32D: *Testear que el UE de tiro por 3 puntos sea 3P.*

Horas hombre: 0.15

US33: *Como game designer quiero que el UE de bloqueo sea $BPG * 0.2$ para hacer el juego más realista.*

Descripción: Cada jugador tiene un Umbral de Exito Bloqueando y éste valor se calcula en función del stat BPG y es igual a $BPG * 0.2$.

Criterios de aceptación:

1. La frecuencia relativa de bloqueos exitosos realizados por un jugador se acerca a $BPG * 0.2$.

Story Points: 2.0

Tarea33A: *Setear BPG para todos los jugadores.*

Horas hombre: 0.25

Tarea33B: *Cuando un jugador intenta bloquear un tiro utilizar como UE su BPG * 0.2.*

Horas hombre: 0.15

Tarea33C: *Diseñar un modelo donde el UE de bloqueo se calcule en función del BPG.*

Horas hombre: 0.50

Tarea33D: *Testear el cálculo del UE de bloqueo.*

Horas hombre: 0.15

US34: *Como game designer quiero que el UE de pase de pelota sea $1 - TO * 0.1$ para hacer el juego más realista.*

Descripción: En el caso de los pases de pelota, el Umbral de Éxito de un jugador se calcula como $1 - TO * 0.1$.

Criterios de aceptación:

1. La frecuencia relativa de pases exitosos realizados por un jugador se acerca a $1 - TO * 0.1$.

Story Points: 2.0

Tarea34A: *Setear TO para todos los jugadores.*

Horas hombre: 0.25

Tarea34B: *Cuando un jugador intenta realizar un pase utilizar como UE su $1 - TO * 0.1$.*

Horas hombre: 0.15

Tarea34B: *Diseñar un modelo donde el UE de pase de pelota se calcule en función de TO.*

Horas hombre: 0.50

Tarea34B: *Testear el calculo del UE de pase de pelota.*

Horas hombre: 0.15

US35: *Como game designer quiero que el UE de robo de pelota sea $SPG * 0.2$ para hacer el juego más realista.*

Descripción: Cada jugador tiene un Umbral de Éxito Robando la Pelota y éste valor se calcula en función del stats SPG y es igual a $SPG * 0.2$.

Criterios de aceptación:

1. La frecuencia relativa de robos exitosos realizados por un jugador se acerca a $SPG * 0.2$.

Story Points: 2.0

Tarea35A: *Setear SPG para todos los jugadores.*

Horas hombre: 0.25

Tarea35B: *Cuando un jugador intenta realizar un pase utilizar como UE su $SPG * 0.2$.*

Horas hombre: 0.15

Tarea35C: *Diseñar un mdeolo donde el UE de robo de pelota se calcule en función de SPG.*

Horas hombre: 0.50

Tarea35D: *Testear el calculo del UE de robo de pelota.*

Horas hombre: 0.15

US36: *Como game designer quiero que el UE de reboteo de pelota sea $RPG * 0.05$ para hacer el juego más realista.*

Descripción: Cada jugador tiene un Umbral de Éxito Rebotando la pelota y este valor se calcula en función del stats RPG y es igual a $RPG * 0.05$

Criterios de aceptación:

1. La frecuencia relativa de rebotes exitosos realizados por un jugador se acerca a $RPG * 0.05$.

Story Points: 2.0

Tarea36A: *Setear RPG para todos los jugadores.*

Horas hombre: 0.25

Tarea36B: *Cuando un jugador intenta realizar un pase utilizar como UE su $RPG * 0.5$.*

Horas hombre: 0.15

Tarea36C: *Diseñar un modelo donde el UE de reboteo de pelota se calcule en función del RPG*

Horas hombre: 0.50

Tarea36D: *Testear el calculo del UE de reboteo de pelota.*

Horas hombre: 0.15

US37: *Como game designer quiero tener las estadísticas originales de los jugadores para que la simulación sea más realista.*

Descripción: Previo a la simulación, obtenemos las estadísticas reales de cada jugador involucrado en uno de los equipos a partir de la API oficial de la NBA (<http://stats.nba.com/>).

Criterios de aceptación:

1. Los logs de las simulaciones se parecen a los partidos reales.

Story Points: 13.0

Tarea37A: *Acceder al sitio de la NBA y cargar las estadísticas de al menos 5 jugadores para que el programa de la simulación puede cargarlos.*

Horas hombre: 2.0

Tarea37B: *Acordar el formato de la entrada para las estadísticas de los jugadores.*

Horas hombre: 1.5

Tarea37C: *Parsear el archivo con las estadísticas de los jugadores.*

Horas hombre: 1.5

Tarea37C: *Diseñar un modelo que contenga las estadísticas reales de los jugadores.*

Horas hombre: 1.5

Tarea37C: *Testear que las estadísticas de los jugadores sean realistas.*

Horas hombre: 1.0

US38: *Como game designer quiero que la popularidad en Twitter de los jugadores para afecte mejorar su performance.*

Descripción: Según la cantidad de comentarios del jugador en Twitter las estadísticas de este mejoran. Los valores de los umbrales van a mejorar o empeorar dependiendo de este valor.

