

# Adaptation of the 6D Gamification Model in a Software Development Course

Sebastián Gómez-Jaramillo  
Tecnológico de Antioquia  
Universidad Nacional de  
Colombia  
Medellín, Colombia  
[sgomezja@tdea.edu.co](mailto:sgomezja@tdea.edu.co)

Julián Moreno-Cadavid)  
Universidad Nacional de  
Colombia  
Medellín, Colombia  
[jmoreno1@unal.edu.co](mailto:jmoreno1@unal.edu.co)

Carlos Mario Zapata-Jaramillo  
Universidad Nacional de  
Colombia  
Medellín, Colombia  
[cmzapata@unal.edu.co](mailto:cmzapata@unal.edu.co)

**Abstract**— This article shows a theoretical adaptation of the 6D gamification model, originally designed to be applied in marketing. The model establishes six steps that must be considered when applying gamification, in order to design the activity in which the game elements are going to be used. Each of these steps are described in a general way and considering how it can be applied in an educational context, specifically in the teaching of software development.

**Keywords**—Gamification, education, software development

## I. INTRODUCCIÓN

La gamificación entendida como la implementación de elementos de juego en un contexto de no juego ha sido ampliamente utilizada en diferentes áreas del conocimiento, incluyendo la educación debido a que permite en los estudiantes generar una mayor motivación tanto extrínseca (al interactuar con los elementos de juego) como intrínseca con las actividades académicas como tal. Dentro de los campos de educación en los que ha sido utilizado se encuentra el área de informática y específicamente en el área de enseñanza de desarrollo de software. En el capítulo siguiente se presentan en los artículos relacionados algunos ejemplos de cómo ha sido aplicada en el contexto educativo de desarrollo de software, en donde se identifica al modelo de las 6D como uno de los que ha sido utilizado en diversos trabajos y que permite ser adaptado al contexto educativo conservando la esencia del mismo.

## II. TRABAJOS RELACIONADOS

La gamificación ha sido implementada para procesos de enseñanza y aprendizaje en el área de desarrollo de software. Sin embargo, se han encontrado diversos casos de aplicación donde no siguen un protocolo específico [1]–[5] sino que utilizan algunos de los elementos de gamificación para ludificar su proceso de enseñanza. Pero al no tener un diseño adecuado tiene una probabilidad de un 80% de fallar en alcanzar los objetivos de la actividad [6] o ser afectados por la denominada “puntificación” que se genera cuando se aplican puntos o elementos de la gamificación para hacer más divertida la actividad, pero sin un objetivo definido.

Otros trabajos han desarrollado un protocolo o modelo de aplicación propio [7]–[9] que les permite replicarlo en otras

actividades y tener mayor probabilidad al momento de alcanzar los objetivos propuesto al gamificar la actividad.

Asimismo, existen otros que han aplicado el modelo definido por Werbach y Hunter conocido como el modelo de las 6D [10]. Este modelo consiste en seguir seis pasos específicos que inician desde la definición de los objetivos del negocio hasta el despliegue de las herramientas propias de la gamificación, incluyendo un paso muy importante denominado “No olvidar la diversión”, debido a que al momento de gamificar se está emulando al juego, el cual debe tener un componente de diversión para ser considerado como tal.

Dentro de los casos que aplican el modelo de las 6D se encuentra el de Uyaguari [11] que aplicaron el modelo paso a paso para un curso de riesgo en ingeniería de software. Los autores concluyen la importancia del modelo de diseño utilizado para este tipo de aplicaciones, ya que permite identificar patrones, métricas y elementos comunes que ayuden a cumplir con el desarrollo de las fases propuestas por la actividad planteada.

Otro caso de estudio fue desarrollado en un grupo de estudio de ingeniería de software que fue basado por completo en el modelo de las 6D [12]. La validación fue realizada tomando como base a grupos de control en donde el rendimiento general fue superior así como el nivel de participación en clase comparado con el grupo que no fue influenciado por la gamificación.

En un curso de diseño de patrones la aplicación del modelo 6D fue enfocado solamente en el despliegue de las herramientas que está basado en dinámicas, mecánicas y componentes los cuales por si solo pueden llevar la actividad de aprendizaje a un entorno de juego a través de la narrativa y progreso que se genera desde las dinámicas, los retos, elementos de competición o colaboración de las mecánicas y de los componentes que hacen parte más del aspecto estético y de interacción del jugador con la actividad. En este caso los resultados obtenidos fueron muy favorables [13].

