



Trabajo Práctico de Entrega Obligatoria N°1 – Entrega 2

Comisión N°3

AGRA, FEDERICO GERMÁN (94186)

DÍAZ FIGUEROA, RODRIGO MARTIN (97400)

Índice

Enunciado General del Proyecto -----	2
Requerimientos -----	4
Listado de Requerimientos -----	4
Especificación de Requerimientos -----	9
Riesgos -----	20
Descripción de Riesgos -----	20
Clasificación de riesgos -----	22
Planes de Reducción, Supervisión y Gestión de Riesgos -----	23
Estructura de Descomposición del Trabajo -----	24
Listado de Actividades -----	25
Planificación -----	26
Diagrama de Gantt -----	26
Diagrama de Red -----	27

Enunciado General del Proyecto

Oportunidad:

Se ha estudiado la gran repercusión y la popularidad que han logrado las series de televisión en la actualidad, y particularmente *Los Simpsons* de la cadena FOX, la cual es la serie animada con mayor audiencia de la televisión mundial. A partir de este estudio se detectó una carencia importante en materia de videojuegos de modalidad *Tower Defense* con temática en dicha serie, lo cual permitiría atraer a su público seguidor y, mediante descargas y posibles microtransacciones, capitalizar el éxito de *Los Simpsons* en favor de nuestro producto.

Nombre del proyecto:

The Simpsons: Tower Defense (LS).

Código:

#00001

Fecha de propuesta:

2 de Septiembre de 2017.

Fecha de aceptación:**Responsables:**

- Agra, Federico Germán (94186).
- Díaz Figueroa, Rodrigo Martin (97400).

Breve descripción:

Se realizará un proceso de desarrollo de software durante cuatro meses para la construcción de un juego multiplataforma de modalidad *Tower Defense*. Estará ambientado en una popular serie animada de televisión llamada *Los Simpsons*, pero también será reutilizable para ser modelado por cualquier otra ambientación.

Meta:

Finalizar exitosamente el desarrollo de LS. Lograr un producto que permita modos de partidas de un jugador y online, y que rinda beneficios económicos de manera periódica, mediante compras electrónicas de artículos.

Objetivos:

- Obtener un producto eficiente en cuanto a recursos y almacenamiento.
- LS debe ser una aplicación multiplataforma (*Windows, Linux y Android*).
- Lograr que LS soporte múltiples lenguajes (español, inglés, alemán, portugués y chino simplificado).
- Incorporar microtransacciones para compras de artículos exclusivos.
- Lograr una ambientación acorde a la serie y obtener la aceptación de un gran porcentaje del público aficionado de *Los Simpsons* (60% de opiniones positivas con respecto a la audiencia total de la serie animada).
- Lograr un alto porcentaje de aceptación por parte de los jugadores con respecto a la facilidad de uso del juego mediante encuestas y opiniones.

<ul style="list-style-type: none"> - Incorporar niveles para modo de un solo jugador, así como también competencias y torneos oficiales entre jugadores registrados para fomentar la modalidad de juego online. - <i>LS</i> debe poder amoldarse fácilmente a otros ambientes, para luego implementarlo con otras series de la cadena, o bien, venderse para otros contextos totalmente diferentes. 	
Factores críticos de éxito: <ul style="list-style-type: none"> - <i>LS</i> debe inicializarse en menos de un minuto en un ordenador con <i>Windows 7</i>, 2 GB de RAM y un procesador <i>dual-core</i>. - <i>LS</i> debe inicializarse en menos de treinta segundos en un ordenador con sistema operativo basado en <i>Linux</i>, 2 GB de RAM y un procesador <i>dual-core</i>. - <i>LS</i> no debe ocupar más de 200 MB de almacenamiento del ordenador. - Se verificará que para su lanzamiento <i>LS</i> soporta los idiomas español, inglés y chino simplificado. - La cantidad de descargas de <i>LS</i> deben ser de al menos del 50% del total de audiencia de la serie animada <i>Los Simpsons</i> en el plazo de los primeros tres meses. - La recaudación por medio de las compras de artículos en el primer semestre debe ser de al menos un 10% más que la cantidad total de descargas. - Se deberá terminar el desarrollo de <i>LS</i> dentro del plazo establecido, ya que el retrasarse le daría una ventaja importante a la competencia. 	
Presunciones, riesgos y obstáculos: <ul style="list-style-type: none"> - Por la gran popularidad de la serie animada <i>Los Simpsons</i>, prevemos que <i>LS</i> tendrá una cantidad de al menos 10.000 descargas en el plazo del primer mes posterior al lanzamiento. - Al preverse una gran cantidad de descargas, podrían colapsarse los servidores de <i>LS</i> debido al ingreso masivo de jugadores registrados. 	
Firma de Recepción de la Propuesta: Firma Aceptación/Rechazo:	Estado: Propuesta Recibida <input type="checkbox"/> Propuesta Aceptada <input type="checkbox"/> Propuesta Rechazada <input type="checkbox"/>

Listado de Requerimientos

Requerimientos Funcionales

Relacionados a la Aplicación principal:

- **RF1: Iniciar Nivel:** El sistema debe iniciar un nivel predeterminado para comenzar con el juego. Debe dibujar el mapa según una configuración precargada para darle la opción al usuario de armar su estrategia. El nivel se constará de tres hordas enemigas, las cuales si el usuario logra sobrevivir a dichas hordas enemigas se dice que el nivel ha sido superado. Deberán existir al menos dos niveles predefinidos.
- **RF2: Reiniciar Nivel:** El sistema debe reiniciar el nivel. Esta tarea consta de vaciar la grilla (mapa) de unidades y objetos permitiendo al usuario volver a jugar el mismo nivel.
- **RF3: Finalizar Nivel Completado:** El sistema debe finalizar el nivel satisfactoriamente para luego cargar el próximo nivel.
- **RF4: Finalizar Nivel Perdido:** El sistema debe finalizar el nivel, resetear la puntuación del usuario y reiniciar el nivel.
- **RF5: Proveer Unidad Aliada:** El sistema debe contar con 5 tipos de unidades aliadas, las cuales serán utilizadas por el usuario para defender la casa de las hordas enemigas. La aparición de las mismas será por medio del usuario. Todas las unidades aliadas contienen:
 - Cantidad de vida máxima.
 - Poder de ataque.
 - Rango de alcance.
 - Costo.
- **RF5.1: Proveer Unidad Bart:** Vida: 20; Ataque: 5; Rango: 3; Costo: 10.
- **RF5.2: Proveer Unidad Homero:** Vida: 25; Ataque: 4; Rango: 3; Costo: 10.
- **RF5.3: Proveer Unidad Lisa:** Vida: 50; Ataque: 10; Rango: 3; Costo: 25.
- **RF5.4: Proveer Unidad Marge:** Ocupa dos cuadros de alto. Vida: 80; Ataque: 20; Rango: 2; Costo: 50.

