

# Jaipur 3D

Planeación del Proyecto

## TABLA DE CONTENIDOS

1	Declaración de alcance.....	4
1.1	Objetivo: .....	4
1.2	Requerimientos: .....	4
1.3	Criterio de aceptación: .....	4
1.4	Entregables:.....	4
1.5	WBS inicial: .....	4
1.6	Requerimientos de aprobación:.....	5
1.7	Integrantes del equipo .....	5
2	Work Breakdown Structure.....	6
3	Roles .....	7
4	Calendarización .....	8
5	Análisis de Riesgos .....	9
5.1	General .....	9
5.2	Diseño de manejador del juego .....	9
5.3	Diseño de otros componentes .....	10
5.4	Diseño de interfaz de usuario .....	10
5.5	Diseño de servidor.....	10
5.6	Diseño de cliente .....	11
5.7	Implementación del menú de juego .....	11
5.8	Implementación del menú principal .....	12
5.9	Programación de otros componentes.....	12
5.10	Programación del cliente y programación del servidor .....	13
5.11	Programación del manejador del juego.....	13
5.12	Integración .....	14
5.13	Recopilación de referencias .....	14
5.14	Diseño del manual de usuario .....	14
5.15	Impresión del manual de usuario.....	15
5.16	Modelado de cartas y fichas .....	15

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

5.17	Texturizado de objetos.....	16
5.18	Proceso de Testing .....	16
5.19	Compra o adquisición de la caja y disco.....	16
5.20	Diseño de la portada de la caja .....	17
5.21	Impresión de la portada de la caja .....	17
5.22	Elaboración de instalador binario .....	17
5.23	Preparación de entregables .....	17
5.24	Resolución de Issues.....	18
5.25	Entrega .....	18
6	Programación Basada en Componentes .....	19
6.1	¿Qué es un componente? .....	19
6.2	¿Cómo y por qué Unity favorece el uso de componentes? .....	19
6.3	Pensando en componentes .....	19
6.4	Conclusión .....	20
7	Proceso de Testing .....	21
7.1	Objetivo .....	21
7.2	Metodología .....	21
7.3	Equipo de Testing .....	21
7.4	Aprobación .....	21

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

# 1 DECLARACIÓN DE ALCANCE

---

## 1.1 OBJETIVO:

Producir el videojuego “Jaipur 3D” en 5 semanas.

## 1.2 REQUERIMIENTOS:

El juego debe tener gráficos con perspectiva tridimensional y multijugador en red (máximo 2 jugadores). Las reglas del juego deben de ser las mismas del juego de mesa “Jaipur” de Sébastien Pauchon.

Debe poder instalarse y jugarse en una computadora que tenga las siguientes especificaciones (para poder jugarse en la configuración de gráficos “baja”):

- Tarjeta de video dedicada o integrada con al menos 512MB de memoria gráfica.
- DirectX 11 o superior.
- Sistema operativo: Windows 10 de 64 bits.
- CPU: Que soporte set de instrucciones SSE2.
- RAM: Al menos 2 GB de memoria.
- Al menos 60 MB de espacio en disco duro.
- Monitor que soporte resolución de 1366 x 768 o 1920 x 1080.
- Teclado y mouse.

## 1.3 CRITERIO DE ACEPTACIÓN:

El juego debe aprobar el proceso de Testing. El desarrollo del proyecto debe de mantener su calendarización para poder ser terminado en el plazo de 5 semanas.

## 1.4 ENTREGABLES:

Instalador del juego binario, código fuente del videojuego, manual de usuario, caja, portada de la caja y disco.

- |                        |                           |                        |
|------------------------|---------------------------|------------------------|
| • Instalador del juego | • Caja                    | • Calendarización      |
| • Código fuente        | • Portada de la caja      | • Documentos de diseño |
| • Manual de usuario    | • Disco con el instalador |                        |

## 1.5 WBS INICIAL:

El desarrollo del proyecto estará dividido en 4 fases: programación, arte, pruebas y manejo de entregables.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

## 1.6 REQUERIMIENTOS DE APROBACIÓN:

“Jaipur 3D” será entregado después de ser aprobado por los mismos desarrolladores y por el asesor del proyecto, el Maestro en Ciencias Juan Francisco Algara Norzagaray.