Finalmente, Cabezas [14], analiza el modelo de Werbach y lo compara con los procesos de acreditación de ABET, proponiendo un nuevo modelo que consiste en enfocarse más en las metas educativas, las características de los estudiantes y agrega un último paso denominado el análisis de datos y la

retroalimentación que permite mejorar el diseño de la actividad gamificada para futuras aplicaciones.

### III. MODELO PROPUESTO

Dado que el modelo de Werbach fue aplicado inicialmente a la industria y con un enfoque hacia el mercadeo que permitiera atraer clientes y sostenerlos a través del tiempo. Se considera que, aunque su aplicación a educación es posible, es importante realizar algunas variaciones dentro de los pasos descritos por el modelo original para que los objetivos académicos se puedan cumplir.

A continuación, se describen cada uno de los pasos originales con su correspondiente propuesta de adaptación para la enseñanza de desarrollo de software.

#### A. *Definición de los objetivos del negocio – Definición de las competencias académicas a desarrollar*

No se refiere a los objetivos generales de la organización, sino a los objetivos especificación del sistema gamificado, que puede ser incrementar el número de clientes, la retención de estos o incrementar el nivel de productividad del empleado.

En su adaptación a la academia los objetivos del negocio se convierten en el objetivo de la actividad pedagógica a desarrollar. Que además debe incluir el fin de la gamificación el cual puede ser aumentar el rendimiento académico, mejorar el nivel de interacción de los estudiantes, aumentar la motivación o el compromiso del estudiante con la actividad, entre otros.

En el caso de desarrollo de software está enmarcado en el tipo de curso o actividad específica que puede ser según la guía de currículo de ingeniería de software de la ACM y IEEE las áreas de diseño de software, verificación y validación, calidad del software, análisis de requisitos y desarrollo de software o programación [15]. La recomendación es que la actividad esté descrita como una competencia académica que se redactan describiendo un dominio cognitivo en forma de actividad y posteriormente va acompañado del área específica [16]. Un ejemplo sería el estudiante es capaz de interpretar el algoritmo dado en lenguaje de programación C++. Que indica que el objetivo es que el estudiante esté en capacidad de hacer una interpretación y para eso se especifica si la gamificación aportará a través del comportamiento esperado del estudiante.

#### B. *Delinear los comportamientos objetivos / Definir los comportamientos esperados de los estudiantes*

Después de definir el por qué se aplicará la gamificación, se debe enfocar en establecer que se quiere que cada jugador realice y como se puede medir esto, vinculando comportamientos y métricas.

En el caso académico los jugadores son estudiantes y se debe definir a través de la actividad gamificada cuáles son los comportamientos que serán afectados por los elementos de juego. Por ejemplo, para mejorar el compromiso del equipo, el comportamiento objetivo es aumentar la participación en el desarrollo de cada uno de los miembros del equipo, cada uno desde su rol. Para el equipo de requisitos, es aumentar el número de requisitos validados para realizar, luego para el

equipo de desarrollo se vincula con generar código más eficiente en menor tiempo, finalmente, para el equipo de prueba puede ser brindar un mayor número de retroalimentaciones, o generar un mayor número de interacciones con el resto del equipo de trabajo.

Ese comportamiento objetivo debe ser medible y evaluado para verificar el cumplimiento del objetivo definido en el paso anterior. La medición se hace a partir del uso de algún instrumento de validación, entre los que se encuentran la analítica de datos del comportamiento de los estudiantes, de evaluaciones cualitativas a través de encuestas, entrevistas, grupos focales u observación, de evaluación cuantitativa a través de la aplicación de evaluaciones, tareas y otras actividades que se usan normalmente en el contexto académico.

#### C. *Describir a los jugadores / Caracterizar los estudiantes*

En este punto se deben describir a los participantes que alcanzarán los objetivos previamente descritos. Indicar quienes son, que tipo de relación tienen con el proveedor del juego, es decir, si son sus clientes, empleados, estudiantes, entre otros. Con esto es importante definir, que puede motivar a los jugadores, tanto desde el punto de vista intrínseco como extrínseco. Igualmente, pensar que factores también pueden generar desmotivación en los participantes, no solamente al momento de jugar, sino en términos generales, que hace que la persona no quiera cumplir con su tarea. Otra posibilidad es hacer segmentación de los tipos de participantes, agrupándolos en subgrupos, buscando generar una estrategia gamificada que aplique a varios de los subgrupos.

Para hacer la descripción en el contexto académico se pueden utilizar los datos demográficos, los conocimientos previos, los estilos de aprendizaje y las herramientas a las cuales tiene acceso en caso de que la actividad sea mediada por herramientas tecnológicas. En un curso de desarrollo de software es muy importante establecer el nivel de conocimiento previo que tiene el estudiante, así como las diferentes herramientas que sabe utilizar.