- **RF5.5: Proveer Unidad Maggie:** Vida: 100; Ataque: 35; Rango: 3; Costo: 100.

- **RF6: Proveer Unidad Enemiga:** El sistema debe contar con 6 tipos de unidades enemigas, las cuales atacarán a las unidades elegidas por el usuario. Estas unidades aparecerán de manera automática y en tandas de cierta cantidad de unidades. Estas unidades en ocasiones pueden aparecer con un campo de fuerza a su favor. Todas las unidades enemigas contienen:
 - Cantidad de vida máxima.
 - Poder de ataque.
 - Rango de alcance.
 - Velocidad de desplazamiento.
 - Mínimo/Máximo de monedas.

- **RF6.1: Proveer Unidad Perro de Burns:** Vida: 10; Ataque: 5; Rango: 3; Velocidad: 4; Monedas Mínimas: 1; Monedas Máximas: 3.

- **RF6.2: Proveer Unidad Loca de los Gatos:** Vida: 20; Ataque: 7; Rango: 3; Velocidad: 3,5; Monedas Mínimas: 2; Monedas Máximas: 4.

- **RF6.3: Proveer Unidad Smithers:** Vida: 30; Ataque: 10; Rango: 3; Velocidad: 4; Monedas Mínimas: 3; Monedas Máximas: 6.

- **RF6.4: Proveer Unidad Skinner:** Vida: 40; Ataque: 15; Rango: 4; Velocidad: 2,5; Monedas Mínimas: 5; Monedas Máximas: 6.

- **RF6.5: Proveer Unidad Jeff:** Vida: 50; Ataque: 20; Rango: 4; Velocidad: 2; Monedas Mínimas: 7; Monedas Máximas: 10.

- **RF6.6: Proveer Unidad Mr. Burns:** Vida: 60; Ataque: 25; Rango: 5; Velocidad: 1; Monedas Mínimas: 12; Monedas Máximas: 15.

- **RF7: Proveer Horda de enemigos:** El sistema debe ser capaz de enviar hordas enemigas para atacar a la defensa del usuario. Estas constarán de entre 10 y 30 unidades enemigas dependiendo que número de horda que sea.

- **RF8: Proveer Premio:** Los enemigos, al ser destruidos, dejan premios (monedas y/u objetos) los cuales el usuario deberá poder recolectar. También pueden aparecer de manera espontánea y al azar en favor del usuario.
 - **RF8.1: Proveer Premio Campo de Protección:** Ataca a cualquier unidad contraria que se encuentre dentro del campo.

- **RF8.2: Proveer Premio Bomba:** Puede ser colocada en cualquier celda del mapa y ataca a todas las unidades (aliadas y enemigas) en un radio de una celda a la redonda.
- **RF8.3: Proveer Premio Magia Ataque:** Aumenta el ataque de todas las unidades aliadas por un tiempo determinado.
- **RF8.4: Proveer Premio Magia Rango:** Aumenta el rango de alcance de todas las unidades aliadas por un tiempo determinado.
- **RF8.5: Proveer Premio Curación:** Recupera algunos puntos de vida a todas las unidades aliadas.
- **RF9: Proveer Objeto Comprable:** El sistema debe contar con cuatro tipos de objetos a disposición del usuario para comprar. Estos objetos se diferencian en dos tipos: Con vida o por tiempo.
 - **RF9.1: Proveer Objeto con vida:** Estos objetos contienen una vida máxima que determina la durabilidad de su beneficio.
 - **RF9.1.1: Proveer Objeto Bola de Nieve:** Vida: 20; Costo: 30.
 - **RF9.1.2: Proveer Objeto Ayudante de Santa:** Vida: 40; Costo: 60.
 - **RF9.2: Proveer Objeto por tiempo:** El beneficio que proveen estos objetos perduran un tiempo determinado.
 - **RF9.1.1: Proveer Objeto Milhouse:** Tiempo de acción: 10 segundos; Costo: 25.
 - **RF9.1.2: Proveer Objeto Flanders:** Tiempo de acción: 20 segundos; Costo: 75.
- **RF10: Proveer Objeto Mapa:** En el mapa se deben colocar aleatoriamente objetos que obstaculicen o dificulten el avance de las unidades tanto aliadas como enemigas. Estos objetos no son comprables por el usuario. Se distinguen dos tipos: Muro y Lodo.
 - **RF10.1: Proveer Objeto Muro:** Objeto que obstaculiza el paso de cualquier unidad que pretende sobrepasarlo. Vida: 20.
 - **RF10.2: Proveer Objeto Lodo:** Objeto temporal que dificulta el paso sobre él. Reduce la velocidad de las unidades enemigas que pasan sobre él. Tiempo: 10 segundos.

- **RF11: Ubicar Elemento en Celda:** Se debe poder ubicar una unidad u objeto en una determinada celda.
- **RF12: Disparar:** Algunas unidades tanto aliadas como las enemigas tendrán la capacidad de realizar disparos para atacar a su contrincante. Esta función debe realizar un decremento en la vida de las unidades contrarias impactadas.
- **RF13: Destruir Unidad Aliada:** Se debe poder eliminar una unidad aliada al ser destruida por un enemigo o alcance de bomba. La celda deberá quedar vacía luego de la eliminación, disponible para ser ocupada por una nueva unidad o bien por algún objeto provisto al azar por el mapa.
- **RF14: Destruir Unidad Enemiga:** El sistema debe poder eliminar una unidad enemiga y actualizar el puntaje del usuario. Una vez eliminada la unidad, en la celda quedarán la/s moneda/s que la unidad contenga y, en ocasiones, un premio.
- **RF15: Recolectar Premio:** El usuario debe poder “tocar” sobre un premio en el mapa y recolectarlo para utilizarlo a su favor cuando lo requiera.
- **RF16: Realizar Compra Online:** El sistema debe proveer la opción de compras de artículos (monedas o premios) mediante internet. Se contará con una tienda con la cual el usuario podrá realizar compras de distintos artículos mediante su tarjeta de crédito.

Relacionados a la GUI de la Aplicación principal:

- **RF17: Mostrar Barra de Menú:** Como parte de la GUI de la aplicación, se mostrará una barra de menú con la cual el usuario podrá seleccionar diferentes opciones.
 - **RF17.1: Mostrar Opciones Juego:** Como parte de la Barra de Menú se contará con una pestaña para mostrar las opciones de las acciones que se pueden realizar sobre el juego. Dentro de esta pestaña se identificarán las siguientes opciones:
 - Iniciar nivel: Inicia el primer nivel.
 - Reiniciar nivel: Reinicia el nivel actual.
 - Ajustes: Abre las configuraciones del juego.
 - Salir: Sale del juego.

