

# Ilos

Fco. Javier Bohórquez Ogalla

# Índice

<b>1. Título</b>	<b>4</b>
<b>2. Documentos básicos</b>	<b>4</b>
<b>3. Volúmenes</b>	<b>4</b>
3.1. Páginas . . . . .	5
3.2. Capítulos y apartados . . . . .	5
3.3. Redacción . . . . .	5
<b>4. Desarrollo de documentos</b>	<b>5</b>
4.1. Índice general . . . . .	5
4.2. Memoria . . . . .	5
4.2.1. Objeto . . . . .	6
4.2.2. Alcance . . . . .	6
4.2.3. Antecedentes . . . . .	6
4.2.4. Normas y referencias . . . . .	7
4.2.5. Definiciones y abreviaturas . . . . .	7
4.2.6. Requisitos de diseño . . . . .	7
4.2.7. Análisis de soluciones . . . . .	8
4.2.8. Resultados finales . . . . .	8
4.2.9. Planificación . . . . .	12
4.2.10. Orden de prioridad de documentos básicos . . . . .	12
4.3. Anexos . . . . .	12

4.4. Planos . . . . .	13
4.5. Pliego de condiciones . . . . .	13
4.6. Estado de mediciones . . . . .	13
4.7. Presupuesto . . . . .	13
4.8. Estudios con entidad propia . . . . .	14

## 1. Título

Rules. Lenguaje para el desarrollo software dentro del ámbito de los juegos de mesa. Rules. Lenguaje para la programación de juegos de mesa. Rules. Lenguaje para el desarrollo de juegos de mesa.

## 2. Documentos básicos

- Índice General
- Memoria
- Anexos
- Planos
- Pliego de condiciones
- Estado de mediciones
- Presupuesto
- Estudio con entidad propia

## 3. Volúmenes

Los documentos pueden estar agrupados en distintos volúmenes o en uno solo.

Por cada volumen

- Número de volumen
- Título del proyecto
- Tipo de documento básico unitario que comprende.
- Organismo o cliente
- Identificación y datos profesionales del autor
- Cuando corresponda, los de la persona jurídica que ha recibido el encargo de su elaboración.

### **3.1. Páginas**

- Número de página o de plano.
- Título del Proyecto o Número o código de identificación del Proyecto.
- Título del documento básico a que pertenece.
- Número o código de identificación del documento.
- Número de edición y, en su caso, fecha de aprobación.

### **3.2. Capítulos y apartados**

Segun la norma UNE 50132. (1.2.3)

### **3.3. Redacción**

- Lenguaje claro, preciso y libre de vagedadez
- Terminología coherente
- El tiempo futuro indicará requisitos obligatorios.
- El tiempo condicional o subjuntivo indicarán sugerencias o propuestas

## **4. Desarrollo de documentos**

### **4.1. Índice general**

Contendrá todos y cada uno de los índices de los diferentes documentos básicos.

### **4.2. Memoria**

- Hoja de identificación (título, datos autor, datos de cliente, fecha y firma)
- Indice de la memoria (indice de cada documento, capítulo o apartado)
- Objeto

- Alcance
- Antecedentes
- Normas y referencias
- Definiciones y abreviaturas
- Requisitos de diseño
- Análisis de soluciones
- Resultados finales
- Planificación
- Orden de prioridad entre los documentos básicos

#### **4.2.1. Objeto**

Comentar el objeto del proyecto desde el punto de vista de la necesidad que pretende cubrir. Centrarse en la tarea de digitalizar juegos y la necesidad de tener recursos que faciliten esta labor.

#### **4.2.2. Alcance**

- Centrarse en la diversidad de juegos de mesas.
- Rules se centra en la definición de reglas y el modelado lógico de los componentes.
- Rules ofrece mecanismos para tratar aspectos comunes en muchos de ellos.
- Existen particularidades de muchos juegos que rules no contempla (Física...).