Criterios de aceptación:

1. En el log se puede ver que los jugadores cuando tienen muchos comentarios en Twitter realizan más acciones exitosas.

Story Points: 13.0

Tarea38A: *Se registrará la cantidad de comentarios que tiene cada jugador elegido en US36 referidos a él en Twitter.*

Horas hombre: 2.0

Tarea38B: *Las fórmulas se adaptarán para tener en cuenta éste peso.*

Horas hombre: 1.5

Tarea38B: *Diseñar un modelo donde se utilice los tweets de Twitter para mejorar el rendimiento de los jugadores.*

Horas hombre: 1.5

Tarea38B: *Testear que el rendimiento de los jugadores mejorar si su popularidad en twitter aumenta.*

Horas hombre: 1.0

4. Diseño

4.1. Descripción informal

El sistema desarrollado consiste en dos programas que pueden ser invocados mediante línea de comandos, `generar-base.py` y `simular-partido.py`. El primero tiene la finalidad de generar una base de datos donde se persisten tanto las estadísticas de los jugadores como los libros de jugadas de los directores técnicos, y el segundo ejecuta una simulación en base a un archivo de entrada de tipo JSON conteniendo la composición de los equipos a enfrentarse.

El programa `generar-base.py` genera un archivo de extensión `db` utilizando el motor SQLite, y luego popula la base de datos con estadísticas de un conjunto de jugadores desde la API oficial de la NBA¹. Adicionalmente, se definen cuáles jugadas tiene cada director técnico en su libro de jugadas, con su correspondiente nivel de preferencia (un número en una escala arbitraria entre 1 y 5).

Una vez generada la base de datos, ya se puede utilizar el programa `simular-partido.py` para realizar una simulación. El mismo lee la conformación de los equipos, y ejecuta la simulación imprimiendo por salida estándar el log generado durante la misma.

4.2. Modelo de la simulación

El simulador está implementado en el módulo Python `simulacion`, el cual consiste de una serie de clases de tipo `Simulador`, representando cada una la simulación de un partido de básquet a distintos niveles de detalle. Hay un total de cuatro niveles de simulación:

1. `SimuladorPartido` representa la simulación de un partido en su totalidad; una ronda de 40 turnos, seguida por tantas rondas de 6 turnos como sean necesarias hasta que el partido tenga un ganador.
2. `SimuladorTurno` representa la simulación de un turno dentro de un partido, estando compuesto por una sucesión de jugadas.
3. `SimuladorJugada` representa la simulación de una jugada dentro de un turno, estando compuesta por una sucesión de momentos.
4. `SimuladorMomento` representa la simulación de un instante dentro de una jugada, en el cual el jugador atacante en posesión de la pelota realiza una acción, y un jugador defensor realiza la acción defensiva correspondiente.

Todos los niveles son polimórficos con respecto al método `simular`. La implementación de éste método sigue el mismo esquema en todos los niveles, y es el siguiente:

1. Registrar comienzo de la simulación de este nivel.
2. Simular una o más ejecuciones del nivel inmediatamente inferior.
 - 2.1. Por cada ejecución del nivel inmediatamente inferior, actualizar el estado de este nivel.
3. Registrar fin de la simulación de este nivel.

4.3. Diagramas de objeto

Incluimos un diagrama de objetos por cada nivel de simulación, representando el estado que se mantiene en cada uno de ellos, así como su interacción con el nivel inmediatamente inferior (si aplica).

En cada diagrama, se indican en rojo las relaciones de conocimiento que surgen del pasaje de parámetros en la construcción. Así podemos ver el estado con el que se construye cada uno de los niveles de simulación.

¹API oficial de la NBA: <http://stats.nba.com>

4.3.1. Simulador de Partido

Diagrama de objetos de un simulador de partido, para el caso en que un turno termina en conversi3n triple, justo antes que el resultado del turno actualice el simulador de partido.