## 1.7 INTEGRANTES DEL EQUIPO

- Christian Ricardo Lugo Arellano
- José David Rochín Cerecer

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

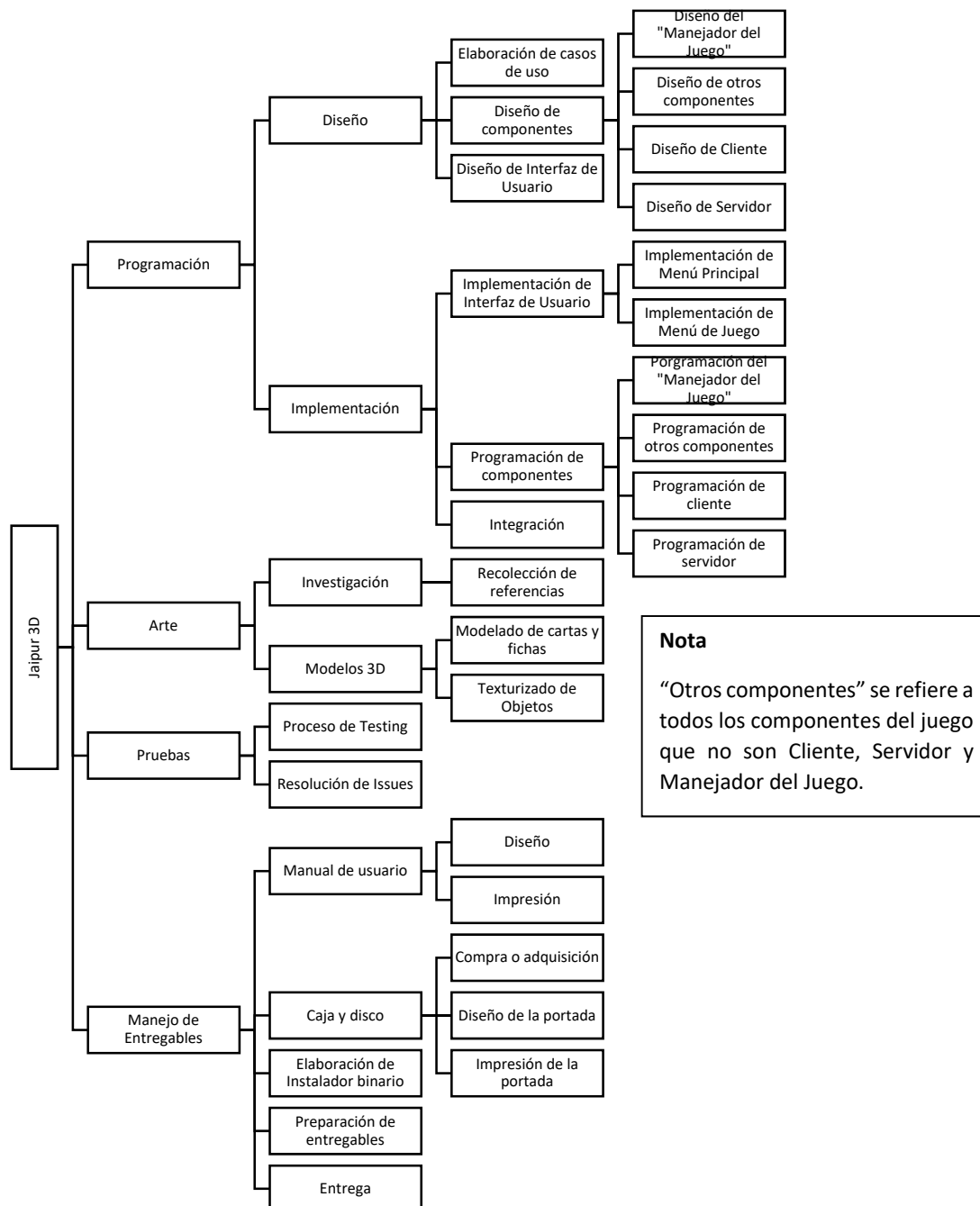
---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

