

**Experimentación y modelación computacional para la construcción de videojuegos:
Actividades interdisciplinarias de bajo umbral y alto techo**

**Experimentation and computational modeling for the construction of videogames:
Interdisciplinary activities of low threshold and high ceiling**

PRETELÍN-RICÁRDEZ, Angel†*

ID 1^{er} Autor: Ángel Pretelín-Ricárdez / ORC ID: 0000-0003-0440-6094, Researcher ID Thomson: K-5747-2015, CVU CONACYT ID: 48627

Instituto Politécnico Nacional, UPIITA, México

A. Pretelín

apretelin@ipn.mx

A. López, C. Lima, J. Reyes (Dir.) Educación para todos. Tópicos Selectos de Educación en CITeM. ©ECORFAN-México, 2018.

Resumen

En este capítulo, se explora cómo estudiantes universitarios mexicanos (Licenciatura en Matemáticas Aplicadas) pusieron en práctica y desarrollaron sus conocimientos matemáticos, físicos y de ingeniería de manera interdisciplinaria, a través del diseño y programación de videojuegos, y de esa manera adquirir experiencia con respecto a la modelación computacional de sistemas ingenieriles.

La idea de que los estudiantes fueran diseñadores y programadores de videojuegos, se fundamentó en el paradigma del construccionismo (Papert & Harel, 1991), en el *Game Design Project* (Kafai, 1994) y también utilizando Actividades de Bajo Umbral y Alto Techo (ABUAT). Las actividades fueron diseñadas tomando en consideración los seis principios de la metodología de Actividades Inductoras de Modelos (*Model-Eliciting Activities / MEAs*): Realidad, construcción del modelo, documentación del modelo, autoevaluación, generalización del modelo, prototipo simple (Hamilton et al., 2008).

Durante la implementación de las actividades, se analizaron los siguientes aspectos:

(1) Cómo llevaron a cabo los estudiantes el proceso de modelación computacional; (2) cómo realizaron los estudiantes la experimentación y la modelación computacional para la construcción de los videojuegos; y (3) cómo usaron la matemática y cómo la integraron con la física y la programación durante las actividades.

Las conclusiones muestran las aportaciones que tiene este trabajo como una metodología construccionista en el nivel superior, a través de un enfoque interdisciplinario de experimentación, modelación y construcción de videojuegos.

Construccionismo, Interdisciplinario e integrado, Modelación computacional, Actividades inductoras de modelos, Videojuegos

Abstract

In this chapter, I explore how Mexican university students (Bachelor of Applied Mathematics) used and developed their knowledge of mathematics, physics and engineering, in an interdisciplinary way, through the design and programming of videogames, and in that way gain experience with respect to the mathematical and computational modeling of engineering systems.

The idea of students as designers and programmers of videogames was based on the constructionism paradigm (Papert & Harel, 1991), Game Design Project (Kafai, 1994) and also using Low Threshold High Ceiling Activities (LTHCA). The activities were designed considering the six principles of the Model Eliciting Activities (MEA) methodology: reality, model construction, model documentation, self-evaluation, model generalization, simple prototype (Hamilton et al., 2008).

During the implementation of the activities, I analyzed the following aspects:

(1) How the students carried out the computational modeling process; (2) how the students carried out their processes of experimentation and of computational modeling for the construction of the videogames; and (3) how they integrated the mathematics with the physics and programming during activities.

The conclusions show the contributions that my work has as a constructionist methodology for higher education, that used an interdisciplinary approach of experimentation, modeling and construction of videogames.

Constructionism, Interdisciplinary and integrated, Computational modeling, Model inducing activities, Video games

1 Introducción

El paradigma del construccionismo (Papert & Harel, 1991) plantea que el aprendizaje se facilita a través de actividades de construcción de objetos externos y compartibles (i.e. objetos en el mundo – no sólo objetos físicos, también puede ser algo como un programa de cómputo, un poema, una teoría, etc.). El construccionismo va más allá de manipular objetos para aprender: se enfoca en crearlos, recrearlos, construirlos y reconstruirlos, y plantea que es a través de esos procesos de construcción y creación (cuando se tiene como meta desarrollar productos significativos y compartibles), que los individuos aprenden.

