

# COFFEE CHALLENGE: UN JUEGO PARA EL APRENDIZAJE DE PATRONES DE DISEÑO DE SOFTWARE

Luz Eliana González Castaño<sup>1</sup>, Sara Valentina Marroquín Soto<sup>2</sup>, Grissa Vianney Maturana González<sup>3,5</sup>, Roberto Antonio Manjarrés Betancur<sup>4,5</sup>

<sup>1</sup> Ingeniera Informática. Politécnico Colombiano Jaime Isaza Cadavid, luz\_gonzalez82152@elpoli.edu.co.

<sup>2</sup> Ingeniera Informática. Politécnico Colombiano Jaime Isaza Cadavid sara\_marroquin82141@elpoli.edu.co

<sup>3</sup> M.Sc. Ingeniería - Sistemas e Informática. Universidad Nacional de Colombia. gvmatura@unal.edu.co.

<sup>4</sup> M.Sc. Ingeniería - Sistemas e Informática, Profesor Asociado. Facultad de Ingenierías, Politécnico Colombiano Jaime Isaza Cadavid, Colombia.ramanjar@elpoli.edu.co, ramanjar@unal.edu.co

<sup>5</sup> Departamento de Ciencias de la Computación y la Decisión, Universidad Nacional de Colombia, Colombia

## RESUMEN

En ingeniería de software el aprendizaje basado en juegos ayuda a incrementar la efectividad del aprendizaje en el aula de clase y permite poner en práctica conocimientos adquiridos. Los patrones de diseño son soluciones reusables a problemas comunes en el diseño de software, permiten crear software limpio y mantenible, sin embargo, para aplicar los patrones de diseño correctamente se requiere experiencia, por lo que es difícil aprenderlos. En este artículo se investigaron los patrones de diseño, su estructura y posibles aplicaciones, además de revisar juegos educativos existentes en el ámbito de la computación. Posteriormente se desarrolla un juego para enseñar patrones de diseño seleccionados según la popularidad del patrón y su posible aplicación en una página web. Se realizan pruebas con estudiantes del área de ingeniería de software y afines, y se discuten los resultados obtenidos en las sesiones piloto realizadas.

**Palabras clave:** Patrones de diseño; aprendizaje basado en juegos; ingeniería de software; diseño de software

Recibido: 9 de febrero de 2021. Aceptado: 25 de abril de 2021

Received: February 9, 2021. Accepted: April 25, 2021

DOI: <http://dx.doi.org/10.33571/rpolitec.v17n33a3>

## Coffee challenge: a game for teaching software design patterns

## ABSTRACT

*Game-based learning related to software engineering aims to increase learning effectiveness in the classroom and allow knowledge acquired to be put into practice. Design patterns are reusable solutions to recurring problems in software design, creating clean and maintainable software. However, applying design patterns correctly requires experience, therefore, it is difficult for students to learn design patterns. In this article, the design patterns, their structure and possible applications were investigated, in addition to reviewing existing educational games in computing. Afterwards, we develop a game for teaching design patterns depending on the popularity of the pattern and its application to a web page. We evaluated the game with computer science students. Finally, we exhibit the results obtained by test pilot sessions.*

**Keywords:** Design Patterns; Game-based learning; software engineering; software design

Cómo citar este artículo: L.E. González-Castaño, S.V. Marroquín Soto, G.V. Maturana Gonzalez, R.A. Manjarrés Betancur. "Coffee challenge: un juego para el aprendizaje de patrones de diseño de software", Revista Politécnica, vol. 17, no.33 pp.36-46, 2021. DOI: <https://doi.org/10.33571/rpolitec.v17n33a3>

---

## 1. INTRODUCCIÓN

No importa que tan específico sea un problema durante el diseño de software, es probable que alguien ya enfrentó un problema similar antes, por lo que puede existir una solución. Los patrones de diseño son soluciones reusables a problemas que se presentan repetidamente durante el diseño de software, permitiendo a los desarrolladores utilizarlos para resolver problemas específicos y ayudando a una mejor comprensión del diseño [1].

El aprendizaje basado en juegos es una integración de juegos en el proceso de aprendizaje utilizado para proporcionar a los educadores una forma de motivar a los estudiantes a aprender; adquirir conocimientos prácticos; aumentar la eficacia del aprendizaje a través de inmersión en el material; y animar a los alumnos a cometer errores y aprender de ellos [2].

La ingeniería de software es una disciplina que aborda soluciones de valor a problemas complejos, lo que requiere una combinación de conocimiento práctico y teórico. En este sentido, los juegos pueden considerarse como herramienta para capacitar a los estudiantes en el área de ingeniería de software y afines, y ayudarlos a adquirir conocimientos prácticos [3]. Los juegos del área de ingeniería de software se destacan dentro de juegos existentes en ciencias de la computación [4].

En la literatura, un gran número de patrones de diseño de software se identifican y documentan para estandarizar soluciones, además de crear novedosas herramientas de desarrollo para su reconocimiento y aplicabilidad [5]. Gestwicki y Sun [6] implementaron un video juego que incluía diversos patrones de diseño en su código para que los estudiantes puedan aprender a través del análisis del código, siendo esta la experiencia más cercana encontrada acerca de patrones de diseño en juegos, aunque los patrones sean un tema importante en ingeniería de software.

Diversos autores afirman que el uso de patrones de diseño conduce a la construcción de software bien estructurado, mantenible y reutilizable, por lo cual es conveniente y útil aplicarlos en el diseño de software, sin embargo, conocer patrones no es suficiente, se debe considerar que patrón usar en un escenario y si es necesario o no usarlo, estudios empíricos muestran que los patrones de diseño a menudo se usan incorrectamente llevando a un impacto negativo en la calidad del software, además es muy difícil para los estudiantes aplicar los patrones en el diseño de software y utilizarlos como soluciones a diversos problemas [5],[7],[8],[9].