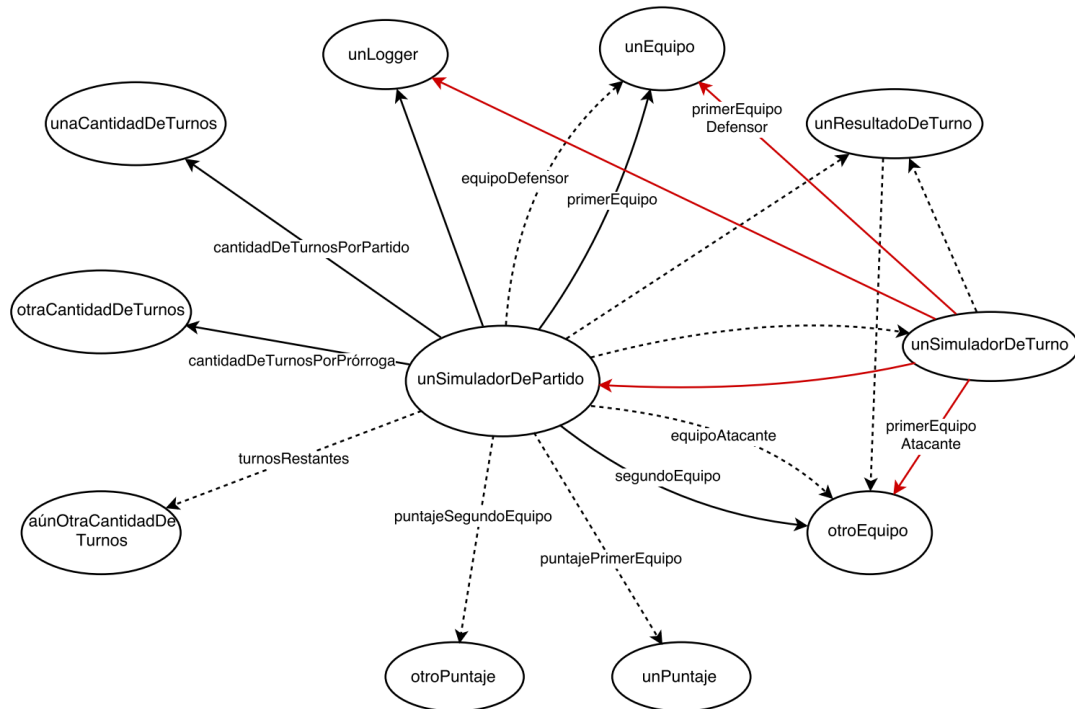
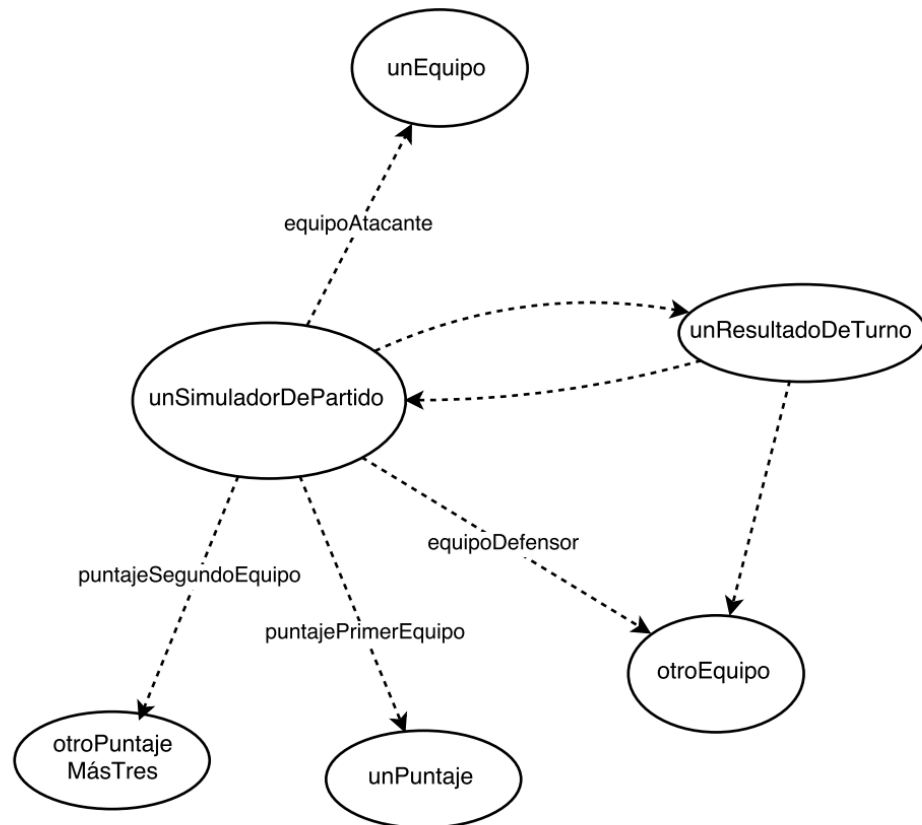


Diagrama de objetos luego de actualizar el simulador de partido.