#### D. *Idear (Devise) los ciclos de actividad*

Los juegos tienen un inicio y generalmente tienen un final, durante ese proceso existen una serie de ciclos que hace que el juego lleve una secuencia y que no se vuelva monótono. En el modelo se proponen dos tipos de ciclos, el ciclo de motivación y el de progresión.

El primero brinda una perspectiva micro, describiendo que deben hacer los jugadores, por qué deben hacerlo y el sistema que hace en respuesta de ello. Por ejemplo, después de cumplir una actividad se generan puntos de recompensa o una medalla, que es la retroalimentación positiva que motiva al usuario a realizar acciones futuras. Siendo el elemento fundamental la retroalimentación permanente, debido a que genera respuestas visibles. La motivación hace que el participante ejecute una acción, la cual genera una retroalimentación instantánea que busca generar mayor motivación en el jugador para que ejecute una nueva acción. El problema se genera, cuando este ciclo se vuelve repetitivo, debido a que la motivación del primer día no

será igual después de un tiempo. Es en este punto donde es necesario el ciclo de progresión.

Este segundo ciclo, genera una perspectiva macro durante el recorrido del jugador a través del mundo gamificado. En los juegos tradicionales, existen niveles, desde uno básico que se supera fácilmente, pero permite aprender la dinámica del juego, hasta alcanzar un nivel avanzado que indica que el jugador tiene un dominio del juego, es decir, alcanza un nivel de maestría sobre el mismo. Generalmente cada nivel, tiene una progresión básica y un enemigo final para poderlo culminar. En el caso de la gamificación ese enemigo final se considera un reto mayor para el jugador que le generará una retroalimentación mayor a un reto tradicional. Los progresos parecen lineales, pero no necesariamente lo son, sin embargo, siempre se debe mantener una posibilidad de mejorar en el juego para poder conservar la motivación de los jugadores.

En un contexto académico el segundo ciclo está relacionado con la taxonomía modificada de Bloom [17] la cual actúa como un ciclo que va desde un nivel inferior hasta uno superior (ver tabla 1). Cada nivel representa un dominio cognitivo que va aumentando su complejidad y que además requiere haber superado un nivel anterior para llegar al siguiente. Similar a como ocurre en un juego. Por ejemplo, para un jugador llegar a tener la capacidad de revisar (que pertenece al quinto dominio) ya debe tener la capacidad de recordar, comprender, aplicar y analizar. Es por esto que organizando una actividad académica siguiendo la taxonomía, se pueden realizar ciclos de progreso asociados al cumplimiento de cada dominio.

TABLA 1 TAXONOMÍA DE BLOOM MODIFICADA POR KRATHWOHL [17]

Nivel	Dominio cognitivo	Procesos asociados
1	Recordar	Reconocer, Recordar, Listar, Describir, Recuperar, Identificar, Localizar, Emparejar
2	Comprender	Interpretar, Ejemplificar, Clasificar, Comparar, Resumir, Defender, Explicar, Parafrasear
3	Aplicar	Ejecutar, Implementar, Resolver, Usar, Modificar, Actualizar
4	Analizar	Diferenciar, Organizar, Atribuir, Comparar, Relacionar, Deconstruir, Estructurar, Integrar
5	Evaluar	Comprobar, Criticar, Revisar, Formular, Realizar hipótesis, Experimentar, Juzgar, Argumentar
6	Crear	Generar, Planear, Diseñar, Construir, Idear, Trazar, Inventar, Mezclar

En el caso de la aplicación de ingeniería de Software, se pueden utilizar ciclos de motivación a través de elementos como la puntuación por el número de participaciones activas y la importancia de las mismas en el proceso de desarrollo, bien sea para los requisitos, el diseño, la implementación o las pruebas. Otro elemento muy utilizado es la tabla de líderes que muestra socialmente como ha sido el aporte de cada uno. En el caso del desarrollo, se puede aplicar retroalimentación por parte de los desarrolladores expertos o los mismos clientes, que se ven reflejado en puntuación para el desarrollador novato. Las progresiones se reflejan en la etapa del ciclo o en la complejidad del software que se está desarrollando, que va desde un nivel básico hasta un aporte más complejo para el equipo de trabajo.