Con este artículo se pretende crear un juego para estudiantes en el área de ingeniería de software y afines, en donde puedan comprender patrones de diseño de software y conocer posibles escenarios para aplicarlos, brindando un enfoque diferente para la enseñanza de patrones de diseño en donde los estudiantes recuerden fácilmente los patrones. El documento se organiza de la siguiente manera: inicialmente se encuentran los antecedentes; luego se presenta el diseño del juego; más adelante están los resultados en donde se muestra cómo se evaluó el juego por medio de sesiones de pruebas, a partir de ahí se realiza el análisis de resultados y por último se exponen las conclusiones y trabajo futuro.

## 2. ANTECEDENTES

Los patrones de diseño de software son un conjunto de artefactos que encapsulan el conocimiento de los problemas de diseño que ocurren en un contexto particular. Se han propuesto varios patrones de software para proporcionar soluciones a problemas de diseño recurrentes [10]. *Gang of Four* (GoF) [11] clasifica estos patrones en 3 grandes categorías basadas en su propósito: creacionales, estructurales y de comportamiento.

- Creacionales: Son los que facilitan la tarea de creación de nuevos objetos, de tal forma que se desacople el proceso de creación de la implementación del resto del sistema. Entre estos se encuentran: *abstract factory*, *builder*, *factory method*, *prototype*, *singleton*.
- Estructurales: Describe la forma en que se componen objetos para obtener nueva funcionalidad y añade la flexibilidad de cambiar la composición en tiempo de ejecución. Algunos de estos patrones son: *adapter*, *bridge*, *composite*, *decorator*, *facade*, *flyweight*, *proxy*.
- Comportamiento: Proporcionan una solución para estructurar datos y los objetos para así organizar el procesamiento entre objetos. Algunos ejemplos de estos patrones son: *chain of responsibility*, *command*, *interpreter*, *iterator*, *mediator*, *memento*, *observer*, *state*, *strategy*, *template method*, *visitor*.

Cabe destacar que los patrones de diseño mencionados anteriormente no son los únicos, ya que cualquier desarrollador puede crear su propio patrón de diseño y estandarizar como una solución reusable de un problema de software [12],[13],[14].

Aunque los patrones GoF son los más populares existen también otros tipos de patrones como los patrones de interfaz. Los patrones de diseño de interfaz de usuario se utilizan para facilitar el desarrollo de aplicaciones y plataformas, porque estas usan una serie de patrones que las hacen similares y por ende intuitivas [15]. Entre ellos se encuentran diferentes patrones de diseño tipo *User Interface* (UI) como: acordeón, asistente, barra de progreso, búsqueda, carrusel, *dashboard*, diálogos, formularios, galería, indicadores de carga, listas, *login*, navegación por etiquetas, notificaciones, menú *dropdown*, segmentos de control, tabla, entre otros [16].

En los últimos años, ha aumentado el interés por examinar el uso de juegos en la educación superior, buscando que los alumnos desarrollen competencias; dichos métodos persiguen los propósitos de integración de sistemas de evaluación innovadores y alternativos que se adapten a los cambios que la sociedad presenta [17]. Algunos juegos educativos tienden a ser exploratorios, esto permite que los alumnos sientan motivación y gratificación. La curiosidad parece ser el componente principal de dicha motivación, de esta forma se atrae la atención de los alumnos para buscar activamente diversas fuentes de desafío en el juego [18].

Algunos autores exploran los beneficios de aprendizaje que ofrece el aprendizaje basado en juegos (*Game Based Learning GBL*), que puede definirse como el uso de tecnología basada en juegos para brindar, apoyar y mejorar la enseñanza, el aprendizaje y evaluación [19]. Los juegos basados en experiencias, también son utilizados como herramienta pedagógica centrada en los alumnos, los cuales encuentran los beneficios de aprender haciendo, al aumentar la velocidad de aprendizaje y facilitar el desarrollo de habilidades sociales como la comunicación; habilidades de negociación y trabajo en equipo; ya que dichas habilidades son difíciles de comprender desde la teoría [20]. Para evaluar el juego además de las preguntas abiertas propuestas en la validación descrita en los juegos basados en experiencias, se utiliza la encuesta propuesta en el MEEGA+ (Modelo para la Evaluación de Juegos Educativos) [21], un método de evaluación para juegos educativos, la cual es una versión mejorada de su predecesor MEEGA. La encuesta tiene como objetivo evaluar la calidad de los juegos educativos utilizados como estrategia para la educación informática y, así, contribuir a su mejora y la adopción efectiva y eficiente en la práctica. Este método fue revisado por expertos en informática y/o estadística, además de ser evaluado con estudiantes de cursos de ciencias de la computación.