## 2 WORK BREAKDOWN STRUCTURE



Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

### 3 ROLES

Persona	Roles
Christian Ricardo Lugo Arellano	Programador de Cliente y Servidor, Testing.
José David Rochín Cerecer	Programador del Juego, Testing.
M.C. Juan Francisco Algara Norzagaray	Asesor del Proyecto

Firmas de enterado y aceptado

\_\_\_\_\_  
Christian Ricardo Lugo  
Arellano

\_\_\_\_\_  
José David Rochín Cerecer

\_\_\_\_\_  
M.C. Juan Francisco Algara  
Norzagaray

## 4 CALENDARIZACIÓN

No.	Persona asignada	Tarea	Diagrama																																			
			Octubre																Noviembre																			
			17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	Christian	Elaboración de Casos de Uso																																				
2	Rochín	Diseño del "Manejador del Juego"																																				
3	Rochín	Diseño de otros componentes																																				
4	Christian	Diseño de Interfaz de Usuario																																				
5	Christian	Diseño del cliente																																				
6	Christian	Diseño del servidor																																				
7	Rochín	Implementación de Menú de Juego																																				
8	Christian	Implementación de Menú Principal																																				
9	Rochín	Programación de otros componentes																																				
10	Christian	Programación del cliente																																				
11	Christian	Programación del servidor																																				
12	Rochín	Programación del "Manejador del Juego"																																				
13	Rochín	Integración																																				
14	Christian	Recolección de referencias																																				
15	Christian	Diseño del Manual de Usuario																																				
16	Christian	Impresión del Manual de Usuario																																				
17	Rochín	Modelado de cartas y fichas																																				
18	Rochín	Texturizado de objetos																																				
19	Christian	Proceso de Testing																																				
20	Christian	Compra o adquisición de la caja y disco																																				
21	Rochín	Diseño de la portada de la caja																																				
22	Rochín	Impresión de la portada de la caja																																				
23	Rochín	Elaboración de Instalador Binario																																				
24	Rochín	Preparación de entregables																																				
25	Todos	Resolución de Issues																																				
26	Todos	Entrega																																				
	Tarea de Christian		Notas																																			
	Tarea de Rochín		"Otros componentes" se refiere a todos los componentes del juego que no son el Cliente, Servidor y Manejador del Juego.																																			
	Tarea de todos																																					
	Se extendió la tarea																																					

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray



## 5 ANÁLISIS DE RIESGOS

### 5.1 GENERAL

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El videojuego no es terminado.	El equipo de trabajo no concluye el desarrollo del videojuego; queda en un estado inutilizable.	10%	40 puntos	Aceptar.
El videojuego es parcialmente terminado.	El equipo de trabajo logra tener un videojuego usable pero no cuenta con todos los requerimientos y características definidas en la planeación.	10%	15 puntos	Mitigar. Durante cada tarea, se analizará si el tiempo determinado para terminar el proyecto es apropiado, y de no ser así, se contratará a alguien externo para ayudar con el desarrollo.
Faltas ortográficas	Los textos de la interfaz gráfica del videojuego tienen faltas ortográficas.	5%	3 puntos	Mitigar. Se verificará la ortografía de los textos del videojuego en la etapa de Testing.

### 5.2 DISEÑO DE MANEJADOR DEL JUEGO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El diseño no se termina, por lo tanto, no se implementa.	20%	30 puntos	Mitigar. Durante la tarea de diseño, el administrador del proyecto verificará que su avance sea suficiente para terminarla.
El diseño es incorrecto.	El diseño del Manejador del Juego es calificado por el asesor como "incorrecto".	60%	1 punto	Mitigar. Se revisará el diseño con el asesor siempre que sea posible.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

### 5.3 DISEÑO DE OTROS COMPONENTES

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El diseño no se termina, por lo tanto, no se implementa.	20%	30 puntos	Mitigar. Durante la tarea de diseño, el administrador del proyecto verificará que su avance sea suficiente para terminarla.
El diseño es incorrecto.	El diseño de la estructura de componentes del Juego es calificado por el asesor como "incorrecto".	60%	1 punto	Mitigar. Se revisará el diseño con el asesor siempre que sea posible.

