



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 2

ISW II

Primer Cuatrimestre de 2015

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

# Índice

|   |           |
|---|-----------|
| <b>1. Introducción</b>  | <b>3</b>  |
| <b>2. Casos de uso</b>  | <b>4</b>  |
| <b>3. Planificación</b>   | <b>5</b>  |
| 3.1. Iteraciones . . . . .                                      | 5         |
| 3.1.1. Fase de iniciación . . . . .                             | 5         |
| 3.1.2. Fase de elaboración . . . . .                            | 5         |
| 3.1.3. Fase de construcción . . . . .                           | 6         |
| 3.1.4. Fase de transición . . . . .                             | 6         |
| 3.2. Alcance de casos de usos de la primera iteración . . . . . | 6         |
| 3.3. Tareas CU Primera iteración . . . . .                      | 7         |
| 3.4. Dependencias entre tareas . . . . .                        | 8         |
| 3.5. Detalle Primera iteración . . . . .                        | 9         |
| 3.6. Plan de Proyecto . . . . .                                 | 10        |
| <b>4. Análisis de riesgos</b>                                   | <b>12</b> |

## 1. Introducción

En trabajo desarrollamos sobre la organización de un proyecto basado en el TP1 pero de mucho mayor alcance, donde la aplicación de desafíos deportivos basados en simulaciones de partidos de básquet se extiende para incluir nuevos deportes y múltiples modos de participación, además de estar desarrollado con la idea de realizar una eventual expansión a escala global.

En las secciones siguientes, detallamos los casos de uso de los subsistemas más relevantes, así como un cronograma para el desarrollo de los mismos en múltiples iteraciones. Para la primera iteración en particular, realizamos una descripción más detallada de los casos de uso que la conforman, así como su alcance y su descomposición en una serie de tareas. Cada tarea lleva una estimación de tiempo de desarrollo en horas hombre, concluyendo con un diagrama de Gantt para visualizar su distribución temporal durante el desarrollo del sistema general.

Por último, realizamos un análisis de riesgo sobre el proyecto y sus instancias de desarrollo, detallando la probabilidad, el nivel de impacto y exposición de cada riesgo, así como posibles estrategias de mitigación y un plan de contingencia ante la situación adversa descripta.



## 2. Casos de uso

A continuación incluimos los principales casos de uso cubriendo la mayor parte de la funcionalidad determinada por los requerimientos del proyecto. Los mismos se encuentran ordenados de acuerdo al subsistema al cual corresponden.

### ■ Aplicaciones Web/Móvil.

- Creando nuevo usuario.
- Iniciando sesión de usuario.
- Ingresando datos de tarjeta / cuenta corriente.
- Visualizando desafíos disponibles.
- Dando de alta un nuevo desafío.
- Ingresando a un desafío **como participante**.
- Armand equipo para desafío.
- Visualizando partido de un desafío.
- Visualizando saldo **de usuario**.
- Visualizando premios ganados **del usuario**.

### ■ Servidor de Juego.

- **Guardando datos de nuevo usuario.**
- **Autenticando datos de usuario.**
- **Guardando datos cifrados de tarjeta / cuenta corriente de usuario.**
- Transmitiendo un partido de desafío.
- Creando desafío.
- Cobrando apuestas de desafío.
- Cobrando cuota de participación en desafío.
- Acreditando ganancias por desafío ganado.
- Acreditando premios al usuario.

### ■ Sistema de Transmisión de Partidos.

- Transmitiendo datos de una simulación en proceso.
- Transmitiendo render de una simulación en proceso.
- Transmitiendo un partido en vivo.

### ■ Sistema de Pagos.

- Validando datos de tarjeta / cuenta corriente.
- Debitando monto a una tarjeta / cuenta corriente.
- Acreditando monto a una tarjeta / cuenta corriente.

### 3. Planificación

#### 3.1. Iteraciones

Dividimos los CU en distintas iteraciones, teniendo en cuenta diversos factores. El criterio principal fue seguir un estilo de desarrollo horizontal, en el sentido de desarrollar varios subsistemas en simultáneo, en contraposición a un desarrollo por bloques, en el cual un subsistema no empieza a desarrollarse hasta que gran parte de otro esté completado. Creemos que este estilo nos permitirá poder modificar más fácilmente los componentes, a medida que tengamos un mejor conocimiento sobre las interacciones entre ellos.