Los juegos son una alternativa para la enseñanza de ingeniería de software, ya que pueden aumentar la efectividad y el compromiso del aprendizaje [4]. Algunos juegos para enseñanza en ingeniería de software son:

- "Metrics War" es un juego para estudiantes de SE. Este juego es para 2 a 4 jugadores, que utilizan métricas de software para lograr la calidad requerida por un producto de software. El jugador recibe una tarjeta de objetivo, que indica un hipotético sistema de software a desarrollar. Cada carta de objetivo exige un total de 50 puntos, e indica cuántos puntos el jugador debe lograr de la calidad del software para ganar el juego. [22]
- "SimulES-W" es un juego de mesa colaborativo, utilizado para enseñar conceptos de un proceso de ingeniería de software. Los jugadores realizan diferentes roles, como ingeniero de software, coordinador técnico, gerente de proyecto y controlador de calidad. Los jugadores trabajan con el presupuesto, la contratación de ingenieros de software y la construcción de diferentes artefactos del software. [23]
- "SEMAT GAME" juego para el aprendizaje de la gestión de proyectos en la ingeniería de software por medio del kernel de Semat que se describe mediante un pequeño conjunto de elementos, que se agrupan en tres áreas de interés: cliente, solución y esfuerzo. Cada área de preocupación se centra en un aspecto específico de la ingeniería de software y se distingue por un color, consiguiendo agrupar los elementos del kernel de Semat.[24]
- "RobotON", es dirigido a jugadores que nunca antes han programado. A diferencia de otros juegos de programación serios, Robot ON! se centra en la comprensión más que en los desafíos de resolución de problemas; los jugadores en realidad no escriben ningún programa, sino que se les asigna la tarea de demostrar su conocimiento y comprensión del comportamiento de un programa. Este incluye herramientas que permiten a los jugadores demostrar comprensión de valores de variables, tipos de datos, declaraciones de programas y flujo de control. [25]
- "RoboBUG" es un juego serio que tiene como público objetivo estudiantes de informática que están aprendiendo a depurar errores de código por primera vez. Es un tipo de rompecabezas, ya que se ha

demostrado que los rompecabezas son efectivos tanto para ayudar para aprender material y demostrar conceptos de nivel superior como el pensamiento crítico y la resolución de problemas [26].

### 3. DISEÑO DEL JUEGO

Para el proceso de diseño del juego se utilizan los pasos propuestos por Gómez [20], donde describe un método para diseño de juegos basados en experiencias, el cual consta de diez tareas que cubren desde la identificación de la temática que se quiere llevar a formato de juego hasta las pruebas piloto y elaboración de la encuesta de evaluación a aplicar a los participantes. A continuación, se enlistan las tareas planteadas en el método y los resultados de realizar dicha actividad en el desarrollo del juego propuesto:

- Identificar la temática del juego: Definición y aplicación de patrones de diseño de software.
- Establecer el propósito del juego: Enseñanza de los conceptos y aplicación de los patrones de diseño de software.
- Plantear los objetivos instruccionales del juego:
  - Caracterizar el posible uso de patrones de diseño en las actividades del proceso de diseño y desarrollo de software.
  - Identificar la existencia de patrones de diseño en un proyecto de desarrollo de software.
- Identificar y definir conceptos generales de la temática:
  - Soluciones: Los patrones de diseño se usan como respuestas eficaces a los problemas.
  - Problemas: Requisitos y necesidades encontradas durante el diseño y desarrollo del software.
- Seleccionar técnicas candidatas: Se toman los objetivos (caracterizar e identificar), y las palabras clave (soluciones y problemas) en su forma básica. Se observa que las palabras problema y solución son palabras claves de las técnicas: Unlock y El Dorado. por tanto, se convierten en las técnicas candidatas seleccionadas para el diseño del juego.
- Seleccionar la(s) técnica(s) más apropiada(s): Se realizan una serie de preguntas de importancia para la caracterización del juego, en la Tabla 1 se presentan las preguntas realizadas y el puntaje para cada técnica candidata:

Tabla 1. Preguntas de caracterización de las técnicas candidatas.

Pregunta	UNLOCK		EL DORADO	
	Respuesta	Puntaje	Respuesta	Puntaje
¿Es un juego colaborativo?	Si	3	Si	3
¿Existen diferentes caminos para encontrar la solución correcta?	No	0	Si	3
¿Se busca resolver un problema?	Si	3	Si	3
¿Hay una historia involucrada a lo largo del juego?	Si	1	Si	3
¿Hay penalidades cuando se realiza una decisión incorrecta?	Si	3	Si	3
¿El tiempo es un factor importante al final de juego?	Si	3	Si	3
Cuando se hace una decisión, ¿Los beneficios son explicados?	No	0	Si	3
	<b>Total:</b>	<b>13</b>	<b>Total:</b>	<b>21</b>

El cociente de puntaje máximo sería entonces para Unlock:  $13/21 = 0.61$ , en este caso debido a que el cociente es menor a 0.85 se descarta la técnica candidata. Por otro lado, el cociente de puntaje máximo para El Dorado sería:  $21/21 = 1$ , al ser mayor que 0.85 la técnica candidata se convierte en la técnica sugerida por el método.

- Incorporar conocimiento específico en el juego: Se toma como base la plantilla técnica sugerida por el método para el juego El Dorado, la cual se modifica para describir el juego objetivo e integrar el

componente educativo, cambiando la metáfora del juego a tener como objetivo: Resolver por medio de patrones de diseño de software los requisitos necesarios para diseñar una página web de venta de productos de café. En la tabla 2 se presenta la plantilla técnica del juego objetivo.

- Desarrollar sesiones piloto del juego: Se realizan 2 pruebas preliminares del juego con los asesores del proyecto, con lo cual han realizado las siguientes recomendaciones:
  - Corrección del idioma: El juego está en idioma inglés, se realizaron mejoras a la traducción de la intención de varias cartas del juego.
  - Mejorar la calidad de la imagen: En algunas cartas los diagramas UML explicativos se veían borrosos, se aumenta la calidad de las imágenes en las cartas.
  - Explicar acrónimos utilizados en el juego: Se utilizan los acrónimos GoF y UI, se explica su significado en las cartas del juego.

Tabla 2. Plantilla técnica *Coffee Challenge*.