### E. No (Don't) olvidar la diversión

Uno de los motivos por los que fracasa la aplicación de la gamificación, es que se enfoca en el uso de elementos sin estructurar realmente el juego. Esto es importante porque el juego trae intrínseco la diversión, si no es divertido no se puede considerar como la aplicación de un juego en un contexto serio. La aplicación de la gamificación debe generar que el participante se motive a jugar, inclusive sin la obligación extrínseca de hacer parte del equipo de trabajo en el proceso de desarrollo de software, sino que por propia voluntad quiere mejorar sus procesos a través del uso de juegos.

### F. Despliegue de las herramientas apropiadas

Finalmente, la última etapa consiste en la selección adecuada de mecánicas y componentes que integran el sistema gamificado que permita alcanzar los objetivos de cada participante y finalmente los del curso o la actividad académica. Para poder conseguirlo se recomienda una persona experta en el objetivo del negocio, un entendimiento del grupo de jugadores y su psicología para genera motivación, diseñadores de juegos, experto en analíticas y datos que permitan utilizar los datos que se van generando a partir del juego, y la tecnología requerida para poder aplicar el modelo de gamificación y empezarlo a implementar.

Las herramientas se dividen en tres aspectos principales, dinámicas, mecánicas y componentes. Las dinámicas son el mayor nivel de abstracción del juego, siendo un contexto grande de los aspectos a ser gamificados. Se compone de elementos como las limitaciones que tendrá el juego, las emociones a generar entre los participantes como la competitividad, curiosidad, frustración o felicidad. Además, tiene la narrativa que es el guion de la historia del juego que se va a desarrollar, las progresiones, tanto a nivel de niveles como el crecimiento y desarrollo del jugador, y por último las relaciones que tendrán los jugadores, que generan trabajo colaborativo, estatus social o altruismo.

Las mecánicas son el proceso básico que llevan a la acción y generan un compromiso y motivación en los jugadores. Se identifican varias mecánicas principales que se listan a continuación en la tabla II.

TABLA 2 MECÁNICAS DE JUEGO [10]

Mecánica	Descripción
Retos	Tareas que requieren esfuerzo para resolverse
Aleatoriedad	Elementos aleatorios que afectan el juego
Competición	Un individuo o equipo compite contra otros para establecer un ganador
Cooperación	Los jugadores deben trabajar en conjunto para alcanzar un objetivo conjunto
Retroalimentación	Información sobre el rendimiento de algún jugador
Recompensas	Beneficios por completar una acción o un reto
Transacciones	Comercio o intercambios entre jugadores directamente o por intermediarios
Turnos	Participación secuencial que se alterna con otros participantes
Estado de victoria	Objetivos que hacen que un jugador o un grupo gane, empate o pierda.

Por último, se encuentran los componentes que son elementos más específicos, generalmente visuales que permiten la interacción como tal de los participantes con el juego. Estos se dividen principalmente en los componentes descritos a continuación en la tabla III.

TABLA 3 COMPONENTES DE JUEGO [10]

Componentes	Descripción
Logros	Objetivos definidos
Avatares	Representación visual de los jugadores
Medallas	Representación visual de los logros obtenidos
Enemigos finales (jefes)	Retos difíciles que permiten culminar un nivel
Colecciones	Conjunto de objetos o medallas para acumular
Regalos	Oportunidad de compartir recursos con otros
Tabla de líderes	Mensaje visual que muestra la progresión de jugadores y logros obtenidos
Niveles	Pasos definidos para medir la progresión de un jugador
Misiones	Retos predefinidos con objetivos y recompensas
Gráficas sociales	Representación visual de redes sociales en el juego
Equipos	Grupos de jugadores que trabajan juntos por un bien común

En una aplicación de gamificación, no es necesario utilizar todas las mecánicas y componentes, solamente elegir cuales pueden ser utilizadas según la dinámica requerida.

Un caso propuesto, donde se ejemplifica la aplicación de las dinámicas, mecánicas y componentes en el proceso de Ingeniería de Software es para el entrenamiento de desarrolladores nuevos en el equipo de trabajo. El objetivo general de la organización es obtener un equipo de trabajo con desarrolladores adecuados a las necesidades de la organización, asimismo la meta de cada jugador es mejorar sus destrezas de desarrollo. Para hacerlo se crea una dinámica competitiva, en donde cada jugador debe superar unos niveles, que van desde uno muy básico hasta uno donde ya se identifica que tiene la capacidad suficiente para integrar el equipo de desarrollo. En cada nivel deben superar unos retos que tendrán una retroalimentación positiva a través de puntos y medallas según su rendimiento, siendo visibles en una tabla de líderes, que genera reconocimiento social. Pero, además, les permitirá identificar los errores cometidos para resolver dichos retos.