En este capítulo se presentan algunos resultados cualitativos, producto de implementar una metodología (*experimentación - modelación - construcción de videojuegos*) basada en el paradigma del construccionismo con estudiantes universitarios de la Licenciatura en Matemáticas Aplicadas de la Universidad Juárez del Estado de Durango (UJED), en México. Dicha metodología establece tres actividades: (i) Actividad de experimentación; (ii) Actividad de simulación (modelación computacional); y (iii) Actividad de construcción de un producto interdisciplinario (en este caso un videojuego).

Los objetivos que se persiguen al implementar esta metodología construccionista son que:

- Los estudiantes construyan un videojuego cuya mecánica de juego o *puzzles* esté basada en los modelos computacionales de ciertos sistemas físicos propuestos en forma de retos.
- Los estudiantes puedan “construir puentes” para complementar y relacionar diferentes disciplinas como la matemática, la física y la programación, a través de la creación de videojuegos.

También, cabe destacar que una de las características importantes de la metodología descrita aquí, fue darle a las actividades el carácter de “bajo umbral”, pero de “alto techo”, es decir, que las actividades pudieran ser realizadas por cualquiera de los estudiantes, no importando el semestre que cursaran, pero que a la vez les permitiera a todos acceder a ideas poderosas (Papert & Resnick, 1996) relacionadas con el uso de la tecnología como medio para “construir puentes” entre las disciplinas (conceptos implícitos) que estaban poniendo en práctica. Para cumplir lo anterior, se diseñaron las actividades tomando en consideración la teoría de Actividades Inductoras de Modelos (*Model-Eliciting Activities* o *MEAs*) propuesta por Lesh y Doerr (2003).

Para describir el diseño, el desarrollo, los resultados y las conclusiones de la propuesta metodológica, el capítulo está organizado en cinco secciones. Primero, se presenta el marco conceptual donde se sitúa la propuesta a partir de cuatro temas: construccionismo (Papert & Harel, 1991), modelación computacional, Actividades Inductoras de Modelos de Lesh y Doerr (2003) y el aprendizaje basado en videojuegos, donde se hace referencia al trabajo de Kafai (1994).

Después, se especifican las características de la metodología que se siguió en esta propuesta, se describen los participantes, las herramientas tecnológicas utilizadas, la manera en que se llevó a cabo el diseño y la implementación de las actividades, para finalizar con la recolección de datos.

Enseguida, se presentan y discuten los resultados para cada una de las actividades realizadas por los estudiantes. Finalmente, se exponen las principales conclusiones que se obtuvieron del estudio y algunas reflexiones finales que se desprenden de los resultados obtenidos.

2 Situando la propuesta a partir del marco conceptual

2.1 El construccionismo y su implementación

Las teorías del constructivismo (derivadas de las propuestas de Piaget, 1968) consideran que la construcción del conocimiento no es un proceso simple y aislado donde todos los sujetos perciben y procesan la realidad de la misma forma; más bien, cada sujeto necesita construir y reconstruir su conocimiento de acuerdo a sus experiencias previas.

Estas teorías son la base del paradigma educativo conocido como “construccionismo” (Papert & Harel, 1991) el cual plantea que a partir de actividades de construcción externa (concretas) se puede favorecer el aprendizaje interno (abstracto) de los estudiantes.

Partiendo de lo anterior, se propuso una secuencia de actividades orientada a que los estudiantes resolvieran retos de modelación computacional (simulación) a través de la experimentación y construcción dentro de un motor de videojuegos o *game engine* (Unity 2D). En este sentido, se considera muy importante la idea expuesta por Ackerman (2001):

El enfoque de Papert [el construccionismo] nos ayuda a entender cómo las ideas se forman [construyen] y transforman [reconstruyen] cuando son expresadas a través de diferentes medios, cuando son actualizadas en determinados contextos, cuando se llevan a cabo por mentes individuales. (p. 441).

De esta manera, se elaboró un conjunto de actividades, donde se quería analizar e intentar entender la manera en que se construyen y reconstruyen las ideas (pero sobre todo las experiencias) de los estudiantes respecto a la modelación computacional de sistemas físicos, a través del uso de ciertos medios, en donde se esperaba que pusieran sus ideas en acción. Los medios a través de los cuales los estudiantes expresaron esas formaciones y transformaciones de ideas fueron: la conceptualización del modelo del sistema a través de bosquejos o diagramas (lápiz y papel), la simulación del modelo que se conceptualizó (programación) y la implementación de la simulación del modelo en las mecánicas de juego de un videojuego (programación).