- **RF17.2: Mostrar Opciones Ayuda (Instrucciones de Uso, Acerca de):** Como parte de la Barra de Menú se contará con una pestaña para mostrar las opciones de ayuda del juego. Estas opciones son:
 - Instrucciones de Uso: Muestra información útil para el uso de la aplicación.
 - Acerca de: Muestra información propia del juego.
- **RF18: Mostrar Botón Añadir Unidad:** Como parte de la GUI de la aplicación, se mostrará un botón para añadir (comprar) una nueva unidad aliada. En esta sección el usuario debe poder elegir un tipo de unidad aliada que desea agregar a su defensa y luego con dicho botón poder comprarla para luego colocarla en el mapa. Luego de realizar la compra se deben decrementar las monedas del usuario dependiendo del costo de la unidad.
- **RF19: Mostrar Botón Comprar Objetos:** La GUI debe contar con un botón que despliegue una lista de objetos disponibles para compras. Luego de realizar la compra se deben decrementar las monedas del usuario dependiendo del costo del objeto adquirido.
- **RF20: Mostrar Botón Vender Unidad:** La GUI debe contar con un botón que permita seleccionar una unidad aliada del mapa y venderla para obtener monedas a cambio. Luego de realizar la venta se deben incrementar las monedas del usuario dependiendo del costo de la unidad, siendo la mitad del costo si la unidad no posee el total de su vida.
- **RF21: Mostrar Estado Usuario:** Como parte de la GUI se mostrará una etiqueta que informe al usuario sobre la puntuación y las monedas que posee actualmente. En esta sección se encontrará la siguiente información:
 - Puntuación: Puntaje actual.
 - Monedas: Cantidad actual de monedas obtenidas.

Requerimientos No Funcionales

- **RNF1: Soportar ejecución en múltiples plataformas:** El sistema debe poder ser ejecutado en múltiples plataformas. Estas serán Windows (Vista, 7, 8, 10), Linux (Distribuciones basadas en Debian) y Android (5.0 o más).
- **RNF2: Soportar múltiples Idiomas:** El sistema debe poderse configurar en distintos idiomas. Los idiomas que deben ser soportados son:
 - Español.
 - Inglés.
 - Alemán.
 - Portugués.
 - Chino Simplificado.

- **RNF3: Facilidad de uso (Usabilidad):** El juego debe ser sencillo y fácil de usar, de manera que el aprendizaje en el uso del mismo no necesite de una previa capacitación.
- **RNF4: Bajo uso de Almacenamiento (Eficiencia):** La aplicación de escritorio debe ocupar menos de 200 Mb de disco (instalado). Mientras que la versión móvil no debe superar los 50 Mb.
- **RNF5: Fluidez (Eficiencia):** El juego debe ser sencillo y fácil de usar, de manera que el aprendizaje en el uso del mismo no necesite de una previa capacitación.

Especificación de Requerimientos

Requerimientos Funcionales

Nombre	Proveer Unidad Aliada.
Id	RF5
Categoría	Funcional
Descripción	<p>El sistema debe contar con 5 tipos de unidades aliadas, las cuales serán utilizadas por el usuario para defender la casa de las unidades enemigas. Todas las unidades aliadas contienen una cantidad de vida máxima, un poder de ataque, un rango de alcance de ataque y un costo. Estos atributos varían según el tipo de unidad aliada.</p> <p><i>Bart:</i></p> <ul style="list-style-type: none"> - Vida: 20 - Ataque: 5 - Rango: 3 - Costo: 10 <p><i>Homero:</i></p> <ul style="list-style-type: none"> - Vida: 25 - Ataque: 4 - Rango: 3 - Costo: 10 <p><i>Lisa:</i></p> <ul style="list-style-type: none"> - Vida: 50 - Ataque: 10 - Rango: 3 - Costo: 25 <p><i>Marge:</i></p> <ul style="list-style-type: none"> - Vida: 80 - Ataque: 20

	<ul style="list-style-type: none"> - <i>Rango: 2</i> - <i>Costo: 50</i> <p><i>Maggie:</i></p> <ul style="list-style-type: none"> - <i>Vida: 100</i> - <i>Ataque: 35</i> - <i>Rango: 3</i> - <i>Costo: 100</i>
Términos	Vida, Ataque, Rango, Costo, Bart, Homero, Lisa, Marge, Maggie, Aliado, Usuario.
Justificación	Permite que cada unidad aliada tenga distintos atributos.
Prioridad	Alta
Dependencias	RF1
Documentos	
Argumentos de factibilidad	Es posible tener distintos tipos de unidades aliadas dado que la aplicación brinda el soporte para esto.
Método de verificación	El usuario puede seleccionar una unidad de cualquier tipo y agregarla al mapa.

Nombre	Proveer Unidad Bart.
Id	RF5.1
Categoría	Funcional
Descripción	<p><i>Bart:</i></p> <ul style="list-style-type: none"> - <i>Vida: 20</i> - <i>Ataque: 5</i> - <i>Rango: 3</i> - <i>Costo: 10</i>
Términos	Vida, Ataque, Rango, Costo, Bart, Aliado, Usuario.
Justificación	Permite añadir una unidad del tipo Bart.
Prioridad	Alta
Dependencias	RF1, RF5
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para crear esta unidad.
Método de verificación	El usuario puede seleccionar una unidad del tipo Bart y agregarla al mapa.

Nombre	Proveer Unidad Homero.
Id	RF5.2
Categoría	Funcional
Descripción	<p><i>Homero:</i></p> <ul style="list-style-type: none"> - <i>Vida: 25</i> - <i>Ataque: 4</i> - <i>Rango: 3</i> - <i>Costo: 10</i>
Términos	Vida, Ataque, Rango, Costo, Homero, Aliado, Usuario.

Justificación	Permite añadir una unidad del tipo Homero.
Prioridad	Alta
Dependencias	RF1, RF5
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para crear esta unidad.
Método de verificación	El usuario puede seleccionar una unidad del tipo Homero y agregarla al mapa.

Nombre	Proveer Unidad Lisa.
Id	RF5.3
Categoría	Funcional
Descripción	<i>Lisa:</i> <ul style="list-style-type: none"> - <i>Vida: 50</i> - <i>Ataque: 10</i> - <i>Rango: 3</i> - <i>Costo: 25</i>
Términos	Vida, Ataque, Rango, Costo, Lisa, Aliado, Usuario.
Justificación	Permite añadir una unidad del tipo Lisa.
Prioridad	Alta
Dependencias	RF1, RF5
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para crear esta unidad.
Método de verificación	El usuario puede seleccionar una unidad del tipo Lisa y agregarla al mapa.