Concretamente, en la primera iteración busca desarrollar mínimamente la Aplicación Web/Móvil, el Servidor de Juego y el Sistema de Transmisión de Partidos, y algunas interacciones entre la Aplicación Web/Móvil y el Sistema de Transmisión de Partidos.

En segundo lugar, tuvimos en cuenta el análisis de riesgos realizado, priorizando casos de uso que evalúen o ataquen en alguna medida a estos riesgos, no necesariamente proveyendo una solución. Concretamente, los riesgos encontrados son la confiabilidad de los motores gráficos, la confiabilidad de los clientes (en términos de calidad de hardware y conexión), y la seguridad de los datos almacenados.

En la primera iteración comenzamos a probar el motor gráfico 2D tratando, apuntando a comenzar a evaluar el funcionamiento de dicho motor, y analizamos y desarrollamos estrategias para garantizar la seguridad de los datos de usuarios y desafíos.

##### 3.1.1. Fase de iniciación

**Esta fase se compone del desarrollo realizado en el TP1.**

##### 3.1.2. Fase de elaboración

###### **Primera iteración** [4 semanas]

1. Guardando datos de nuevo usuario.
2. Visualizando partido de un desafío.
3. Transmitiendo datos de una simulación en proceso.
4. Creando desafío.

###### **Segunda iteración** [3 semanas]

1. Iniciando sesión de usuario.
2. Autenticando datos de usuario.
3. Ingresando datos de tarjeta / cuenta corriente.
4. Guardando datos cifrados de tarjeta / cuenta corriente de usuario.

###### **Tercera iteración** [2 semanas]

1. Dando de alta un nuevo desafío.
2. Ingresando a un desafío como participante.
3. Transmitiendo un partido de desafío.
4. Guardando datos de nuevo usuario.
5. Armandando equipo para desafío.

**Cuarta iteración** [4 semanas]

1. Visualizando saldo de usuario.
2. Cobrando cuota de participación en desafío.
3. Acreditando premios al usuario.
4. Visualizando desafíos disponibles.

**Quinta iteración** [3 semanas]

1. Cobrando apuestas de desafío.
2. Visualizando premios ganados del usuario.
3. Transmitiendo render de una simulación en proceso.
4. Validando datos de tarjeta / cuenta corriente.

**Sexta iteración** [4 semanas]

1. Transmitiendo un partido en vivo.
2. Debitando monto a una tarjeta / cuenta corriente.
3. Acreditando monto a una tarjeta / cuenta corriente.
4. Acreditando ganancias por desafío ganado.

**3.1.3. Fase de construcción**

Consiste del desarrollo de cada uno de los casos de uso definidos en la fase de elaboración. Al terminar cada una de las iteraciones, deberíamos tener un software testeable. Al terminar todas las iteraciones, deberíamos tener el software que cumple con todas las especificaciones indicadas por el enunciado y los stakeholders.

**3.1.4. Fase de transición**

Ya con un sistema que tiene todas las características especificadas implementadas, comienza a utilizarse también por usuarios reales. Esto permite detectar problemas y errores, que deben arreglarse en esta fase. Además, se provee la asistencia necesaria para aprender a manejar todos los aspectos del sistema y su mantenimiento.

**3.2. Alcance de casos de usos de la primera iteración****Guardando datos de nuevo usuario.**

El **Servidor de Juego** recibió los datos de un usuario a suscribir, y debe almacenarlos en algún medio físico, garantizando su disponibilidad, y haciéndolo de forma segura.

Los datos de un usuario podrían ser almacenados en un único servidor o en un conjunto de servidores. Dependiendo de la escala que pretenda alcanzar la primer versión del sistema, optaremos por un almacenamiento centralizado o distribuido. Si la **cantidad de usuarios es del orden de los millones, una base de datos relacional y centralizada puede ser suficiente**. Si no, es posible que sea necesaria una base de datos no relacional y distribuida.

La replicación de datos es un factor a considerar, relacionado con la persistencia de estos datos. Observar que la necesidad de replicación depende del volumen de los datos, **sino del volumen de pedidos al servidor**. En efecto, por más que todos los usuarios sean almacenables en forma localizada en una sola

máquina, debemos asegurar que estén disponibles, dada la cantidad de accesos a esa información. Para esto podemos replicar una misma base de datos, **sobre muchas máquinas**.