PLANTILLA TÉCNICA	
<b>GENERALIDADES</b>	
Nombre Técnica	Design Patterns for Coffee Challenge
Objetivo del Juego	Resolver por medio de patrones de diseño de software los requisitos necesarios para diseñar una página web de venta de productos de café.
Público objetivo	Estudiantes del área de Ingeniería de software y afines
<b>COMPONENTE EDUCATIVO</b>	
Nombre de la Temática	Patrones de diseño de software
Propósito	Enseñanza
Conceptos básicos de la temática	Soluciones, problemas
<b>REGLAS DEL JUEGO</b>	
1	No desordenar las cartas.
2	No voltear las cartas hasta que las indicaciones lo digan.
4	Marcar reloj como penitencia cada vez que el juego lo indique.
5	Tomar decisiones como equipo.
6	Tomar únicamente las cartas que el juego indique.
<b>CRITERIO DE SELECCIÓN DEL GANADOR</b>	
Este es un juego cooperativo, todo el equipo gana cuando se implemente la página web objetivo. Al competir contra otros equipos gana el que termine la página web en el menor tiempo posible.	

- Consolidar el juego: Se hacen las respectivas correcciones, sin embargo como las correcciones realizadas eran de mejora de contenido la plantilla técnica original no cambia.
- Elaborar encuesta de evaluación del juego: Para la encuesta de validación del juego se crean dos encuestas diferentes:
  - Pre Juego: se realiza una encuesta de validación de conocimientos para revisar los conocimientos previos de los participantes antes del juego, en esta encuesta se incluyen preguntas demográficas para diferenciar el público participante.
  - Post Juego: Se utiliza la encuesta MEEGA+ para evaluar la usabilidad y la experiencia del jugador con el juego, además se realizan las mismas preguntas con que se evaluaron los conocimientos inicialmente para obtener resultados en donde se valide el nivel de aprendizaje.

Para desarrollar el juego después de elegir la técnica candidata se eligieron algunos patrones de diseño, para ello, se toman en cuenta los patrones de diseño GoF más usados [27]. Los patrones de diseño de interfaz usuario se eligieron según su posible aplicación, en este caso los más acordes al diseño de una página web de venta. Luego de seleccionar los patrones se ajustaron posibles requisitos de la página web para que concuerden con problemas que pudieran solucionar alguno(s) de los patrones seleccionados. Posteriormente, se relacionan los conceptos del juego al implementar una metáfora de la realidad que justifica las actividades que se realizan a lo largo del juego. Adicionalmente, se construyen diagramas UML para ejemplificar los problemas y soluciones asociados a los patrones presentados a lo largo del juego.

La narrativa del juego tiene una estructura lineal, por ello, la carta inicial explica a los jugadores que no deben voltear ni tomar cartas a menos que el juego lo indique. Las cartas deben estar ordenadas como lo indica el número en la esquina superior derecha, de esta forma el jugador toma decisiones mientras va avanzando a lo largo de la historia. En la metáfora del juego, los jugadores hacen el papel de diseñadores de software, inicialmente se describe un concurso ficticio para que grupos de personas puedan demostrar sus habilidades como diseñadores de software, el reto de dicho concurso es aplicar de la mejor forma los patrones de diseño de software, para esto se plantea diseñar una página web de venta de productos de café en donde el sistema autoriza diferentes operaciones como vista del catálogo, compra de productos, gestión de las órdenes y gestión de los clientes.

A medida que los estudiantes avanzan carta por carta se les explica la metáfora del juego, es decir como asumirá su rol cada uno y cuál es el objetivo del juego, se va construyendo la página web y se explica el sistema de penalidades, cada vez que se implemente mal un patrón se castigará con tiempo que será sumado al final del juego. Cada carta de explicación y requisitos en la parte de abajo tienen indicaciones sobre qué hacer a continuación: seguir avanzando, elegir un patrón acorde o marcar o borrar un reloj, que es la indicación de penalidades de tiempo. Gracias a la metáfora del juego se propone el nombre del juego: *Coffee Challenge*. El juego se crea en inglés debido a que es un idioma fundamental en el ámbito profesional y educativo en ciencias de la computación, además la mayoría de la documentación encontrada acerca de patrones de diseño de software y juegos de ingeniería de software está en inglés.

En el juego hay dos elementos básicos: Patrones de diseño y requisitos, los patrones de diseño se separan en dos grupos:

- Patrones de diseño expuestos en 1994 por *Gang of Four* [11], los cuales desataron la popularidad de los patrones de diseño con su libro en el que exponen 23 patrones de diseño clásicos que aún están vigentes en el diseño de software, en la Fig 1. se muestra el diseño que representa este tipo de patrones en el juego:

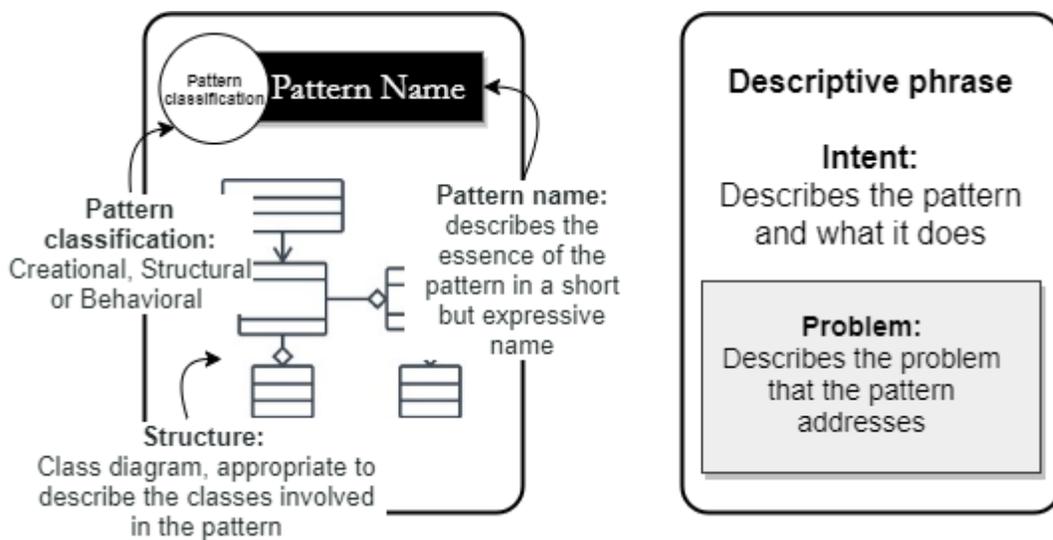


Figura 1. Representación de patrones de diseño GoF

- Patrones de diseño de interfaz, los cuales describen problemas comunes de interfaz de usuario y su solución, ayudan a que la interfaz sea usable y comprensible para el usuario final. En la Fig 2. se muestra el diseño que representa este tipo de patrones en el juego:

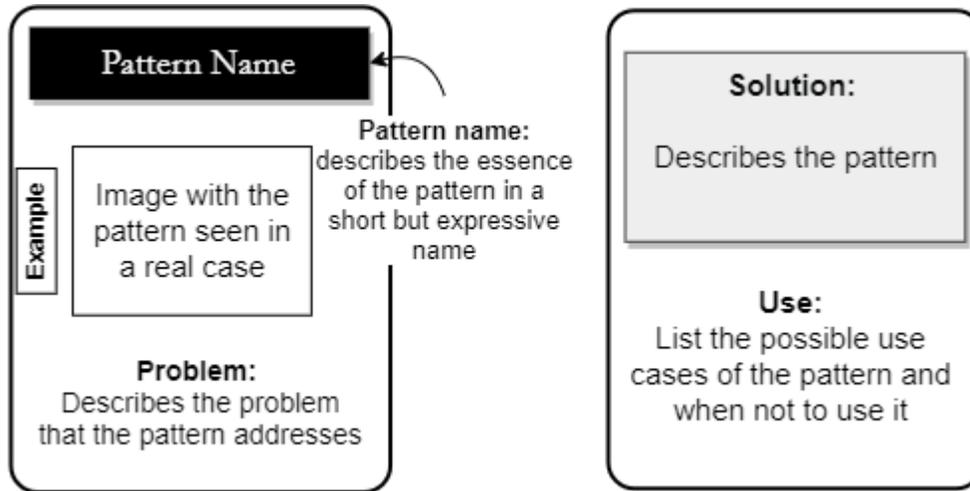


Figura 2. Representación patrones de diseño de interfaz

Por otro lado, los requisitos contienen una frase explicativa describiendo una necesidad, los diseñadores deberán seleccionar el patrón de diseño más adecuado para resolver el requisito. Al reverso de cada requisito se encuentra él o los patrones adecuados para solucionarlo; el diagrama UML del patrón resolviendo el problema y una descripción de porque puede resolverlo.

Al aplicar un patrón de interfaz a un requisito en vez de ver el diagrama UML aplicado, se tiene la oportunidad de aplicar dicho patrón a la interfaz del producto final, por lo que cada requisito de interfaz de usuario permite tomar una carta que representa una parte de la página web terminada, al final del juego es posible armar tres interfaces diferentes de la página web.

#### 4. VALIDACIÓN DEL JUEGO DISEÑADO

Para realizar las sesiones de validación del juego objetivo se utiliza la plataforma *Tabletopia*®, la cual es una plataforma digital para publicar juegos que simula el ambiente de un juego de mesa de manera virtual y permite jugar con varias personas al mismo tiempo. La validación se realiza con estudiantes de ingeniería informática mayormente del Politécnico Colombiano Jaime Izasa Cadavid, se prueba el juego con 5 grupos de 3 a 6 estudiantes y 8 estudiantes que jugaron de manera individual, la mayoría de estudiantes están entre 5to y 6to semestre, rara vez juegan juegos no digitales y tienen un rango de edad entre 18 y 28 años, los que jugaron de manera individual sí pudieron completar el juego en totalidad. A continuación, se presentan los resultados de las personas que jugaron en grupo, primeramente en la encuesta Pre-juego:

- 93% de los estudiantes tienen una idea de los que es un patrón de diseño y para qué sirve.
- 45% de los estudiantes pueden nombrar patrones de diseño propuestos por GoF.
- 53% de los estudiantes conocen la clasificación de los patrones de diseño propuesta por GoF.

Luego de terminar la sesión de juego se envía una encuesta post juego a los mismos estudiantes en donde pueden dar sus opiniones de la experiencia del jugador, la usabilidad y que tanto aprendieron con el juego, a continuación de presentan los resultados:

En el componente de aprendizaje:

- 100% de los estudiantes lograron explicar que es un patrón de diseño de software y para que se utiliza.

- 
- 100% de los estudiantes nombraron al menos 3 patrones de diseño propuestos por GoF y conocen la clasificación de dichos patrones.