#### IV. CONCLUSIONES

La gamificación ha sido aplicada no solo en un contexto industrial y comercial sino también en uno académico, permeando diferentes áreas del conocimiento como lo es el desarrollo de software. Dentro de este contexto se han identificado diferentes formas de aplicarse, teniendo mejores resultados los que han aplicado un modelo o un protocolo definido paso a paso que permita aplicar la gamificación como un diseño completo que aporte al curso y no solamente como el uso de ciertos elementos de juego.

El modelo de las 6D permite diseñar la actividad gamificada desde los objetivos propios del curso, tomando en

cuenta elementos importantes como los comportamientos esperados en los alumnos según la caracterización de los mismos, indicando los ciclos de motivación y el despliegue de herramientas que permiten alcanzar los objetivos esperados de la actividad.

#### REFERENCES

- [1] W. Q. Qu, Y. F. Zhao, M. Wang, and B. Q. Liu, "Research on teaching gamification of software engineering," Proc. 9th Int. Conf. Comput. Sci. Educ. ICCSE 2014, no. Iccse, pp. 855–860, 2014.
- [2] R. F. Maia and F. R. Graeml, "Playing and learning with gamification: an in-class concurrent and distributed programming activity," Front. Educ. Conf., 2015.
- [3] J. Beltran, H. Sanchez, and M. Rico, "Incrementar la motivación en el aprendizaje de Fundamentos de Programación Java usando Moodle Gamificado Increase motivation in learning Java Programming," Inf. Syst. Technol. (CISTI), 2016 11th Iber. Conf., 2016.
- [4] S. Butler and D. T. Ahmed, "Gamification to Engage and Motivate Students to Achieve Computer Science Learning Goals," 2016 Int. Conf. Comput. Sci. Comput. Intell. Gamification, pp. 237–240, 2016.
- [5] U. Sch, "Training Scrum with Gamification Lessons learned after two teaching periods," Glob. Eng. Educ. Conf., no. April, pp. 754–761, 2017.
- [6] O. Pedreira, F. García, N. Brisaboa, and M. Piattini, "Gamification in software engineering - A systematic mapping," Inf. Softw. Technol., vol. 57, no. 1, pp. 157–168, 2015.
- [7] C. Su, "The effects of students' motivation, cognitive load and learning anxiety in gamification software engineering education: a structural equation modeling study," Multimed. Tools Appl., pp. 10013–10036, 2016.
- [8] M. Piteira and C. J. Costa, "Gamification: Conceptual Framework to Online Courses of Learning Computer Programming," Inf. Syst. Technol. (CISTI), 2017 12th Iber. Conf., 2017.
- [9] M. C. Gómez Alvarez, G. P. Gasca-hurtado, J. A. Calvo-Manzano, and T. San Feliu Guilabert, "Diseño de un instrumento pedagógico para la enseñanza de la mejora de procesos software," in 9th Iberian Conference on Information Systems and Technologies (CISTI), 2014, pp. 1–7.
- [10] K. Werbach and D. Hunter, For the Win. Philadelphia: Wharton Digital Press, 2012.
- [11] F. Uyaguari, M. Intriago, and E. Salazar Jacome, "Gamification Proposal for a Software Engineering Risk Management Course," in New Contributions in Information Systems and Technologies, Advances in Intelligent Systems and Computing, vol. 353, 2015, pp. III–IV.
- [12] P. Gomes Fernandes and C. Correa da Silva, "Game elements in a software engineering study group: a case study," 2017 IEEE/ACM 39th Int. Conf. Softw. Eng. Softw. Eng. Educ. Train. Track, pp. 160–169, 2017.
- [13] A. Bartel and G. Hagel, "Gamifying the Learning of Design Patterns in Software Engineering Education," Glob. Eng. Educ. Conf., no. 01, pp. 74–79, 2016.
- [14] I. Cabezas, "On combining gamification theory and ABET criteria for teaching and learning engineering," Front. Educ. Conf. (FIE), 2015. 32614 2015. IEEE, pp. 1–9, 2015.
- [15] A. for C. M. ACM and C. S. IEEE, "Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering," Comput. Curricula Ser., no. February, pp. 664–75, 2014.
- [16] S. Gómez Jaramillo and J. Moreno Cadavid, "Selection of collaborative learning techniques using Bloom's taxonomy," Commun. Comput. Inf. Sci., no. 606, pp. 1–12, 2016.
- [17] D. R. Krathwohl, "A Revision of Bloom's Taxonomy: An Overview," vol. 41, no. 4, 2002.