Con respecto a los datos concretos almacenados, en principio serán sólo los personales, aunque en el futuro también podría utilizarse y guardarse información sobre preferencias y gustos, para la visualización personalizada de publicidad.

Como la seguridad es un atributo de calidad prioritario, debemos almacenar todos estos datos en forma segura, por ejemplo mediante **hashing y salt**, aunque debe invertirse cierto tiempo en estudiar otras alternativas.

### Visualizando partido de un desafío.

La Aplicación Web/Móvil debe mostrar la ejecución de una simulación (si el desafío es de modo simulación) o la transmisión de un partido real (si es de modo fantasía). En la primera iteración, sólo nos interesa visualizar una simulación 2D.

Debe analizarse las tecnologías a utilizar para desarrollar los clientes, dado que para crecer en cantidad de usuarios, necesitamos soportar determinadas plataformas masivamente usadas, como por ejemplo Android.

### Transmitiendo datos de una simulación en proceso.

El Servidor de Juego debe enviar a todos los clientes visualizando cierto desafío, el flujo de datos que produce la simulación asociada. Este envío de datos se da únicamente en el caso que un usuario opta por la renderización 2D, y además su conexión de red no soporta la bajada de un flujo de video, que requiere un gran ancho de banda. En estos casos, los datos enviados corresponden a una descripción del transcurso de la simulación, que es renderizada localmente por el usuario.

En la primera iteración definiremos interfaces sencillas para la comunicación entre un cliente y un servidor. El cliente simplemente pedirá los datos de una simulación, y el servidor transmitirá un flujo de datos correspondiente a una simulación artificial.

### Creando desafío.

El Servidor de Juego recibió un nuevo desafío, y lo debe almacenar en algún medio físico, garantizando su disponibilidad, y haciéndolo de forma segura.

En esta primera iteración nos interesa decidir cuestiones de arquitectura, disponibilidad y seguridad, similares a las dichas en el alcance del CU *Guardando datos de nuevo usuario*.

## 3.3. Tareas CU Primera iteración

A continuación se detallan las tareas diagramadas para los casos de uso incluidos en la primera iteración con su respectiva estimación de horas hombre.

|      |   |            |
|------|---|------------|
| CU1  | Guardando datos de nuevo usuario.   | 114h       |
| T1.1 | Definir motor de base de datos a utilizar, contemplando la escalabilidad. | 8h         |
| T1.2 | Definir método de encriptación para contraseñas y datos sensibles.        | 10h        |
| T1.4 | Definir conjunto de datos a persistir por cada usuario.                   | <b>16h</b> |
| T1.5 | Diseñar base de datos de usuarios.  | 10h        |
| T1.6 | <b>Implementación de CU1</b>  | 40h        |
| T1.7 | Testing de CU1  | 30h        |
| CU2  | Visualizando partido de un desafío.                                       | 161h       |
| T2.1 | Investigar tecnologías web a utilizar para el cliente web.                | 16h        |
| T2.2 | Investigar tecnologías a utilizar para los clientes móviles.              | 16h        |
| T2.3 | <b>Desarrollar stubs de las aplicaciones web y móviles.</b>               | 8h         |
| T2.4 | Integrar motor de gráficos 2D a la aplicación web.                        | 18h        |
| T2.5 | Integrar motor de gráficos 2D a la aplicación móvil.                      | 18h        |
| T2.6 | Implementación de una visualización de partido de prueba.                 | 45h        |
| T2.7 | Testing de CU2  | 40h        |

|      |   |      |
|------|---|------|
| CU3  | Transmitiendo datos de una simulación en proceso.                         | 103h |
| T3.1 | Investigar plataforma para el servidor de transmisión.                    | 10h  |
| T3.2 | Definir la interfaz de comunicación con el servidor.                      | 10h  |
| T3.3 | Desarrollar stub del servidor.  | 8h   |
| T3.4 | Implementar transmisión de datos de una simulación de prueba.             | 35h  |
| T3.5 | Desarrollar cliente simple de prueba.                                     | 10h  |
| T3.6 | Testing de CU3  | 30h  |
| CU4  | Creando desafío.  | 99h  |
| T4.1 | Definir motor de base de datos a utilizar, contemplando la escalabilidad. | 8h   |
| T4.2 | Definir conjunto de datos a persistir por cada desafío.                   | 10h  |
| T4.3 | Investigar alternativas para la persistencia segura de datos del desafío. | 10h  |
| T4.4 | Diseñar base de datos de desafíos.  | 16h  |
| T4.5 | Implementación de CU4   | 30h  |
| T4.6 | Testing de CU4  | 25h  |