Nombre	Proveer Unidad Marge.
Id	RF5.4
Categoría	Funcional
Descripción	<i>Marge:</i> <ul style="list-style-type: none"> - <i>Vida: 80</i> - <i>Ataque: 20</i> - <i>Rango: 2</i> - <i>Costo: 50</i>
Términos	Vida, Ataque, Rango, Costo, Marge, Aliado, Usuario.
Justificación	Permite añadir una unidad del tipo Marge.
Prioridad	Alta
Dependencias	RF1, RF5
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para crear esta unidad.
Método de verificación	El usuario puede seleccionar una unidad del tipo Marge y agregarla al mapa.

Nombre	Proveer Unidad Maggie.
Id	RF5.5
Categoría	Funcional
Descripción	<p><i>Maggie:</i></p> <ul style="list-style-type: none"> - <i>Vida: 100</i> - <i>Ataque: 35</i> - <i>Rango: 3</i> - <i>Costo: 100</i>
Términos	Vida, Ataque, Rango, Costo, Maggie, Aliado, Usuario.
Justificación	Permite añadir una unidad del tipo Maggie.
Prioridad	Alta
Dependencias	RF1, RF5
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para crear esta unidad.
Método de verificación	El usuario puede seleccionar una unidad del tipo Maggie y agregarla al mapa.

Nombre	Proveer Unidad Enemiga.
Id	RF6
Categoría	Funcional
Descripción	<p>El sistema debe contar con 6 tipos de unidades enemigas, las cuales atacarán a las unidades elegidas por el usuario. Todas las unidades enemigas contienen una cantidad de vida máxima, un poder de ataque, un rango de alcance de ataque, una velocidad y un mínimo y máximo de monedas que puede contener. Estos atributos varían según el tipo de unidad enemiga.</p> <p><i>Perros de Burns:</i></p> <ul style="list-style-type: none"> - <i>Vida: 10</i> - <i>Ataque: 5</i> - <i>Rango: 3</i> - <i>Velocidad: 4</i> - <i>Monedas Mínimas: 1</i> - <i>Monedas Máximas: 3</i> <p><i>Loca de los Gatos:</i></p> <ul style="list-style-type: none"> - <i>Vida: 20</i> - <i>Ataque: 7</i> - <i>Rango: 3</i> - <i>Velocidad: 3,5</i> - <i>Monedas Mínimas: 2</i> - <i>Monedas Máximas: 4</i>

	<p><i>Smithers:</i></p> <ul style="list-style-type: none"> - Vida: 30 - Ataque: 10 - Rango: 3 - Velocidad: 4 - Monedas Mínimas: 3 - Monedas Máximas: 6 <p><i>Skinner:</i></p> <ul style="list-style-type: none"> - Vida: 40 - Ataque: 15 - Rango: 4 - Velocidad: 2,5 - Monedas Mínimas: 5 - Monedas Máximas: 6 <p><i>Jeff:</i></p> <ul style="list-style-type: none"> - Vida: 50 - Ataque: 20 - Rango: 4 - Velocidad: 2 - Monedas Mínimas: 7 - Monedas Máximas: 10 <p><i>Mr. Burns:</i></p> <ul style="list-style-type: none"> - Vida: 60 - Ataque: 25 - Rango: 5 - Velocidad: 1 - Monedas Mínimas: 12 - Monedas Máximas: 15
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Perros de Burns, Skinner, Loca de los Gatos, Smithers, Jeff, Mr. Burns, Enemigo, Usuario.
Justificación	Permite que cada unidad enemiga tenga distintos atributos.
Prioridad	Alta
Dependencias	RF1
Documentos	
Argumentos de factibilidad	Es posible tener distintos tipos de unidades enemigas dado que la aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir aleatoriamente una unidad enemiga de cualquier tipo al mapa.

Nombre	Proveer Unidad Perro de Burns.
Id	RF6.1
Categoría	Funcional

Descripción	<i>Perro de Burns:</i> <ul style="list-style-type: none"> - <i>Vida: 10</i> - <i>Ataque: 5</i> - <i>Rango: 3</i> - <i>Velocidad: 4</i> - <i>Monedas Mínimas: 1</i> - <i>Monedas Máximas: 3</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Perro de Burns, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Perro de Burns y visualizarse en el mapa.

Nombre	Proveer Unidad Loca de los Gatos.
Id	RF6.2
Categoría	Funcional
Descripción	<i>Loca de los Gatos:</i> <ul style="list-style-type: none"> - <i>Vida: 20</i> - <i>Ataque: 7</i> - <i>Rango: 3</i> - <i>Velocidad: 3,5</i> - <i>Monedas Mínimas: 2</i> - <i>Monedas Máximas: 4</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Loca de los Gatos, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Loca de los Gatos y visualizarse en el mapa.

Nombre	Proveer Unidad Smithers.
Id	RF6.3
Categoría	Funcional
Descripción	<i>Smithers:</i> <ul style="list-style-type: none"> - <i>Vida: 30</i> - <i>Ataque: 10</i>

	<ul style="list-style-type: none"> - <i>Rango: 3</i> - <i>Velocidad: 4</i> - <i>Monedas Mínimas: 3</i> - <i>Monedas Máximas: 6</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Smithers, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Smithers y visualizarse en el mapa.

Nombre	Proveer Unidad Skinner.
Id	RF6.4
Categoría	Funcional
Descripción	<i>Skinner:</i> <ul style="list-style-type: none"> - <i>Vida: 40</i> - <i>Ataque: 15</i> - <i>Rango: 4</i> - <i>Velocidad: 2,5</i> - <i>Monedas Mínimas: 5</i> - <i>Monedas Máximas: 6</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Skinner, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Skinner y visualizarse en el mapa.

Nombre	Proveer Unidad Jeff.
Id	RF6.5
Categoría	Funcional

Descripción	<i>Jeff:</i> <ul style="list-style-type: none"> - <i>Vida: 50</i> - <i>Ataque: 20</i> - <i>Rango: 4</i> - <i>Velocidad: 2</i> - <i>Monedas Mínimas: 7</i> - <i>Monedas Máximas: 10</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Jeff, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Jeff y visualizarse en el mapa.