4.3.2. Simulador de Turno

Diagrama de objetos de un simulador de turnos, para el caso en que una jugada termina en pase robado, justo antes que el resultado de la jugada actualice el simulador de turno.

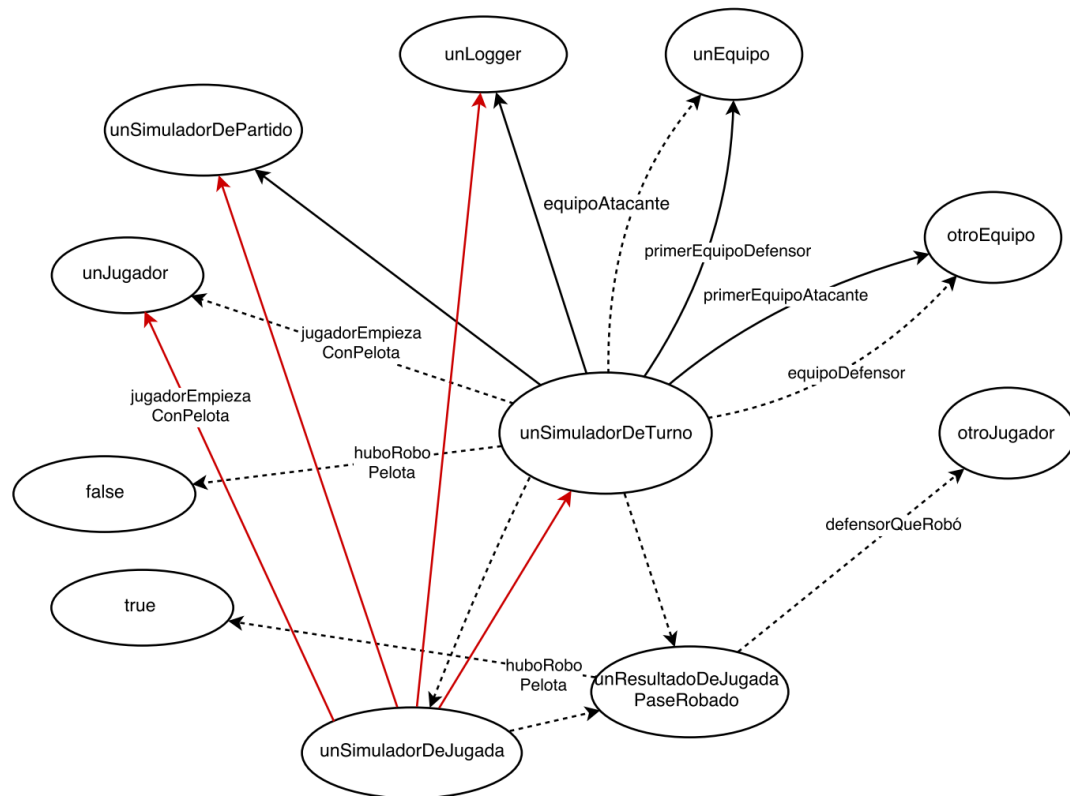


Diagrama de objetos luego de actualizar el turno.

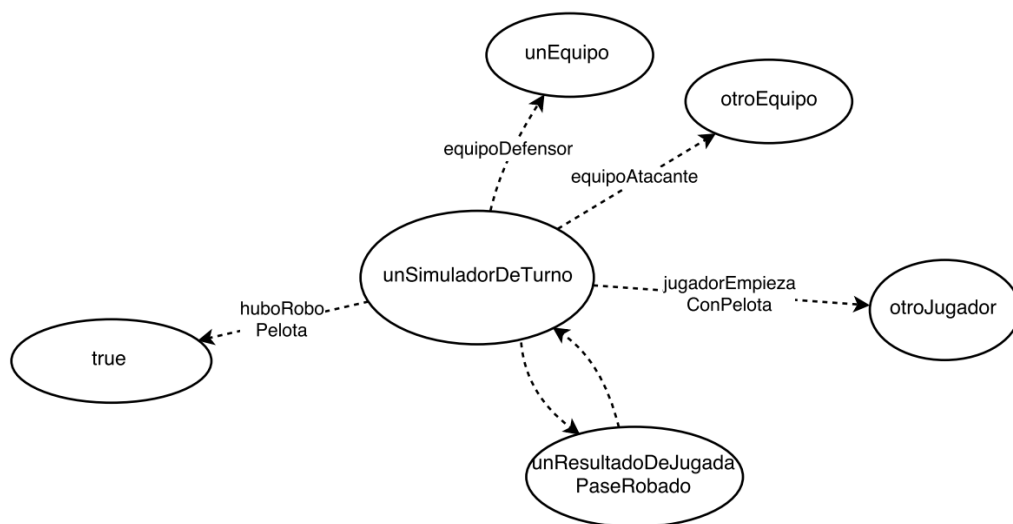
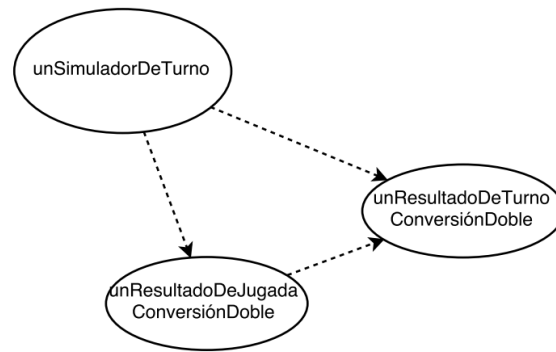