### 3.4. Dependencias entre tareas

Las tareas involucradas en cada caso de uso no son necesariamente independientes, en el sentido de que alguna de ellas puede requerir que otra sea finalizada para comenzar con su desarrollo. En los siguientes diagramas, incluimos una representación gráfica de las dependencias entre las diferentes tareas para cada caso de uso.

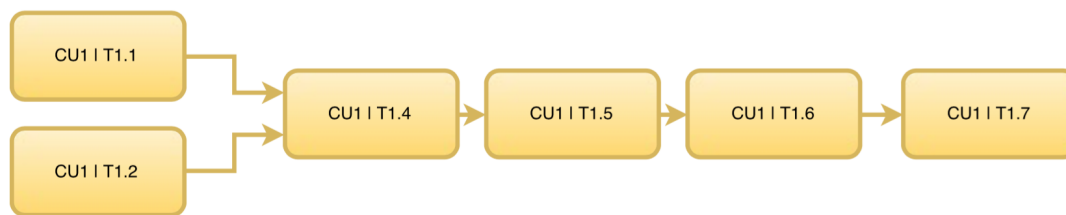


Figura 1: Diagrama de dependencias entre las tareas del CU nro 1.

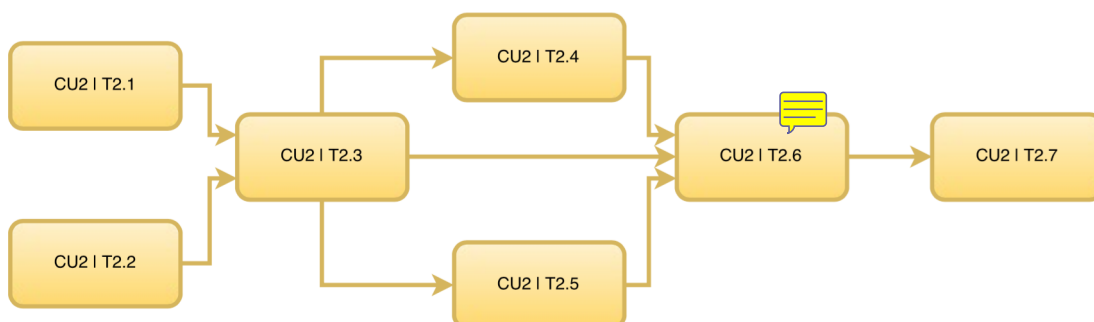


Figura 2: Diagrama de dependencias entre las tareas del CU nro 2.



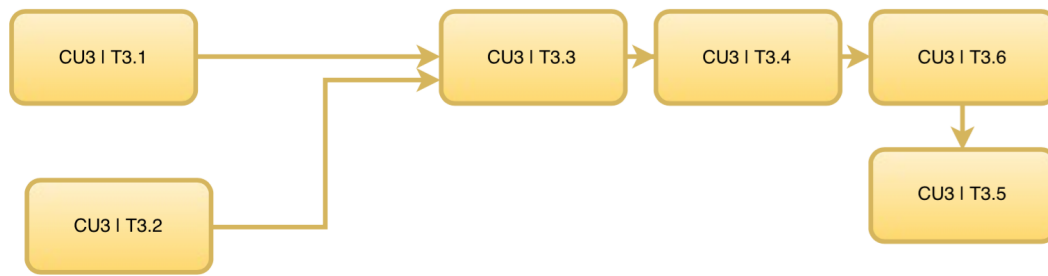


Figura 3: Diagrama de dependencias entre las tareas del CU nro 3.