Nombre	Proveer Unidad Mr. Burns.
Id	RF6.6
Categoría	Funcional
Descripción	<i>Mr. Burns:</i> <ul style="list-style-type: none"> - <i>Vida: 60</i> - <i>Ataque: 25</i> - <i>Rango: 5</i> - <i>Velocidad: 1</i> - <i>Monedas Mínimas: 12</i> - <i>Monedas Máximas: 15</i>
Términos	Vida, Ataque, Rango, Velocidad, Monedas Mínimas, Monedas Máximas, Mr. Burns, Enemigo, Usuario.
Justificación	El sistema debe poder agregar una unidad enemiga.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	La aplicación brinda el soporte para esto.
Método de verificación	El sistema puede añadir una unidad enemiga del tipo Mr. Burns y visualizarse en el mapa.

Nombre	Ubicar Elemento en Celda.
Id	RF11
Categoría	Funcional
Descripción	Se debe poder ubicar una unidad aliada u objeto en una determinada celda del mapa.
Términos	Celda, Unidad aliada, Objeto.

Justificación	El usuario puede ubicar una unidad aliada u objeto específico en la celda que deseada.
Prioridad	Media
Dependencias	RF1, RF5, RF6, RF8, RF9, RF10
Documentos	
Argumentos de factibilidad	Es posible llevarlo a cabo ya que se conocen todos los elementos que pueden verse implicados en la configuración de la defensa del usuario, así como también la grilla en la que se los ubicará.
Método de verificación	Se ubicará exitosamente el elemento seleccionado en la celda deseada.

Nombre	Destruir Unidad Enemiga.
Id	RF14
Categoría	Funcional
Descripción	El sistema debe poder eliminar una unidad enemiga y actualizar el puntaje del usuario. Una vez eliminada la unidad, en la celda quedarán la/s moneda/s que la unidad contenga, y en ocasiones un premio.
Términos	Unidad aliada, Unidad enemiga. Objeto, Premio, Moneda, Celda.
Justificación	Cuando una unidad aliada destruye a una unidad contraria debe ser eliminado del mapa.
Prioridad	Alta
Dependencias	RF1, RF6
Documentos	
Argumentos de factibilidad	Es posible llevarlo a cabo ya que se cuenta con la información correspondiente al estado del usuario.
Método de verificación	Se eliminará exitosamente el enemigo del mapa al ser destruido y se actualizará correctamente la puntuación del usuario.

Nombre	Mostrar Estado Usuario
Id	RF21
Categoría	
Descripción	Como parte de la GUI se mostrará una etiqueta que informe al usuario sobre la puntuación y las monedas que posee actualmente. En esta sección se encontrará la siguiente información: <ul style="list-style-type: none"> - Puntuación: Puntaje actual. - Monedas: Cantidad actual de monedas obtenidas.
Términos	GUI, Puntuación, Monedas.
Justificación	Se requiere mantener información visible del estado de la partida del usuario.

Prioridad	Baja
Dependencias	RF1
Documentos	
Argumentos de factibilidad	Es posible ya que se mantiene la información tanto del puntaje actual como de la cantidad de monedas disponibles.
Método de verificación	Inicializar una partida y verificar que tanto la puntuación actual y la cantidad de monedas se inicializan en cero. Luego de destruir a un enemigo y recolectando las monedas que deja caer, verificar si se actualiza el estado de la partida.

Requerimientos No Funcionales

Nombre	Soportar ejecución en múltiples plataformas (Portabilidad).
Id	RNF1
Categoría	No Funcional
Descripción	El sistema debe poder ser ejecutado en múltiples plataformas. Estas serán Windows (Vista, 7, 8, 10), Linux (Distribuciones basadas en Debian) y Android (5.0 o más).
Escala	Cantidad de plataformas soportadas.
Test	El sistema se ejecutará en distintas plataformas verificando su correcto funcionamiento.
Peor caso	El sistema es soportado por una plataforma (Windows).
Nivel planificado	El sistema es soportado por tres plataformas (Windows, Linux, Android).
Mejor caso	El sistema se ejecuta correctamente en las tres plataformas.
Nivel actual	

Nombre	Soportar múltiples idiomas.
Id	RNF2
Categoría	No Funcional
Descripción	El sistema debe poderse configurar en distintos idiomas. Los idiomas que deben ser soportados son: <ul style="list-style-type: none"> - Español. - Inglés. - Alemán. - Portugués. - Chino simplificado.
Escala	Cantidad de idiomas que soporta.
Test	Se configurará el juego en todos los idiomas disponibles (Español, Inglés, Alemán, Portugués y Chino simplificado) y se verifica su correcto funcionamiento.

Peor caso	Que el sistema esté disponible solo en un idioma.
Nivel planificado	Que el sistema esté disponible en cinco idiomas.
Mejor caso	Que el sistema permita agregar más paquetes de idiomas.
Nivel actual	

Nombre	Facilidad de uso (Usabilidad).
Id	RNF3
Categoría	No Funcional
Descripción	El juego debe ser sencillo y fácil de usar, de manera que el aprendizaje en el uso del mismo no necesite de una previa capacitación.
Escala	Porcentaje de personas que aprenden a utilizar la aplicación en menos de 10 minutos.
Test	Se realizará pruebas sobre 100 usuarios novatos y se medirá el tiempo que tardan en superar el primer nivel.
Peor caso	El 30% supera el primer nivel en menos de 10 minutos.
Nivel planificado	El 80% supera el primer nivel en menos de 10 minutos.
Mejor caso	El 100% supera el primer nivel en menos de 10 minutos.
Nivel actual	

Nombre	Bajo uso de Almacenamiento (Eficiencia).
Id	RNF4
Categoría	No Funcional
Descripción	La aplicación de escritorio debe ocupar menos de 200 Mb de disco (instalado). Mientras que la versión móvil no debe superar los 50 Mb.
Escala	Megabytes ocupados en disco o almacenamiento interno.
Test	Se instalará el juego en un ordenador y en un dispositivo con Android y se verificará cuanto tamaño en disco y en almacenamiento interno ocupa en cada caso.
Peor caso	Entre 200-500 Mb (Escritorio) – Entre 50-150 Mb (Móvil).
Nivel planificado	Entre 180-200 Mb (Escritorio) – Entre 30-50 Mb (Móvil).
Mejor caso	Entre 100-180 Mb (Escritorio) – Entre 15-30 Mb (Móvil).
Nivel actual	

Nombre	Fluidez (Eficiencia).
Id	RNF5
Categoría	No Funcional
Descripción	La aplicación debe funcionar correctamente y sin perder fluidez en un ordenador o dispositivo móvil de gama baja.
Escala	Fotogramas por segundos (fps) del videojuego en ejecución.