#### **4.2.3. Antecedentes**

- Lenguajes formales
- Compiladores y traductores
- Algoritmia
- Estructuras de datos
- Base de datos

- Aspectos de los lenguajes de programación
- ORMs

#### **4.2.4. Normas y referencias**

- Disposiciones legales y normas aplicadas
  - Licencias a las que está sujeta cada software utilizada
  - Licencias sobre las que se desarrolla el proyecto
- Bibliografía
  - libro. Compiladores y procesadores del lenguaje
  - documento. Manual GNU Bison
  - documento. Manual GNU Flex
  - artículo. Wikipedia. Analizador léxico
  - artículo. Wikipedia. Analizador sintáctico
- Programas de cálculo ???
- Plan de gestión de calidad
  - Pruebas unitarias
  - Pruebas de rendimiento
- Otras referencias

#### **4.2.5. Definiciones y abreviaturas**

#### **4.2.6. Requisitos de diseño**

- Requisitos funcionales: Casos de usos
- Requisitos no funcionales
  - Sistema de base de datos portable
  - Rendimiento
  - Minimizar dependencias prácticas con otros sistemas software
  - Software libre
  - Interacción con base de datos relacional

#### 4.2.7. Análisis de soluciones

- Análisis de los casos de uso
- Decisiones de diseño

#### 4.2.8. Resultados finales

- Tokens y analizador léxico
- Gramática y analizador sintáctico
- Tabla de símbolos
- Nodos semánticos ejecutables
  - Generales
    - Nodo ejecutable
    - Nodo imprimible
    - Nodo expresión (definida y no definida)
    - Nodo lógico
    - Nodo aritmetico
    - Nodo textual
  - Tipos simples
    - Booleano
    - Numerico
    - Carácter
    - Valor nulo
  - Tipos compuestos
    - Arrays
    - Cadenas
    - Expresiones regulares
    - Objetos (prototipado)
    - Componente
  - Definiciones
    - Ámbito de variables
    - Funciones (parámetros por defecto y por referencia, funciones anónimas, referencias a funciones)
    - Clases (herencia, new , this)
    - Componentes



- Sentencias de control de flujo
  - if
  - if else
  - if elif ... else
  - case
  - for
  - while
  - do while
  - iloop (variable de control iterativo)
  - foreach
  - goto (labels)
  - include
  - break
  - continue ¡=====
  - return
  - exit
  - Llamada a función
- Generadores de expresiones
- Sentencias operacionales
  - Tipos
    - ◊ typeof
    - ◊ bool
    - ◊ int
    - ◊ float
    - ◊ size
  - Datos
    - ◊ Asignación
    - ◊ Asignación por referencia
    - ◊ Ternario (simplificado, con valor por defecto nulo)
    - ◊ No nulo
    - ◊ rand ¡=====
  - Entrada/Salida
    - ◊ echo (salida estandar)
    - ◊ input (entrada estándar, prompt)
    - ◊ inputline (entrada estándar, prompt)
  - Operaciones lógicas
    - ◊ OR lógico
    - ◊ OR bit a bit ¡=====
    - ◊ OR bit a bit y asignación ¡=====

- ◇ AND lógico
- ◇ AND bit a bit j=====
- ◇ AND bit a bit y asignación j=====
- ◇ NOT Lógico
- ◇ Igualdad
- ◇ Desigualdad
- ◇ Menor que
- ◇ Menor o igual que
- ◇ Idéntico j=====
- ◇ No idéntico j=====
- Operaciones aritméticas
  - ◇ Suma
  - ◇ Resta
  - ◇ Producto
  - ◇ División
  - ◇ Potencia
  - ◇ Módulo
  - ◇ Incremento y asignación
  - ◇ Asignación e incremento
  - ◇ Decremento y asignación
  - ◇ Asignación y decremento
  - ◇ Suma y asignación
  - ◇ Resta y asignación
  - ◇ Producto y asignación
  - ◇ División y asignación
  - ◇ Potencia y asignación
  - ◇ Módulo y asignación
- Cadenas
  - ◇ Concatenación
  - ◇ Concatenación y asignación
  - ◇ sprintf
  - ◇ find
  - ◇ replace
  - ◇ upper
  - ◇ lower
- Arrays
  - ◇ implode
  - ◇ explode
  - ◇ empty