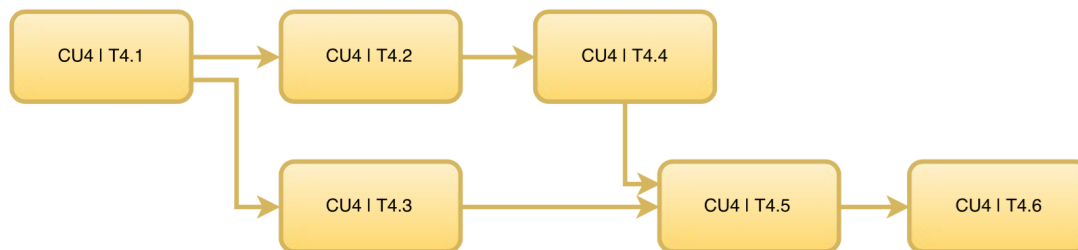


Figura 4: Diagrama de dependencias entre las tareas del CU nro 4.

### 3.5. Detalle Primera iteración

- **Identificación:** E1
- **Tipo de iteración:** Elaboración
- **Cantidad total de horas:** 480
- **Tareas:**
  1. Definir motor de base de datos a utilizar. (8h)
  2. Definir método de encriptación para contraseñas y datos sensibles. (10h)
  3. Definir conjunto de datos a persistir por cada usuario. (16h)
  4. Diseñar base de datos de usuarios. (10h)
  5. Implementación de CU1 (40h)
  6. Testing de CU1 (30h)
  7. Investigar tecnologías web a utilizar para el cliente web. (16h)
  8. Investigar tecnologías a utilizar para los clientes móviles. (16h)
  9. Desarrollar stubs de las aplicaciones web y móviles. (8h)
  10. Integrar motor de gráficos 2D a la aplicación web. (18h)

11. Integrar motor de gráficos 2D a la aplicación móvil. (18h)
12. Implementación de una visualización de partido de prueba. (45h)
13. Testing de CU2 (40h)
14. Investigar plataforma para el servidor de transmisión. (10h)
15. Definir la interfaz de comunicación con el servidor. (10h)
16. Desarrollar stub del servidor. (8h)
17. Implementar transmisión de datos de una simulación de prueba. (35h)
18. Desarrollar cliente simple de prueba. (10h)
19. Testing de CU3 (30h)
20. Definir motor de base de datos a utilizar contemplando la escalabilidad. (8h)
21. Definir conjunto de datos a persistir por cada desafío. (10h)
22. Investigar alternativas para la persistencia segura de datos del desafío. (10h)
23. Diseñar base de datos de desafíos. (16h)
24. Implementación de CU4 (30h)
25. Testing de CU4 (25h)



### 3.6. Plan de Proyecto

En el siguiente diagrama detallamos el plan de desarrollo del proyecto, con una planificación estimada del desarrollo de las tareas a realizar en la primera iteración. Para la distribución en tiempo de las mismas, se asignó el tiempo de cuatro desarrolladores con una dedicación de cuatro horas diarias al proyecto.