Test	Se probará la aplicación en dos ordenadores y en dos dispositivos móviles de gama baja y se verificarán los fps en ejecución con un escenario con la máxima cantidad de unidades en el mapa.
Peor caso	Pico mínimo de 20 fps en una partida de prueba.
Nivel planificado	Pico mínimo de 60 fps en una partida de prueba.
Mejor caso	Pico mínimo de 90 fps en una partida de prueba.
Nivel actual	

Riesgos

Descripción de Riesgos

En el transcurso del desarrollo del proyecto se detectaron los siguientes riesgos:

- Cantidad de descargas menor al mínimo previsto:

Debido a que el target final de usuarios puede no verse interesado en el producto por falta de exposición, la probabilidad de que ocurra es media pero su impacto es alto ya que la oportunidad de negocio depende de que los usuarios finales descarguen la aplicación.

- Falta de experiencia en el personal:

Debido a que el personal puede tener poca o ninguna experiencia en el desarrollo de este tipo de aplicación la probabilidad es media, pero con un alto impacto si esto ocurre ya que se cuenta con personal limitado.

- Tamaño de la aplicación mayor al previsto:

Existe una probabilidad media de que la interfaz gráfica contribuya a que la aplicación exceda el tamaño previsto, sin embargo el impacto (si es que ocurre) es bajo ya que no afecta demasiado a los usuarios finales.

- Baja usabilidad reportada por usuarios:

Existe una probabilidad media de que los usuarios finales encuentren a la aplicación difícil de utilizar o no amigable. Si esto ocurre, el impacto sería crítico ya que podría evitar que los usuarios dejen de utilizar la aplicación completamente perdiendo la oportunidad de negocio detectado.

- Fecha límite de finalización comprometida:

Existe una baja probabilidad de que el tiempo de desarrollo del proyecto se extienda más allá de la fecha límite de finalización prevista. Esto podría tener un impacto catastrófico ya que se podría perder la oportunidad de negocio a la competencia si esta lanza un producto similar.

- Recaudación por microtransacciones menor a la prevista:

Existe una baja posibilidad de que las recaudaciones por microtransacciones luego del lanzamiento de la aplicación sean menores a las previstas, lo cual implicaría un impacto crítico en el negocio ya que estas la mayor parte de donde provienen las recaudaciones del producto.

- Baja cantidad o calidad de documentación:

La posibilidad de que la cantidad de documentación sea baja o de baja calidad podría dificultar el testeado o la futura reutilización y/o venta del producto para próximas implementaciones, lo cual tendría un impacto medio o marginal en el proyecto. Sin embargo, se estima que la probabilidad de que esto ocurra es baja.

- Baja posibilidad de reutilización para próximas implementaciones:

Uno de los objetivos del proyecto es desarrollar un producto que pueda ser reutilizable para próximas implementaciones dentro de esta empresa o incluso para su venta. Aunque la posibilidad de que el producto desarrollado no sea lo suficientemente adaptable es baja, esto conllevaría un impacto crítico por las razones antes mencionadas.

- Baja aceptación de usuarios con respecto a la temática y jugabilidad:

Existe la posibilidad de que los usuarios finales encuentren la jugabilidad y/o la temática de la aplicación fuera de sus expectativas esperadas como televidentes y aficionados de la serie, lo cual implicaría un impacto alto o crítico ya que podría provocar que dejen de utilizar la aplicación, así como también una publicidad no deseada mediante las opiniones negativas de los usuarios. Sin embargo se estima que la probabilidad de que esto ocurra es baja ya que estas características son las principales en el proceso de desarrollo.

- Baja performance de la aplicación:

Se estima que una baja performance del producto puede tener un impacto medio ya que es posible que para algunos usuarios esta característica provoque desuso de la aplicación. Sin embargo, debido a que el producto final no será de gran complejidad o escala, se estima que la probabilidad de mala performance de la aplicación sea baja.

- Colapso de servidores por mayor cantidad de descargas que las previstas:

Se tomó en cuenta que un posible escenario luego de lanzar el producto sea la alta cantidad de descargas simultáneas de la aplicación, lo cual podría colapsar los servidores y tener un impacto crítico en el negocio debido a que los posibles usuarios finales no podrán obtener su copia del producto. Sin embargo se estima que la probabilidad de que esto ocurra es baja ya que los servidores con los que se cuenta son de alta capacidad.

- Personal insuficiente:

Se estima que hay una baja probabilidad de que el personal con el que se cuenta sea insuficiente para el desarrollo del producto lo cual tendría un impacto medio debido a que la posibilidad de incorporación de nuevo personal limitada.

- Producto final no posee soporte para los idiomas básicos:

Se estima que la posibilidad de que el producto final no tenga soporte para los idiomas básicos (español, inglés y chino simplificado) es muy baja y de bajo impacto, ya que la aplicación no tendrá grandes cantidades de texto para su traducción y además será desarrollada con énfasis a un entorno mayormente gráfico e intuitivo.

Clasificación de Riesgos

Para una buena clasificación de los riesgos, a continuación se muestra una tabla la cual contiene la categoría, la probabilidad, el impacto y, en caso de ser necesario, un plan de reducción, supervisión y gestión de cada riesgo.

Riesgo	Categoría	Probabilidad	Impacto	Plan
Cantidad de descargas menor al mínimo previsto.	NE	55%	3	I.
Personal con poca experiencia en el desarrollo de esta tecnología.	PE	50%	3	II.
Tamaño de la aplicación mayor a la prevista.	TE	50%	1	III.
Baja usabilidad reportada por usuarios.	DE	40%	3	IV.
Línea de corte				
Fecha límite de finalización comprometida.	NE	35%	4	
Recaudación por microtransacciones menor a la prevista.	NE	35%	3	
Baja cantidad o calidad de documentación.	NE	35%	2	
Baja posibilidad de reutilización del producto para próximas implementaciones.	DE	30%	3	
Baja aceptación de usuarios con respecto a la temática y jugabilidad.	NE	30%	2	
Baja performance de la aplicación.	TE	25%	2	
Colapso de servidores por mayor cantidad de descargas que las previstas.	TA	20%	3	
Personal insuficiente.	PE	10%	2	
Producto final no posee soporte para los idiomas básicos.	DE	10%	1	

Impactos	Categorías:
4: Catastrófico	NE: Negocio
3: Crítico	PE: Personal
2: Marginal	TE: Técnico
1: Insignificante	DE: Desarrollo
	TA: Tamaño

Planes de Reducción, Supervisión y Gestión de Riesgos (PRSG)

I - Cantidad de descargas menor al mínimo previsto.

Reducción

Planificar un plan publicitario del producto que abarque diferentes medios y plataformas.

Supervisión

Una vez lanzado el producto, monitorear las descargas, realizar métricas y proyectar la cantidad de descargas a futuro.