Diagrama de objetos luego de la última jugada del turno.



4.3.3. Simulador de Jugada

Diagrama de objetos de un simulador de jugada, para el caso en que un momento termina en pase exitoso, justo antes que el resultado del momento actualice el simulador de jugada.

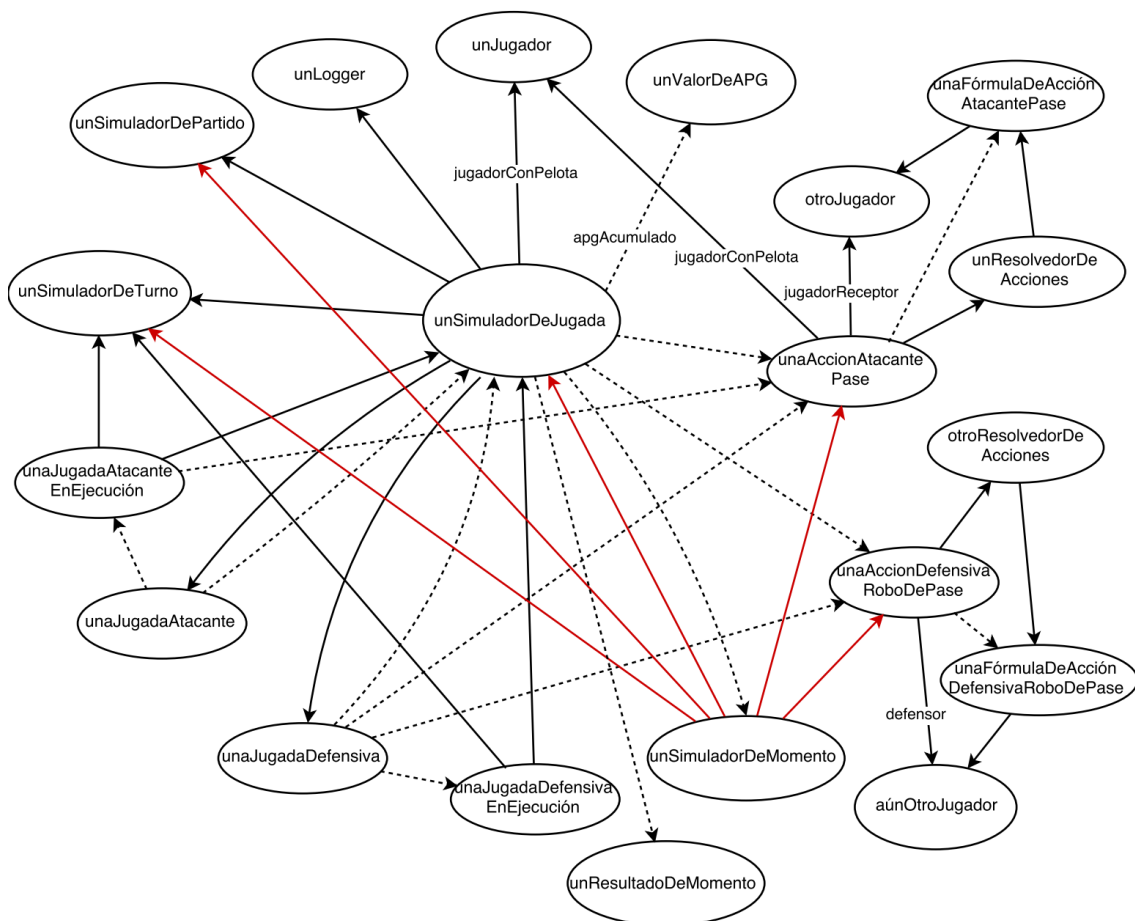
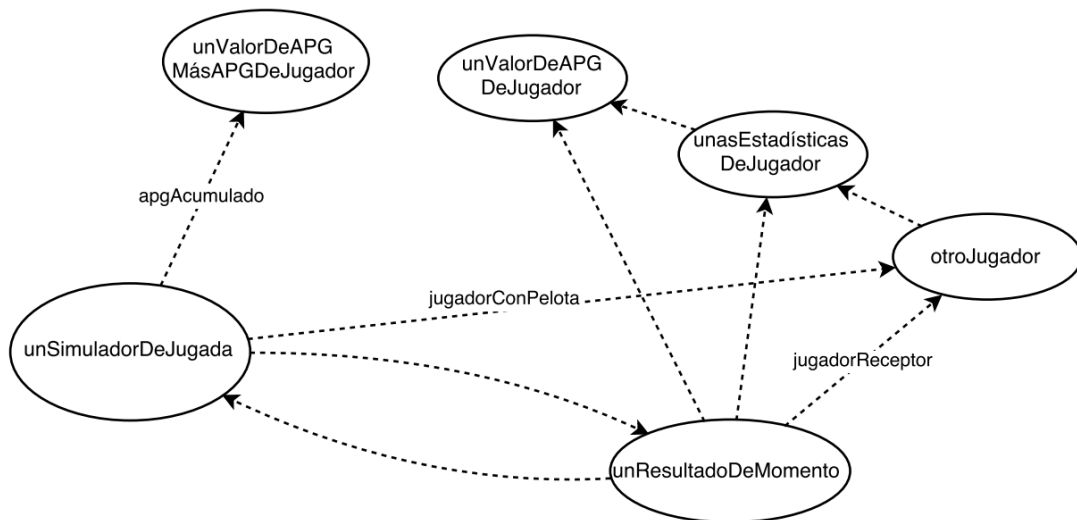
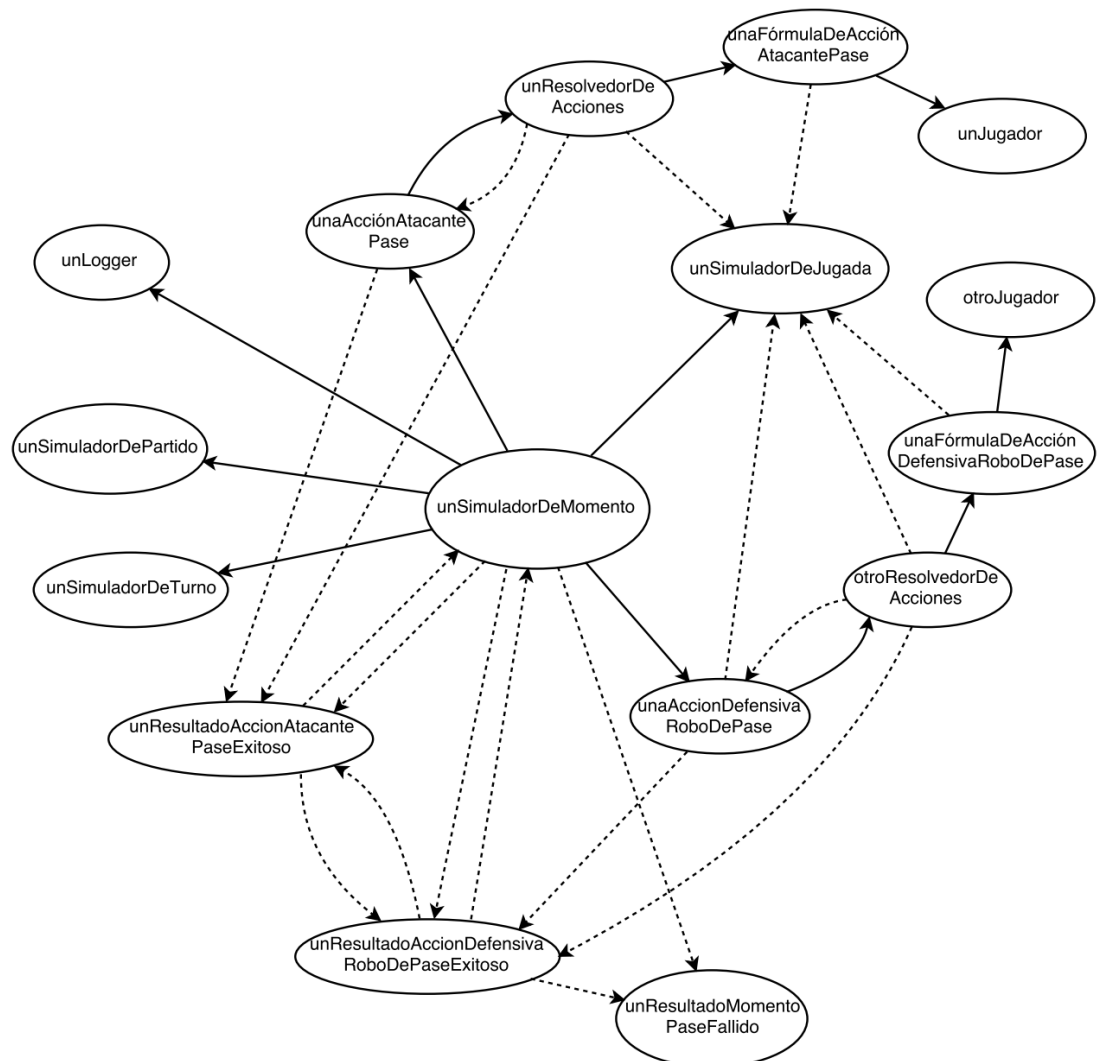