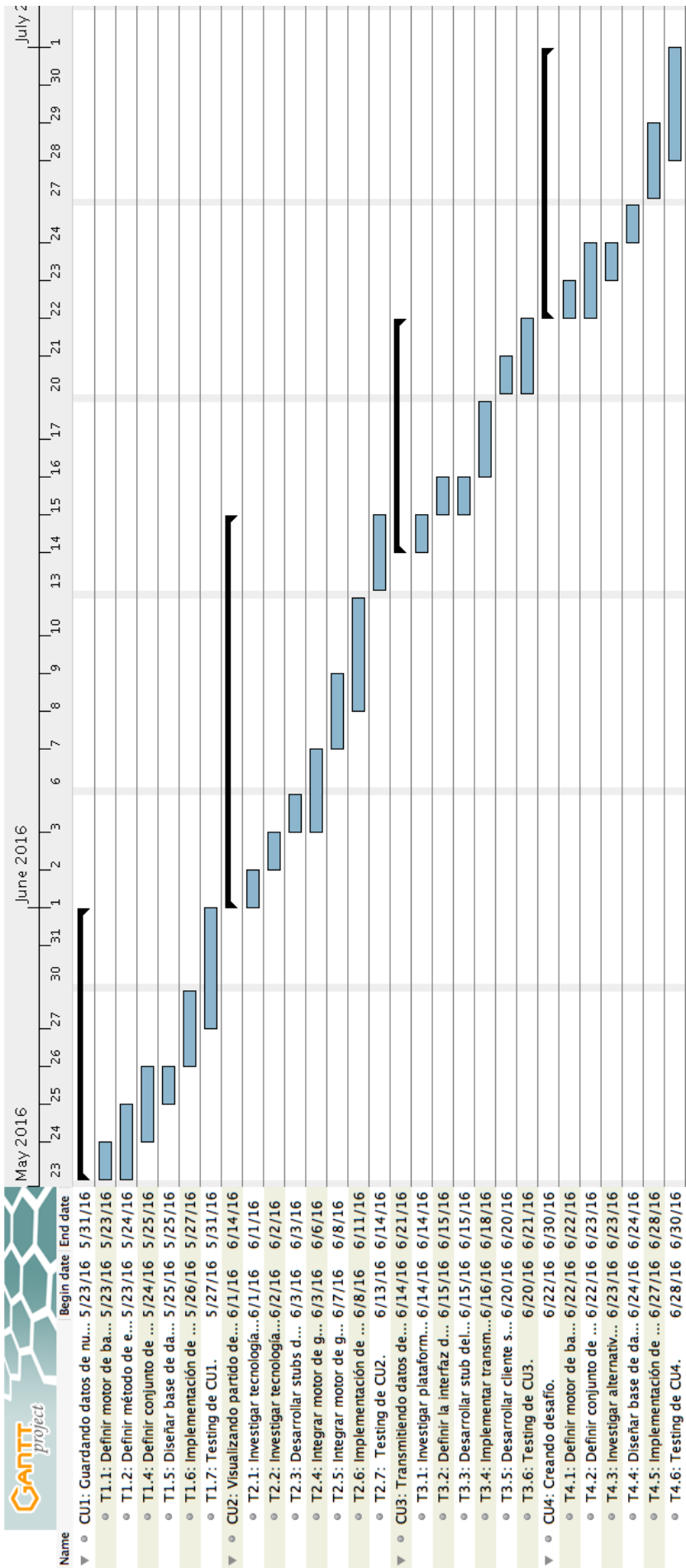


Figura 5: Diagrama de Gantt de la 1era iteración con la asignación de tiempo



## 4. Análisis de riesgos

### Riesgo 1:

- **Descripción:** Los motores gráficos 2D y 3D son desarrollados por terceros. Si alguna de las dos compañías deja de funcionar, perdemos soporte para el correspondiente motor gráfico e incrementa fuertemente el costo y tiempo de desarrollo para realizar mejoras o agregar funcionalidad en el mismo.
- **Probabilidad:** Baja
- **Impacto:** Alto
- **Exposición:** Media
- **Mitigación:** Establecer una capa de abstracción encima de los motores gráficos para que los mismos sean reemplazables sin que sea necesario modificar los demás sistemas.
- **Plan de contingencia:** Licitación el motor gráfico que ha perdido soporte entre nuevas empresas.

### Riesgo 2:

- **Descripción:** Actualmente, los usuarios tienen múltiples modos para visualizar un partido: simulación 2D o 3D a partir del feed de la simulación, streaming de partidos reales y streaming de un render de la simulación realizado del lado del servidor. Si un usuario tiene un dispositivo sin las prestaciones necesarias para realizar una simulación de forma nativa, cada vez que tenga mala conectividad tampoco será capaz de visualizar streamings y no podrá obtener información sobre los partidos de ninguna manera.
- **Probabilidad:** Media
- **Impacto:** Medio
- **Exposición:** Media
- **Mitigación:** Desarrollar un feed de información acerca del partido en ejecución que sea apto para clientes con bajos recursos de hardware y mala conectividad.
- **Plan de contingencia:** Proponer al usuario afectado la visualización del feed del partido basado en texto.

### Riesgo 3:

- **Descripción:** El sistema en su conjunto procesa varios conjuntos de datos altamente sensibles que refieren a la identidad de sus usuarios, a sus medios de pago, y la realización de las simulaciones que determinan los desafíos para bajo apuestas o con premios, así como los resultados de los mismos. Si alguno de estos sistemas es comprometido, la integridad personal o económica de los usuarios o del sistema mismo pueden sufrir consecuencias irreversibles.
- **Probabilidad:** Media
- **Impacto:** Alto
- **Exposición:** Media
- **Mitigación:** Investigar fuertemente las mejores prácticas en seguridad para el desarrollo de los sistemas que procesan información sensible.
- **Plan de contingencia:** **Suspensión inmediata de actividades en la plataforma** con posibles resultados irreversibles (pagos, entrega de premios) e investigación sobre el siniestro cooperando con las autoridades.