### 5.4 DISEÑO DE INTERFAZ DE USUARIO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El diseño no se termina, por lo tanto, no se implementa.	30%	30 puntos	Mitigar. Durante la tarea de diseño, el administrador del proyecto verificará que su avance sea suficiente para terminarla.
El diseño es incorrecto.	El diseño de la interfaz de usuario es calificado por el asesor como "incorrecto".	60%	1 punto	Mitigar. Se revisará el diseño con el asesor siempre que sea posible.

### 5.5 DISEÑO DE SERVIDOR

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El diseño no se termina, por lo tanto, no se implementa.	30%	30 puntos	Mitigar. Durante la tarea de diseño, el administrador del proyecto verificará que su avance sea suficiente para terminarla.
El diseño es incorrecto.	Los diseños del cliente y el servidor son calificados por el asesor como "incorrectos".	60%	1 punto	Mitigar. Se revisará el diseño con el asesor siempre que sea posible.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

## 5.6 DISEÑO DE CLIENTE

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El diseño no se termina, por lo tanto, no se implementa.	30%	30 puntos	Mitigar. Durante la tarea de diseño, el administrador del proyecto verificará que su avance sea suficiente para terminarla.
El diseño es incorrecto.	Los diseños del cliente y el servidor son calificados por el asesor como "incorrectos".	60%	1 punto	Mitigar. Se revisará el diseño con el asesor siempre que sea posible.

## 5.7 IMPLEMENTACIÓN DEL MENÚ DE JUEGO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Es implementado de una manera muy diferente a su diseño.	La implementación no sigue las partes básicas e importantes del diseño.	20%	1 punto	Aceptar. Se tiene poca experiencia en diseño de interfaces, así que es muy probable que la implementación no siga algunas partes básicas del diseño.
No se termina de implementar.	No se termina de implementar, por lo tanto, el juego no funciona parcial o totalmente.	20%	30 puntos	Mitigar. Durante la tarea de implementación, el administrador del proyecto verificará que su avance sea suficiente para terminarla.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

## 5.8 IMPLEMENTACIÓN DEL MENÚ PRINCIPAL

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Es implementado de una manera muy diferente a su diseño.	La implementación no sigue las partes básicas e importantes del diseño.	20%	1 punto	Aceptar. Se tiene poca experiencia en diseño de interfaces, así que es muy probable que la implementación no siga algunas partes básicas del diseño.
No se termina de implementar.	No se termina de implementar, por lo tanto, el juego no funciona parcial o totalmente.	20%	30 puntos	Mitigar. Durante la tarea de implementación, el administrador del proyecto verificará que su avance sea suficiente para terminarla.

## 5.9 PROGRAMACIÓN DE OTROS COMPONENTES

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Es programado de una manera muy diferente a su diseño.	La programación no sigue las partes básicas e importantes del diseño.	20%	1 punto	Aceptar. Se tiene poca experiencia en diseño de software, así que es muy probable que la programación no siga algunas partes básicas del diseño.
No se termina de programar.	No se termina de programar, por lo tanto, el juego no funciona parcial o totalmente.	20%	30 puntos	Mitigar. Durante la tarea de programación, el administrador del proyecto verificará que su avance sea suficiente para terminarla.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

### 5.10 PROGRAMACIÓN DEL CLIENTE Y PROGRAMACIÓN DEL SERVIDOR

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Es programado de una manera muy diferente a su diseño.	La programación no sigue las partes básicas e importantes del diseño.	20%	1 punto	Aceptar. Se tiene poca experiencia en diseño de software, así que es muy probable que la programación no siga algunas partes básicas del diseño.
No se termina de programar.	No se termina de programar, por lo tanto, el juego no funciona parcial o totalmente.	20%	30 puntos	Mitigar. Durante la tarea de programación, el administrador del proyecto verificará que su avance sea suficiente para terminarla.