Diagrama de objetos luego de actualizar el simulador de jugada.



4.3.4. Simulador de Momento

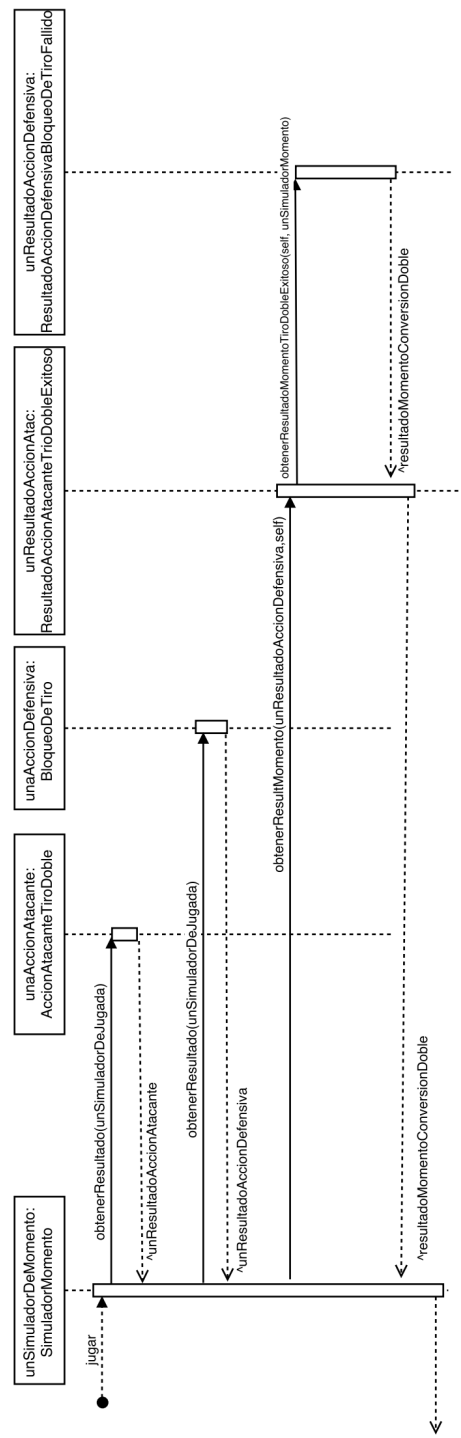
Diagrama de objetos de un simulador de momento, para el caso de un pase fallido por robo de pelota.



4.4. Diagrama de secuencia

Incluimos un diagrama de secuencia que abarca la secuencia de colaboraciones que desencadena el envío del mensaje jugar (simular, finalmente) a la clase SimuladorMomento con el objetivo de mostrar claramente la interacción mediante la cual se resuelven las acciones atacantes y defensivas en un Momento particular.

El diagrama muestra que para simular un MomentoDeJugada se deben obtener el resultado de simular ambas acciones utilizando las fórmulas de resolución de acciones, ofensiva y defensiva, independientemente. Una vez definido el resultado de cada una de las acciones mediante un double dispatch se obtiene el resultado del momento. El double dispatch se realiza entre el resultado de la acción atacante al cual se le envía el mensaje obtenerResultadoMomento recibiendo como colaborador externo al resultado de la acción defensiva. El resultado de la acción atacante invoca el mensaje correspondiente sobre el resultado de la acción defensiva en función de su propio tipo. Finalmente, el resultado de acción defensiva devolverá el resultado del momento en función de su propio tipo, completando el double dispatch.