Tomando en consideración lo expuesto arriba, para hablar de la forma en la que se implementa el construccionismo, se tiene que hablar de sus fundamentos en las construcciones computacionales.

Para Papert, las computadoras tenían un papel central en el construccionismo, como un medio para ofrecer a los estudiantes mejores oportunidades para construir su aprendizaje. En su libro *Desafío de la mente*, Papert (1981) explica que se busca:

convertir a las computadoras en instrumentos lo suficientemente flexibles para que muchos niños logren crear, cada uno para sí mismo, algo parecido a lo que los engranajes fueron para mí. (p. 13)

Por tanto, el construccionismo, desde sus orígenes en los años 60s, plantea el potencial de las computadoras para que los sujetos, por sí mismos, a través de la programación, puedan crear y construir cosas que sean significativas para ellos.

Otros conceptos importantes dentro del paradigma del construccionismo que son necesarios para comprender a lo que se refería Papert cuando hablaba de “mejores oportunidades para aprender”, y que sirven para situar mejor la metodología que se está proponiendo (ver la sección 3 de este capítulo) son: objetos para pensar (*Objects-to-think-with*) y entidades públicas (*public entities*).

2.1.1 Los objetos para pensar

Papert (1981) hace un recuento de la gran influencia que tuvieron en su infancia los engranes y explica cómo estos “objetos” funcionaron para que él pudiera pensar y construir sobre otras cosas (proporciones matemáticas en su caso) situadas en distintos contextos. Estos engranes son un ejemplo concreto de lo que es un objeto para pensar dentro del paradigma construccionista. Un objeto para pensar es un medio que puede ser utilizado para experimentar con él, reconstruirlo, observar su funcionamiento y compartirlo; y, de esta forma, inducir a pensar y construir sobre otras cosas. “Los objetos para pensar [...] pasan a ser una parte inherente de la construcción de conocimiento” (Badilla & Chacón, 2004, p. 8) y de experiencias.

Durante la implementación de la metodología construccionista que se describe en este capítulo, los estudiantes trabajaron con varios objetos para pensar: computadoras, motores de videojuegos (Unity 2D), simulaciones, modelos y videojuegos.

2.1.2 Las entidades públicas

Una entidad pública es una construcción que permite representar, de manera visual o auditiva, ideas y conceptos para experimentar con ellos (Badilla & Chacón, 2004). En términos generales, es posible decir que es todo lo que se puede mostrar o compartir para aprender: un proceso, un producto, etc. Papert y Harel (1991) dan como ejemplos de entidades públicas a castillos de arena en la playa o teorías del universo.

En lo que respecta a la propuesta presentada aquí, los diagramas, esquemas, bosquejos, las simulaciones y videojuegos que construyen los estudiantes, así como la socialización del proceso que ellos siguen para construirlos, se pueden considerar como entidades públicas, ya que son cosas que se discuten y comparten con otros y, de esta manera, pueden reforzar el aprendizaje y las experiencias.

2.2 Modelación computacional (exploratoria y expresiva)

Para situar el tema de modelación computacional con el de la metodología constructivista que se aborda en este capítulo, son necesarias algunas definiciones que se utilizan en el diseño y desarrollo de las actividades: sistema, modelo y simulación. Se hace la definición de estos tres conceptos, porque comúnmente un licenciado en matemáticas aplicadas debería de saber construir y validar modelos matemáticos de sistemas (físicos) auxiliándose de la construcción de simulaciones computacionales. Se partirá entonces, de la definición del concepto de sistema dada por Hestenes (2010):

Un *sistema* es un conjunto de *objetos* relacionados, que pueden ser reales o imaginarios, físicos o mentales, simples o compuestos. La *estructura* de un sistema es un conjunto de relaciones entre sus objetos. El sistema en sí se llama el *referente* del modelo (p. 17).

La definición anterior establece en su última oración, la relación que guarda un sistema respecto a un modelo, el cual el mismo Hestenes (2010) define, de manera breve pero concisa, como “una representación de la estructura en un sistema dado” (p. 17).

Situándose un poco más en el ámbito de la matemática educativa, una definición más extensa de modelo la ofrecen Lesh y Harel (2003):

Los modelos son sistemas conceptuales que por lo general tienden a ser expresados usando una variedad de medios de interacción y representación, que pueden incluir símbolos escritos, lenguaje hablado, gráficos por computadora, diagramas o gráficas en papel, o metáforas basadas en experiencia. Sus propósitos son construir, describir o explicar otro(s) sistema(s).