### 5.11 PROGRAMACIÓN DEL MANEJADOR DEL JUEGO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Es programado de una manera muy diferente a su diseño.	La programación no sigue las partes básicas e importantes del diseño.	20%	1 punto	Aceptar. Se tiene poca experiencia en diseño de software, así que es muy probable que la programación no siga algunas partes básicas del diseño.
No se termina de programar.	No se termina de programar, por lo tanto, el juego no funciona parcial o totalmente.	20%	30 puntos	Mitigar. Durante la tarea de programación, el administrador del proyecto verificará que su avance sea suficiente para terminarla.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

## 5.12 INTEGRACIÓN

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se termina la integración.	No se termina la integración de los componentes, por lo tanto, el juego funciona parcial o totalmente.	10%	30 punto	Mitigar. En cuanto se detecte que el tiempo de integración no es suficiente, se recalendarizará la tarea.

## 5.13 RECOPIACIÓN DE REFERENCIAS

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se encuentran todas las referencias necesarias.	La búsqueda de referencias no concluye debido a que las fuentes consultadas no contienen la información necesaria para hacerlo.	20%	1 punto	Mitigar. Si se detecta que no hay mucha información, se conseguirá una copia física del juego de mesa.
Se pierden los archivos.	Los archivos resultantes de la recopilación son extraviados o dañados.	5%	5 puntos	Mitigar. Los archivos se guardarán en el repositorio remoto siempre que se tenga la oportunidad.

## 5.14 DISEÑO DEL MANUAL DE USUARIO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se termina.	El diseño del manual de usuario no es terminado.	10%	5 punto	Aceptar.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

### 5.15 IMPRESIÓN DEL MANUAL DE USUARIO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El manual no se imprime.	Debido a dificultades de cualquier tipo, la impresión del manual no se lleva a cabo.	10%	5 punto	Mitigar. Se asignarán dos días completos para la impresión del manual.
Las medidas de la impresión son incorrectas.	Ya sea por error del responsable o por problema técnico, la impresión resulta tener el tamaño incorrecto.	20%	1 punto	Mitigar. Se verificará la configuración de las medidas de la impresión en una impresora casera antes de ir a imprimirlo a color a un negocio.

### 5.16 MODELADO DE CARTAS Y FICHAS

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se terminan de modelar los objetos.	La persona encargada de modelar los objetos no termina de hacerlo; por lo tanto, el videojuego se queda sin modelos.	5%	20 puntos	Mitigar. En caso de que el tiempo no alcance para modelar, se usaran modelos primitivos que vienen incluidos con Unity.
Se pierden los archivos.	Los archivos resultantes del modelado son extraviados o dañados.	5%	5 puntos	Mitigar. Los archivos se guardarán en el repositorio remoto.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

### 5.17 TEXTURIZADO DE OBJETOS

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se terminan de texturizar los objetos.	La persona encargada de texturizar los objetos no termina de hacerlo.	5%	1 punto	Aceptar.
Se pierden los archivos.	Los archivos resultantes del texturizado son extraviados o dañados.	5%	1 punto	Mitigar. Los archivos se guardarán en el repositorio remoto siempre que se tenga la oportunidad.
Se encuentran texturas libres de uso en internet.	Nos ahorramos trabajo de texturizado debido a que encontramos texturas de uso libre en internet que sirven para el juego.	90%	1 punto	Aceptar.

### 5.18 PROCESO DE TESTING

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se detectan errores importantes durante el proceso de Testing.	Se encuentran problemas importantes en el juego después (y fuera) del proceso de Testing.	20%	1 punto	Mitigar. Además de llevar a cabo el proceso de Testing interno, se le pedirá informalmente a amigos y familiares que prueben el juego y ayuden a encontrar errores.