4.5. Diagramas de clase

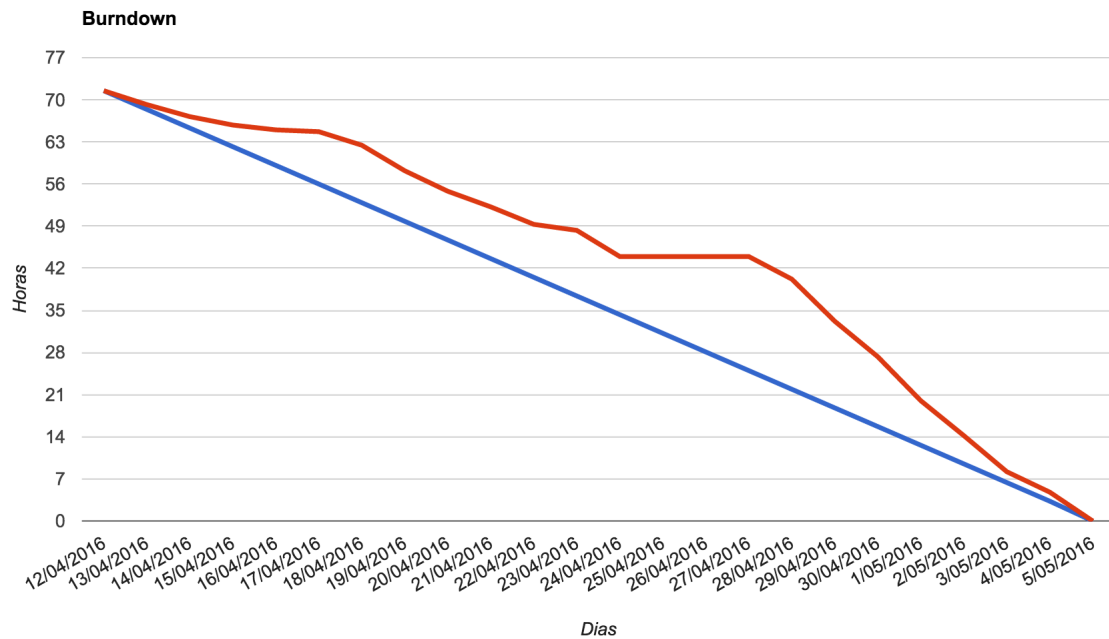
El correspondiente diagrama de clases lo enviamos adjunto con el informe a causa del tamaño del gráfico. A continuación, realizamos un par de aclaraciones sobre el gráfico:

1. Agregamos clases en color verde, estas clases son clases replicadas, es decir, clases que duplicamos para hacer mas legible el diagrama.
2. Los conectores de herencia estan hechos con flechas de tamaño mas grande que las comunes, de esta forma el diagrama es mas legible.

5. Burndown

A continuación mostramos el diagrama Burndown, el cual se utiliza para graficar el trabajo pendiente; en particular, la cantidad de horas pendientes del proyecto a lo largo del tiempo. En el gráfico vemos la recta azul que simboliza un progreso equitativo y cumpliendo las horas propuestas, y en rojo vemos la curva real. Idealmente, la curva que describe la realidad se parecería lo más posible a la lineal.

Analizando el gráfico podemos ver que el proyecto se llevó adelante a un ritmo que se aproxima a un ritmo lineal, a excepción de algunos picos donde se demoró el desarrollo, pero luego convergió al resultado esperado.



6. Conclusión

El desarrollo del sistema nos facilitó la apreciación de la utilidad de varias de las herramientas que usamos como los gráficos y las heurísticas de diseño. Sin embargo, en varios momentos el orden en el que realizamos las tareas consideramos que nos perjudicó. Por ejemplo, el subdividir el trabajo en stories sin haber pensado en el diseño del sistema antes, produjo una correspondencia pobre entre las stories que habíamos definido y las tareas que finalmente tuvimos que realizar.

Los gráficos resultaron ser una buena base para discutir las opciones de diseño de forma barata sin tener que invertir el tiempo que hubiera tomado programar cada una de las alternativas. Sin embargo, sucedió varias veces que no nos percatábamos de ciertos problemas hasta que no los implementábamos. Además, uno de los mayores desafíos que encontramos en el TP fue pensar diseños que satisficieran todas las heurísticas de diseño que teníamos presentes (alta cohesión, bajo acoplamiento, alta modularidad, etc.). La mayoría de las veces tuvimos que evaluar las ventajas y desventajas de cada alternativa para finalmente decidir la que nos pareciera mejor para la circunstancia particular y los posibles ejes de cambio.

Finalmente, nos gustaría comentar que la utilización de la herramienta de Scrum Target Process no nos pareció que nos ayudara a organizarnos o mejorar. Probablemente esto se deba a que el proyecto fue demasiado corto para poder utilizar las herramientas de seguimiento de Target Process durante las retrospectivas y organizarnos mejor.