Gestión

- Invertir en una campaña publicitaria de alcance mayor a la previa, posiblemente contratando a terceros.
- Discutir la posibilidad de publicitar el producto desde la propia serie televisiva.

Observaciones

Esto se realizará en el sprint 4.

II - Personal con poca experiencia en el desarrollo de esta tecnología.

Reducción

Capacitar y brindar los conocimientos adecuados al personal actual sobre la nueva tecnología desarrollar mediante cursos rápidos.

Supervisión

Mantenerse en contacto con el personal para detectar problemas con su entendimiento de las nuevas tecnologías a desarrollar.

Gestión

Contratar nuevo personal para poder capacitar, y posiblemente guiar, al personal actual con la nueva tecnología.

III - Tamaño de la aplicación mayor a la prevista.

Reducción

-

Supervisión

-

Gestión

-

Observaciones

No se toma en cuenta un plan RSGR para el riesgo "Tamaño de la aplicación mayor a la prevista" ya que aunque la probabilidad es media, su impacto es insignificante.

IV - Baja usabilidad reportada por usuarios.

Reducción

- Hacer énfasis en la simplicidad y claridad de la interfaz con el fin de que los usuarios encuentren a la aplicación simple y fácil de usar.
- Realizar prototipos (Simples, dibujos en papel) para usar como tests con usuarios ajenos al equipo de desarrollo.

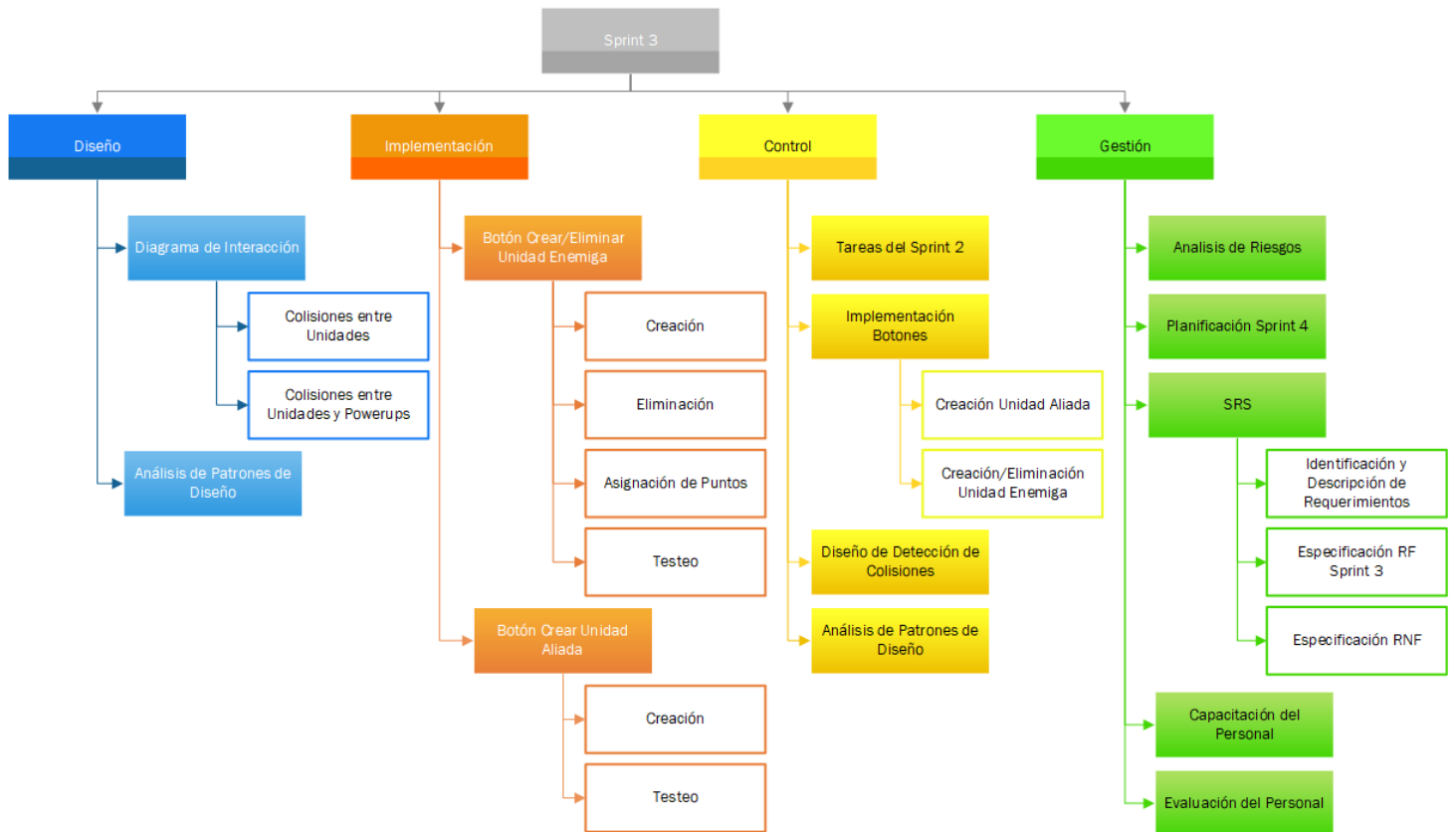
Supervisión

Encuestar a los testers con respecto a la usabilidad del producto. Esto se realizará en el sprint 4.

Gestión

Contratar nuevo personal especializado con el motivo de reimplementar una nueva interfaz de usuario en base a los resultados de los tests realizados.