### 5.19 COMPRA O ADQUISICIÓN DE LA CAJA Y DISCO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se encuentra la caja.	No se encuentra una caja adecuada para contener el disco del instalador del juego.	5%	1 punto	Aceptar. Se usará una caja usada que ya posee el equipo.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray



## 5.20 DISEÑO DE LA PORTADA DE LA CAJA

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El diseño no es terminado.	El responsable de elaborar el diseño de la caja no lo termina antes de la fecha especificada.	10%	2 puntos	Mitigar. Se elaborará un diseño básico que solo incluya el nombre del juego.

## 5.21 IMPRESIÓN DE LA PORTADA DE LA CAJA

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se imprime la portada.	La portada de la caja no se imprime.	10%	1 punto	Mitigar. La tarea de la impresión de la portada se calendarizará días antes de la fecha de entrega.
Las medidas de la impresión son incorrectas.	Ya sea por error del responsable o por problema técnico, la impresión resulta tener el tamaño incorrecto.	20%	1 punto	Mitigar. Se verificarán las medidas del archivo imprimiéndolo antes en una impresora casera en blanco y negro.

## 5.22 ELABORACIÓN DE INSTALADOR BINARIO

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se elabora el instalador.	Debido a problemas de cualquier tipo, el responsable de elaborar el instalador del juego no lo hace.	5%	1 punto	Mitigar. Si el responsable no termina, se recalendarizará la tarea y se cambiará de responsable.

## 5.23 PREPARACIÓN DE ENTREGABLES

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
El disco pasa a ser inservible.	Por problemas técnicos, el disco que llevará el instalador del juego "se echa a perder"; o sea, pasa a ser inservible.	25%	1 punto	Mitigar. Se comprarán al menos 3 discos de diferentes marcas.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

## 5.24 RESOLUCIÓN DE ISSUES

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
No se resuelven Issues de prioridad alta.	El tiempo no alcanza para resolver todas las Issues con prioridad alta.	30%	1 punto	Mitigar. Se asignarán a todos los miembros del equipo a resolver Issues para aumentar las probabilidades de terminar.

## 5.25 ENTREGA

Riesgo	Descripción	Probabilidad	Costo	Acción a tomar
Los entregables no están listos.	Los entregables no están listos para el día de su entrega.	5%	10 puntos	Aceptar.

Firmas de enterado y aceptado

Christian Ricardo Lugo  
Arellano

José David Rochín Cerecer

M.C. Juan Francisco Algara  
Norzagaray

## 6 PROGRAMACIÓN BASADA EN COMPONENTES

---

Mientras que la clásica Programación Orientada a Objetos (POO) puede ser, y es usada, el flujo de trabajo de Unity se construye alrededor de la estructura de componentes—lo cual requiere pensamiento basado en componentes.

### 6.1 ¿QUÉ ES UN COMPONENTE?

En el mundo de la programación, el concepto de “componentes” y “desacoplamiento” van de la mano. Se puede decir que un componente es una pieza pequeña de una máquina más grande. Cada componente tiene su propio trabajo específico, y pueden (óptimamente) cumplir su tarea o propósito sin la ayuda de fuentes externas. Adicionalmente, los componentes rara vez pertenecen a una sola máquina, y pueden ser unidos con varios sistemas para cumplir con su tarea, pero generar otro resultado. Esto se debe a que los componentes no solo no les importa saber sobre el sistema al que pertenecen, sino que tampoco saben que existe.

Considere un mando de Xbox 360. Tiene 2 sticks análogos, varios botones, gatillos, etc. No solo el mando entero es un componente, sino que también, cada aspecto individual del control es un componente.

El botón X puede: ser presionado; informar que fue presionado; ser liberado; informar que fue liberado. No tiene idea de que hay varios otros botones al lado de él y no le importa.

El mando en sí es un componente, compuesto de otros componentes (todos los botones, joysticks y gatillos), porque puede mandar datos sin importar donde esté conectado, y que tipo de objeto es (Xbox, PC, alguna creación de Arduino u otra cosa). El botón X y el mando en sí, no necesitan saber que juego está jugando, y hará su trabajo sin importarle el receptor de la información que mandan.