#### Resultados de la encuesta de usabilidad:

- Estética:
  - El diseño del juego es atractivo (interfaz, gráficos, etc.):
    - 42.8 % - Muy de acuerdo
    - 42.8 % - De acuerdo
    - 14.4 % - Ni de Acuerdo ni desacuerdo
  - Los textos, colores y fuentes combinan y son consistentes.
    - 57.2 % - Muy de acuerdo
    - 42.8 % - De acuerdo
- Capacidad de aprendizaje
  - No fue necesario aprender muchas cosas para poder empezar a usar el juego.
    - 21.5 % - Muy de acuerdo
    - 42.8 % - De acuerdo
    - 28.6 % - Ni de Acuerdo ni desacuerdo
    - 7.1% - En desacuerdo
  - Aprender a jugar este juego fue fácil para mí.
    - 57.2 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 21.4 % - Ni de Acuerdo ni desacuerdo
  - Creo que la mayoría de las personas pueden aprender a jugar con este juego rápidamente.
    - 28.6 % - Muy de acuerdo
    - 50 % - De acuerdo
    - 21.4 % - Ni de Acuerdo ni desacuerdo
- Operabilidad
  - Creo que el juego es fácil de jugar.
    - 64.3 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 14.3 % - Ni de Acuerdo ni desacuerdo
  - Las reglas del juego son claras y comprensibles.
    - 35.7 % - Muy de acuerdo
    - 50 % - De acuerdo
    - 14.3 % - Ni de Acuerdo ni desacuerdo
- Accesibilidad
  - Las fuentes (tamaños y estilo) utilizadas en el juego son legibles.
    - 57.1 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
    - 7.1 % - Muy en desacuerdo
  - Los colores utilizados en el juego son comprensibles.
    - 64.3 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
- Confianza
  - El contenido y la estructura del juego me dan confianza suficiente para reconocer que con el uso del juego puedo aprender algo.
    - 57.2 % - Muy de acuerdo
    - 35.7 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
- Desafío
  - El juego me plantea retos apropiados.
    - 64.3 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
  - El juego ofrece nuevos desafíos (obstáculos, situaciones, variaciones) a un ritmo adecuado.

- 57.2 % - Muy de acuerdo
  - 35.7 % - De acuerdo
  - 7.1 % - Ni de Acuerdo ni desacuerdo
- El juego no se torna monótono en sus tareas (repetitivo o con tareas aburridas).
  - 50 % - Muy de acuerdo
  - 28.6 % - De acuerdo
  - 7.1 % - Ni de Acuerdo ni desacuerdo
  - 14.3 % - En desacuerdo
- Satisfacción
  - Completar las tareas del juego me permitió obtener una sensación de logro.
    - 50 % - Muy de acuerdo
    - 35.7 % - De acuerdo
    - 14.3 % - Ni de Acuerdo ni desacuerdo
  - Gracias a mi esfuerzo personal he avanzado en el juego.
    - 35.7 % - Muy de acuerdo
    - 42.9 % - De acuerdo
    - 21.4 % - Ni de Acuerdo ni desacuerdo
  - Me siento satisfecho con las cosas que he aprendido con el juego.
    - 64.3 % - Muy de acuerdo
    - 35.7 % - De acuerdo
  - Recomendaría el uso del juego a mis amigos.
    - 57.1 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 14.3 % - Ni de Acuerdo ni desacuerdo
- Interacción social
  - He tenido la oportunidad de interactuar con los demás durante el juego.
    - 78.6 % - Muy de acuerdo
    - 21.4 % - De acuerdo
  - El juego promueve momentos de cooperación y/o competición entre los jugadores.
    - 71.4 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 7.2 % - Ni de Acuerdo ni desacuerdo
  - Me he divertido con otras personas durante el juego.
    - 42.8 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 28.6 % - Ni de Acuerdo ni desacuerdo
- Diversión
  - Me he divertido.
    - 50 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 14.3 % - Ni de Acuerdo ni desacuerdo
    - 7.1 % - En desacuerdo
  - Durante el juego hubo algo (elementos del juego, competición, etc.) que me hizo sonreír.
    - 42.9 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 35.7 % - Ni de Acuerdo ni desacuerdo
- Atención enfocada
  - Había algo interesante al comienzo del juego que me llamó la atención.
    - 57.1 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
    - 14.4 % - En desacuerdo
  - Mientras estaba concentrado en el juego, el tiempo ha pasado rápidamente.
    - 64.3 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - En desacuerdo
  - Me he sentido totalmente inmerso en la atmósfera del juego, olvidándose incluso de lo que había a mi alrededor.
    - 42.9 % - Muy de acuerdo

- 21.4 % - De acuerdo
  - 28.6 % - Ni de Acuerdo ni desacuerdo
  - 7.1 % - En desacuerdo
- Relevancia
  - El contenido del juego es relevante para mis intereses.
    - 71.4 % - Muy de acuerdo
    - 21.4 % - De acuerdo
    - 7.2 % - Ni de Acuerdo ni desacuerdo
  - El contenido del juego está relacionado con los conocimientos y contenidos de la asignatura.
    - 64.3 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
  - El uso del juego es un método de aprendizaje apropiado para esta asignatura.
    - 78.6 % - Muy de acuerdo
    - 21.4 % - De acuerdo
  - Considero que el uso de este juego es más apropiado para mi aprendizaje que otros métodos dentro de la asignatura.
    - 57.2 % - Muy de acuerdo
    - 7.1 % - De acuerdo
    - 28.6 % - Ni de Acuerdo ni desacuerdo
    - 7.1 % - En desacuerdo
- Aprendizaje percibido
  - El juego contribuyó a mi aprendizaje en la asignatura.
    - 64.3 % - Muy de acuerdo
    - 28.6 % - De acuerdo
    - 7.1 % - Ni de Acuerdo ni desacuerdo
  - En comparación con otras actividades del curso, el juego ha sido eficaz en mi aprendizaje.
    - 57.1 % - Muy de acuerdo
    - 14.3 % - De acuerdo
    - 28.6 % - Ni de Acuerdo ni desacuerdo
  - El juego contribuye a conocer patrones de diseño de software
    - 71.4 % - Muy de acuerdo
    - 28.6 % - De acuerdo
  - El juego contribuye a saber aplicar patrones de diseño de software
    - 64.3 % - Muy de acuerdo
    - 35.7 % - De acuerdo