Estructura de Descomposición del Trabajo



Listado de Actividades

Id	Nombre	Duración	Comienzo	Fin	Predecesoras	Entregables
1	1. Diseño	2,62 días?	19/09/2017	22/09/2017		
2	1.1. Análisis de Patrones de Diseño	1,62 días?	19/09/2017	21/09/2017		Doc. Patrones de Diseño
3	1.2. Diagrama de interacción	1 día?	21/09/2017	22/09/2017	2	
4	1.2.1. Colisiones entre Unidades	0,5 días?	21/09/2017	21/09/2017		Diag. de Interacción
5	1.2.2. Colisiones entre Unidades y Powerups	0,5 días?	21/09/2017	22/09/2017	4	Diag. de Interacción
6	1.3. Prototipos de Interfaz de Usuario	1 día	19/09/2017	20/09/2017		Prototipo en papel
7	2. Implementación	6,01 días?	21/09/2017	29/09/2017		
8	2.1. Botón Creación y Eliminación de Unidad Enemiga	6,01 días?	21/09/2017	29/09/2017		
9	2.1.1. Creación Unidad Enemiga	2 días?	21/09/2017	25/09/2017	2	Cód. Fuente
10	2.1.2. Eliminación Unidad Enemiga	2 días?	25/09/2017	27/09/2017	9	Cód. Fuente
11	2.1.3. Asignación de Puntos	1,01 días?	27/09/2017	28/09/2017	10	Cód. Fuente
12	2.1.4. Testeo	1 día?	28/09/2017	29/09/2017	9;10;11	Doc. Resultado Testeo
13	2.2. Botón Creación y Eliminación de Unidad Aliada	3 días?	21/09/2017	26/09/2017	2	
14	2.2.1. Creación Unidad Aliada	2 días?	21/09/2017	25/09/2017		Cód. Fuente
15	2.2.2. Testeo	1 día?	25/09/2017	26/09/2017	14	Doc. Resultado Testeo
16	3. Control	8,23 días?	19/09/2017	29/09/2017		
17	3.1. Tareas de Sprint 2	0,5 días?	19/09/2017	19/09/2017		Checklist de Tareas
18	3.2. Implementación botones	3,61 días?	26/09/2017	29/09/2017	8;13	
19	3.2.1. Creación Unidad Aliada	0,6 días?	26/09/2017	26/09/2017		Doc. Control, Cód. Fuente
20	3.2.2. Creación/Eliminación Unidad Enemiga	0,6 días?	29/09/2017	29/09/2017		Doc. Control, Cód. Fuente
21	3.3. Diseño de Detección de Colisiones	0,5 días?	22/09/2017	22/09/2017	3	Doc. Control, Diag. de Interacción
22	3.4. Análisis de Patrones de Diseño	0,5 días?	21/09/2017	21/09/2017	2	Doc. Control, Doc. Patrones de Diseño
23	4. Gestión	9,8 días?	19/09/2017	03/10/2017		
24	4.1. SRS	6 días?	19/09/2017	27/09/2017		
25	4.1.1. Identificación y Descripción de RF	4 días?	19/09/2017	25/09/2017		SRS
26	4.1.2. Especificación de RF Sprint 3	6 días?	19/09/2017	27/09/2017		SRS
27	4.1.3. Especificación de RNF	4 días?	19/09/2017	25/09/2017		SRS
28	4.2. Elaboración de Plan RSGR	4 días?	19/09/2017	25/09/2017		PRSGR
29	4.3. Planificación de Sprint 4	1,3 días?	27/09/2017	28/09/2017	24	Diag. De Gantt
30	4.4. Capacitaciones de Personal	2,5 días?	28/09/2017	03/10/2017	28	
31	4.5. Reuniones Evaluativas de Personal	1 día?	28/09/2017	29/09/2017	28;30	Resultados Evaluaciones

Planificación

Diagrama de Red

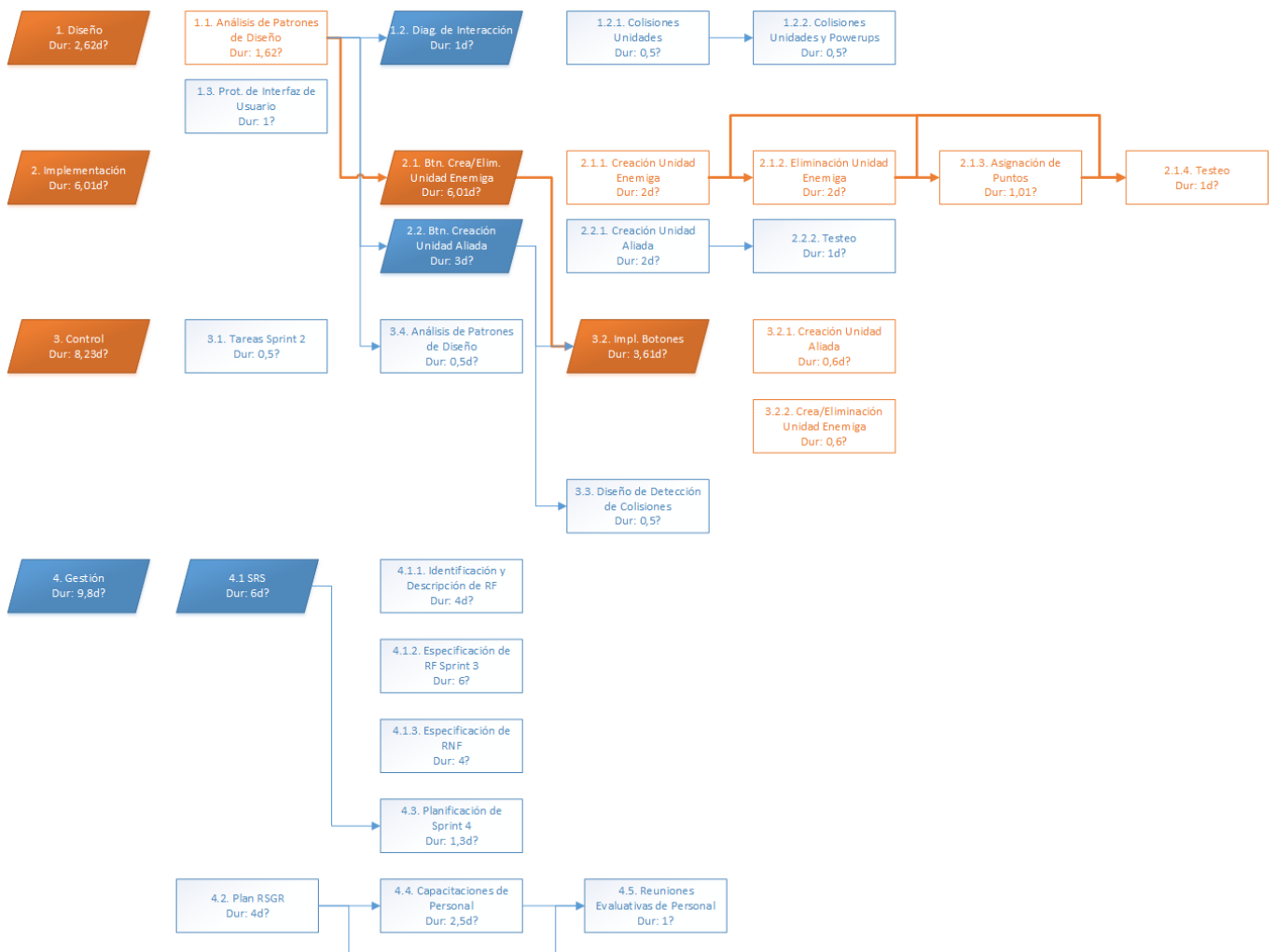


Diagrama de Gantt