- Expresiones regulares
  - ◇ match
  - ◇ search
  - ◇ replace
- Base de datos i=====
  - ◇ connect
  - ◇ select
  - ◇ insert
  - ◇ update
  - ◇ delete
- Fechas i=====
  - ◇ date
  - ◇ time
  - ◇ sleep
- Ficheros i=====
  - ◇ fopen
- Llamadas al sistema
  - ◇ Ejecución de comando (exec)
  - ◇ Interpretación de cadena (eval)
- Procesos
  - ◇ fork
- Componentes
  - ◇ Acceso
  - ◇ Búsqueda
  - ◇ Borrado
  - ◇ Asignación de atributos
- Control de errores i=====
- Sistema de eventos i=====
- Sistema de juego i=====
  - Definición de juego
    - Bloque de juego
    - Bloque de juego con cambio de turno
    - Bloque de inicialización de juego
    - Bloque de finalización de juego
    - Condiciones de finalización
    - Acotación de jugadores
  - Jugadores (jugadores totales, jugador activo, iteraciones con cambios de jugador)

- Turnos (cambio de turno, penalización de turnos, cambios con regreso)
  - Posicionamiento lógico (zona de juego, mesa)
  - Flujos de movimiento
  - Control de flujo de juego (finalización de ronda o de juego)
  - Bloque de decisiones y opciones (decide, select)
  - Dados
  - Mazos (barajar, robar, descartar, mirar componentes, buscar componentes...)
  - Reservas (barajar, robar, devolver, mirar componentes, buscar componente...)
- Gestión de memoria

#### **4.2.9. Planificación**

Se definirán las diferentes etapas, metas o hitos a alcanzar, plazos de entrega y cronogramas o gráficos de programación correspondientes Diagrama de gant, planner. Rango de un año poco más.

#### **4.2.10. Orden de prioridad de documentos básicos**

Si no se indica será el siguiente:

- Planos
- Pliego de Condiciones
- Presupuesto
- Memoria

### **4.3. Anexos**

Se compone de los siguientes documentos

- Documentación de partida (doc. para generar los requisitos, reglamentos de juegos)
- Cálculos. (justificación de soluciones: gestión de memoria, tamaño de tipos, algoritmos)
- Anexos de aplicación en el ámbito del Proyecto. (licencias, seguridad)
- Otros documentos: (Juegos desarrollados)

## **4.4. Planos**

Índice que hará referencia a cada uno de ellos, indicando su ubicación, con el fin de facilitar su utilización.

- Diagramas de casos de uso
- Diagrama conceptual de clases
- Diagrama de clases
- ... (ver libro UML)

## **4.5. Pliego de condiciones**

Condiciones que deben darse para que el objetivo del proyecto pueda materializarse

## **4.6. Estado de mediciones**

Desglose del proyecto en unidades de obras

## **4.7. Presupuesto**

- Cuadro de precios unitarios de materiales, mano de obra y elementos auxiliares.
- Cuadro de precios unitarios de materiales, mano de obra y elementos auxiliares por unidades de obras.
- Presupuesto: valoración económica global, desglosada y ordenada según el Estado de mediciones.

Aspectos a considerar

- Gastos generales y beneficio industrial
- Impuestos, tasas y otras contribuciones
- Seguros
- Costes de certificación y visado

- Permisos y licencias
- Cualquier otro concepto que influya en el coste final de materialización del objeto del Proyecto.

#### **4.8. Estudios con entidad propia**

Documentos requeridos por exigencias legales.