## 5. ANÁLISIS DE RESULTADOS

En general el juego tuvo una buena aceptación por parte de los estudiantes. Todos los estudiantes estuvieron de acuerdo con que el juego se acercaba a la realidad ya que representaban requisitos de una página web de comercio real y simulaba la toma de decisiones en equipo, estuvieron de acuerdo con que el juego ayuda a conocer y aplicar los patrones y les plantea retos interesantes. En retroalimentación después de las sesiones los estudiantes manifestaban que fue una experiencia positiva en declaraciones como "Se puede comprender mejor el concepto de patrón de diseño al presentar una aplicación real".

Al inicio del juego 45% de los estudiantes conocían algunos patrones GoF y al finalizar el 100% pudo nombrar al menos 3 patrones GoF, se refleja por las preguntas respondidas que el aprendizaje fue exitoso hasta el porcentaje completado del juego.

Tan solo un estudiante en el grupo de los que jugaron individualmente conocía los patrones de diseño de interfaz al inicio del juego y al terminar, todos pudieron nombrar más de 3 de estos tipos de patrones, a diferencia de los que jugaron en grupo que no pudieron completar el juego, por tanto no vieron ningún patrón de interfaz los cuales se presentan en la segunda mitad del juego, al final de la sesión de juego se les comparte el link para que puedan completar el juego si desean, por los resultados obtenidos tan solo dos de los estudiantes que jugaron en grupo completaron el juego debido que al inicio no conocían patrones de

interfaz y al llenar la encuesta pudieron nombrar 3, el resto de los estudiantes no pudieron nombrar ningún patrón de interfaz.

Se aprecia que los estudiantes que jugaron en grupo se divertieron durante el juego, a diferencia de los estudiantes que jugaron en solitario, al no tener interacción social el juego se les hacía monótono.

Las sugerencias presentadas de posibles cambios al juego entre los que jugaron individualmente y en grupo fueron orientadas a la plataforma y el idioma, no hubo críticas del contenido y mecánicas del juego. Se evidencia que al describir claramente los patrones de diseño y luego presentar el requisito los estudiantes eligen el patrón adecuado a la necesidad planteada y entienden que al aplicar malas soluciones se pierde tiempo en el diseño de software.

## 6. CONCLUSIONES Y TRABAJO FUTURO

Los estudiantes pueden comprender de una manera directa los conceptos esenciales de algunos patrones de diseño y pueden discutir de su correcta aplicación mientras van conociéndolos. Sin embargo, se nota una barrera con el idioma, al ser un juego hecho en inglés y aplicado con estudiantes de habla hispana no podían comprender los conceptos explicados en su totalidad, además debido a una pandemia mundial no se pudo probar el juego de mesa de manera presencial, por lo que, al tener una interacción virtual, la participación de los estudiantes se hacía difícil, ningún grupo de estudiantes pudo llegar más lejos de 41% del juego en 75 minutos aproximadamente. Por otro lado, los que jugaron individual completaron el 100% del juego en aproximadamente en 100 minutos.

Como trabajo futuro se plantea implementar el juego en diferentes idiomas para hacerlo amigable con los estudiantes y no tener la barrera del idioma en el aprendizaje. Además, es posible modificar fácilmente la metáfora del juego según las necesidades de enseñanza para que tenga otro dominio diferente a una página web de venta de café. El reto del juego se puede aplicar a otros productos, como diseñar una red social o un juego online, de esta forma se pueden aplicar patrones diferentes, ya sea al agregar o reemplazar patrones para cumplir los objetivos del nuevo reto a resolver.

## 7. AGRADECIMIENTOS

Se agradece a Daniel Correa Botero, doctor en ingeniería de software y a Juan Felipe Quintana Gómez con diez años de experiencia trabajando en ingeniería de software por su ayuda al definir posibles necesidades y requisitos en el diseño de la página web y seleccionar los mejores patrones de diseño que pueden solucionar dichas necesidades. Nuestra gratitud también a cada docente quienes con su apoyo y enseñanza constituyen la base de nuestra vida profesional, y a los estudiantes que nos ayudaron a probar el juego. Se agradece a Colciencias por la financiación otorgada en la convocatoria 785 de 2017 y Sapiencia, quienes son patrocinadores de la asesora de este artículo Grissa Maturana.

## 8. REFERENCIAS BIBLIOGRÁFICAS

[1] Mu H. y Jiang S. (2011), "Design patterns in software development," 2011 IEEE 2nd International Conference on Software Engineering and Service Science, pp. 322-325, doi: 10.1109/ICSESS.2011.5982228.

[2] Vlachopoulos, D y Makri, A. (2017). The effect of games and simulations on higher education: a systematic literature review. *Int J Educ Technol High Educ* 14, 22 . <https://doi.org/10.1186/s41239-017-0062-1>

[3] Kosa, M., Yilmaz, M., O'Connor, R., & Clarke, P. (2016). Software engineering education and games: a systematic literature review. *Journal of Universal Computer Science*, 22(12), 1558-1574.

[4] Battistella, P. y C. V. Wangenheim. (Marzo 2016) "Games for Teaching Computing in Higher Education A Systematic Review" in *IEEE Technology and Engineering Education*, vol. 1, no. 3, pp. 8-30.