La función del mando es unidireccional, y su tarea nunca va a cambiar de receptor a receptor. Esto es lo que lo convierte en un componente exitoso; puede hacer su trabajo él solo, y puede hacer su trabajo conectado a múltiples dispositivos.

### 6.2 ¿CÓMO Y POR QUÉ UNITY FAVORECE EL USO DE COMPONENTES?

Unity se construyó teniendo a los componentes en mente, y lo demuestra. Uno de los aspectos más valorados y distintivos de Unity, es que es un programa muy visual. Unity te deja ver todo en lo que estás trabajando en tiempo real. Esto significa que puedes probar tu proyecto, ver tu proyecto corriendo en una ventana separada, hacer cambios al código y ver esos cambios reflejados en tiempo real. La cantidad de poder que este sistema le da al desarrollador es inmensa. Todo esto fue posible gracias a la arquitectura basada en componentes de Unity.

### 6.3 PENSANDO EN COMPONENTES

La parte más difícil de trabajar con componentes, es aprender como estructurar tus proyectos al usarlo. Para la mayoría de los programadores, esto probablemente significa crear un montón de scripts que harán tareas más pequeñas y específicas.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

La comunicación entre scripts también es un obstáculo decente, ya que tendrás más piezas y menos clases gigantes en las que cada objeto conoce a todos los demás. Obviamente, hay maneras de superar este obstáculo, como variables estáticas (para partes clave del juego como jugadores, puntuación, etc.), pero eso raramente funciona para cualquier cosa. También hay métodos avanzados para estructurar correctamente los componentes y mantenerlos desacoplados.

Por suerte, como Unity fue construido pensando en componentes, tiene una variedad de funciones integradas que nos ayudan a lograr esto. Hay funciones para obtener referencias a un componente específico, para revisar todos los objetos y ver si contienen un componente específico, etc. Con estas funciones uno puede fácilmente conseguir la información necesaria para crear una comunicación correcta entre componentes.

## 6.4 CONCLUSIÓN

Pensar en componentes puede ser difícil, pero definitivamente tiene sus beneficios. Al usar Unity, uno puede elegir el método que prefiera, pero definitivamente, pensar en componentes es la mejor opción.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

## 7 PROCESO DE TESTING

---

### 7.1 OBJETIVO

La función principal del proceso de testeo es el descubrimiento y documentación de los defectos en el videojuego (bugs).

### 7.2 METODOLOGÍA

Para empezar el proceso de Testing, se verificará mediante la prueba del videojuego que todos los casos de uso elaborados se cumplan. Posteriormente se revisará que todas las reglas oficiales del juego de mesa original sean aplicadas en el videojuego.

Todos los errores que sean identificados se documentaran, de tal manera que sea posible reproducir el error para su posterior estudio y corrección. El equipo de Testing documentará los siguientes aspectos:

- Resultado esperado
- Resultado obtenido
- Pasos para reproducirse.

Posteriormente se le asignará un grado de severidad y una prioridad.

Durante todo el proceso de Testing, se verificará constantemente la ortografía de los textos mostrados en el videojuego.

### 7.3 EQUIPO DE TESTING

Debido a que solo habrá 2 personas trabajando en el proyecto, el equipo de testeo estará conformado por los mismos desarrolladores del videojuego.

### 7.4 APROBACIÓN

Para que el proceso de Testing pueda ser calificado como “terminado”, deberá haber cumplido con lo siguiente:

1. Se deberá haber revisado el cumplimiento de todos los casos de uso.
2. Se deberá haber verificado el cumplimiento de todas las reglas del juego original.
3. Se deberá haber comprobado la ortografía de todos los textos del juego.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