- 
- [5] Wang Y., Zhang C. y Wang F. (2018), "What Do We Know about the Tools of Detecting Design Patterns?," 2018 IEEE International Conference on Progress in Informatics and Computing (PIC), pp. 379-387, doi: 10.1109/PIC.2018.8706318.
- [6] Gestwicki P., y Sun F.S. (2008) "Teaching Design Patterns Through Computer Game Development", in ACM Journal on Educational Resources in Computing, pp. 1-21. <https://doi.org/10.1145/1348713.1348715>
- [7] Lotlikar P., and Wagh R. (2016) "Using POGIL to Teach and Learn Design Patterns-A Constructionist Based Incremental, Collaborative Approach", in 2016 IEEE 8th International Conference on Technology for Education Using, Mumbai, pp. 46-49. <https://doi.org/10.1109/T4E.2016.018>
- [8] Zhu H. y Bayley I..(2017) "On the Composibility of Design Patterns: Extended Abstract", in 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), Turin, pp. 369-370. <https://doi.org/10.1109/COMPSAC.2017.177>
- [9] Gómez M., Jiménez G., y Arroyo J. (2009). "Teaching Design Patterns Using a Family of Games", in Proceedings of the 14th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education, Paris, pp. 268-272. [Online]. <https://doi.org/10.1145/1562877.1562960>
- [10] Dwivedi A., Tirkey A., y Rath S. (2018), "Software design pattern mining using classification-based techniques". *Front. Comput. Sci.* 12, 908–922. <https://doi.org/10.1007/s11704-017-6424-y>
- [11] Gamma E., Helm R., Johnson R., y Vlissides J. "Design Patterns: Elements of Reusable Object-Oriented Software". Massachusetts: Addison-Wesley Professional, 1994
- [12] Ibarra J., Osuna J., Reyes R., Lepe J. y Peraza A. (2019). "Clasificación de los patrones de diseño idóneos en programación android" 2018 Vol. 3 Núm. 3: Revista ReDTiS.
- [13] Debrauwer L. (2014). "Patrones de diseño en Java: Los 23 modelos de diseño: descripción y solución ilustradas en UML 2 y Java"
- [14] Pressman R.S. y Maxim B. R. (2015) "Software Engineering: A Practitioner's Approach", 8a ed. New York: McGraw-Hill Education.
- [15] Cortes C.A., Olivares B.A., Alor G., L. Rodriguez y Pelaez S.G. (2016). "Análisis comparativo de patrones de diseño de interfaz de usuario para el desarrollo de aplicaciones educativas" *Research in Computing Science* pp. 31–41
- [16] Tidwell J., "Designing Interfaces" (2011). Second ed., O'Reilly Media, pp. 578
- [17] Lopez C., Benedito V. y León M.J. (2016) "The Approach by Competency in University Training and its Impact on Student Evaluation. The Perspective of a Team of Pedagogy Experts" *Form. Univ.* vol.9, no.4, La Serena, <http://dx.doi.org/10.4067/S0718-50062016000400003>.
- [18] Yang J., Chien K., Liu T. (April 2012) "A digital game-based learning system for energy education: an energy conservation pet" 2012, *The Turkish Online Journal of Educational Technology*, vol. 11, Issue 2
- [19] Vlachopoulos D. y Makri A.(2017) "The effect of games and simulations on higher education: a systematic literature review" *International Journal of Educational Technology in Higher Education*, vol. 14, no. 1, pp. 1-33.
- [20] Gómez M.C. (2010) "Definición de un método para el diseño de juegos orientados al desarrollo de habilidades gerenciales como estrategia de entrenamiento empresarial", [Tesis de maestría, Universidad Nacional de Colombia], Repositorio institucional - Universidad nacional de Colombia.
- [21] Petri G., von Wangenheim C.G. y Ferreti A. (2018). "MEEGA+: A Method for the Evaluation of Educational Games for Computing Education", Brazilian Institute for Digital Convergence.

- [22] Silva G. y Travassos V. "Metrics War: A Board Game Proposal for Teaching Software Metrics and Quality Attributes" 2019, Computing Track Short Papers, SBC Proceedings of SBGames 2019
- [23] Suescún E., Toro M., Mazo R., Velasquez D., Vallejo P., Cardona J., Rincón R., Werneck M. y Sampaio do Prado Leite J.C. (2018) "SimulES-W: A Collaborative Game to Improve Software Engineering Teaching" 2018, Comp. y Sist. vol.22 no.3 México jul./sep.. <https://doi.org/10.13053/cys-22-3-2711>
- [24] Zapata C.M., Rojas M.D., Arango R.E. y Jimenez L.D., (2015) "SEMAT GAME: Applying a Project Management Practice", 2015, Developments in Business Simulation and Experiential Learning, 42, 2015, pp. 133–143.
- [25] Miljanovic M.A. y Bradbury J.S., (2016) "Robot ON!: A Serious Game for Improving Programming Comprehension," 2016 IEEE/ACM 5th International Workshop on Games and Software Engineering (GAS), Austin, TX, 2016, pp. 33-36, <https://doi.org/10.1109/GAS.2016.014>.
- [26] Miljanovic M.A. y Bradbury J.S. (2017), "RoboBUG: a serious game for learning debugging techniques" Proceedings of the 2017 ACM Conference on International Computing Education Research (ICER 2017), pp. 93–100, <https://doi.org/10.1145/3105726.3106173>
- [27] Guerrero C.A., Suárez J.M., y Gutiérrez L.E. (January 2013). "Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web", Información tecnológica, vol.24, no. 3, pp. 103-114,