## 8 GASTOS DEL PROYECTO

---

No.	Fecha	Descripción	Gasto
1	17/10/2017	Impresión de documentos para revisión y folder	\$ 8.00
2	24/10/2017	Impresión de documentos para revisión y folder	\$ 12.50
3	31/10/2017	Impresión de documentos para revisión y folder	\$ 15.00
4	07/11/2017	Impresión de documentos para revisión y folder	\$ 9.50
5	10/11/2017	CD en blanco y repuesto	\$ 19.98
6	10/11/2017	Caja y repuesto de caja para el CD	\$ 59.98
7	10/11/2017	Impresión de manual de usuario	\$ 59.99
8	12/11/2017	Impresión de portada y contraportada	\$ 12.50
9	20/11/2017	Sobre manila para entrega	\$ 5.00
10	21/11/2017	Impresión y engargolado final	\$ 30.00
11			
12			
13			
14			
15			
<b>Total</b>			<b>\$ 232.45</b>

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

---

M.C. Juan Francisco Algara  
Norzagaray

# Jaipur 3D

Documentos de Aceptación

## DOCUMENTO DE ACEPTACIÓN

---

miércoles, 1 de noviembre del 2017

Mediante este documento, yo, José David Rochín Cerecer, administrador del proyecto Jaipur 3D, hago constar que acepto los resultados de las tareas **Programación del Cliente** y **Programación del Servidor** entregados el miércoles 1 de noviembre del año en curso por el responsable de la tarea, Christian Ricardo Lugo Arellano, los cuales están reflejados en el commit #4c58f56b36aad3e668902d690425e5b6a84eee44 del repositorio remoto.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer



## DOCUMENTO DE ACEPTACIÓN

---

jueves, 9 de noviembre del 2017

Mediante este documento, yo, José David Rochín Cerecer, administrador del proyecto Jaipur 3D, hago constar que acepto los resultados de la tarea **Diseño del Manual de Usuario** entregados el jueves 9 de noviembre del año en curso por el responsable de la tarea, Christian Ricardo Lugo Arellano, los cuales están reflejados en el commit [512d6d1e48f66bae6f14c312c40b1a3400e7e674](#) del repositorio remoto en GitHub.

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

# Jaipur 3D

Documentos de Petición

## DOCUMENTO DE PETICIÓN

---

sábado, 28 de octubre de 2017

Dirigido a:

Christian Ricardo Lugo Arellano

**Programador de Netcode**

Después de revisar el diseño de los componentes Cliente y Servidor, me he percatado de que necesitan modificaciones. Primeramente, se necesita que el componente Cliente cuente con los siguientes métodos:

- void EnviarMovimiento(Movimiento)
- int[] SolicitarOrdenGrupoPrincipal()

El primero deberá enviar el objeto de tipo Movimiento proporcionado al Cliente del otro jugador. El segundo deberá devolver un arreglo de números enteros, el cual podrá ser solicitado al ManejadorJuego mediante el Metodo "GetOrdenGrupo(Grupo)".

Pedido por:

José David Rochín Cerecer

**Programador del Juego**

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

## DOCUMENTO DE PETICIÓN

---

domingo, 29 de octubre de 2017

Dirigido a:

José David Rochín Cerecer

**Programador del Juego**

Necesito una manera de obtener los objetos Movimiento en forma serializada para poder enviarlos mediante TCP. Mi propuesta son los siguientes métodos en la clase Movimiento:

- `byte[] Serializar()` – Regresa el objeto serializado en un arreglo de bytes.
- `static Movimiento Deserializar(byte[] arr)` – Regresa un objeto de tipo Movimiento a partir de un arreglo de bytes.

Pedido por:

Christian Ricardo Lugo Arellano

**Programador de Netcode**

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

## DOCUMENTO DE PETICIÓN

---

Miércoles, 01 de noviembre de 2017

Dirigido a:

José David Rochín Cerecer

**Programador del Juego**

Necesito que la clase "Movimiento" herede de "MessageBase" debido a que tanto el cliente y como servidor necesitan enviar este tipo de mensajes.

Necesito que los implementes antes de 1:00 PM del 01 de noviembre del 2017.

Pedido por:

Christian Ricardo Lugo Arellano

**Programador de Netcode**

Firmas de enterado y aceptado

---

Christian Ricardo Lugo  
Arellano

---

José David Rochín Cerecer

# Jaipur 3D

Documentos de Diseño

# 1 DISEÑO DE COMPONENTES