Los modelos incluyen tanto: (a) un sistema conceptual para describir o explicar los objetos matemáticos pertinentes, relaciones, acciones, patrones y regularidades que son atribuidos a la situación de resolución de problemas; y (b) los procedimientos de acompañamiento para generar construcciones útiles, manipulaciones o predicciones para el logro de objetivos claramente reconocidos. Los modelos matemáticos son distintos de otras categorías de modelos principalmente porque se centran en las características estructurales (en lugar de, por ejemplo, características físicas, biológicas, o artísticas) de los sistemas que describen. (p. 159)

Ahora bien, en el ámbito de la matemática aplicada, cuando un estudiante de esta disciplina construye un modelo, éste debe utilizar como medios de interacción y representación una gran variedad de entornos digitales de simulación. Así, se enuncia la siguiente definición de Dodge (2008) respecto a la simulación:

La simulación es un método para analizar, diseñar y operar sistemas complejos. [...] implica el diseño de un modelo de un sistema y la realización de experimentos sobre él a medida que se avanza (p. 498).

En esta definición se observa que, simular tiene que ver con la realización de experimentos que nos permitan analizar, diseñar y operar sistemas. Esta experimentación se realiza para poder llegar a la mejor simulación (o representación) de un modelo.

Cabe recalcar que durante la implementación de la metodología, los estudiantes no solo experimentaron la manipulación de variables o parámetros en una simulación computacional, sino que la idea principal fue que la construyeran y reconstruyeran para que, posteriormente, experimentaran insertando valores iniciales a ciertas variables físicas virtuales (i. e. gravedad, fuerza, fricción, tipo de material, elasticidad/“rebotabilidad”) que permitan modificar los parámetros de un modelo simulado de forma computacional.

Siguiendo con lo planteado en el párrafo anterior, la simulación computacional de un modelo es definida por Ifenthaler (2012) como “un programa informático o algoritmo que simula cambios de un sistema modelado en respuesta a señales de entrada” (p. 710). Esta definición se acerca más a lo que los estudiantes estuvieron realizando en las actividades que se les plantearon.

Sin embargo, como se mencionó antes, los estudiantes no solo manipularon los parámetros de un modelo a través de una simulación (o software de simulación), sino que construyeron (programaron de manera implícita, pues no escribieron ninguna línea de código) dicha simulación; es decir, que ellos modelaron computacionalmente los sistemas para después poderlos implementar dentro de un videojuego diseñado y programado por ellos.

La diferencia entre simulación computacional y modelación computacional, es abordada por Araujo, Veit, y Moreira (2007), quienes afirman que “estos dos tipos de actividades se distinguen por el acceso que el alumno tiene al modelo matemático [...] subyacente a la implementación de la actividad” (p. 504). Una descripción más extensa de estas diferencias se puede encontrar en López, Veit, y Araujo (2011):

En una simulación computacional que representa un modelo físico, el alumno puede insertar valores iniciales para variables, alterar parámetros y, de forma limitada, modificar las relaciones entre las variables; pero no tiene autonomía para modificar la estructura de la simulación (modelo matemático o icónico pre-especificado); o sea, acceso a los elementos más básicos que la constituyen.

La interacción del estudiante con la simulación tiene un carácter eminentemente exploratorio; mientras que, en la modelación computacional el estudiante tiene acceso a los primitivos que constituyen el modelo computacional, pudiendo construirlos desde el principio y reconstruirlos conforme desee. (p. 205)

Después de haber mostrado algunas características que permiten diferenciar la simulación computacional con respecto de la modelación computacional, se puede decir que la primera presenta un carácter exploratorio de la modelación, mientras que la segunda un carácter expresivo. Lo del carácter exploratorio y expresivo de la modelación (o los modelos), se puede describir citando el siguiente fragmento de Doerr (1995), que retoma algunos conceptos desarrollados por Bliss y colegas (Bliss & Ogborn, 1989; Bliss et al., 1992):

Modelos exploratorios [o modelación exploratoria] son aquellos modelos que son construidos por expertos para representar saberes o conocimientos en algún dominio de contenido [...]. Los estudiantes típicamente exploran las consecuencias de sus acciones dentro de los límites de estos modelos de dominio de contenido. Estos modelos son en esencia micromundos que proporcionan al estudiante un conjunto de mundos simulados e idealizados que encarnan, por ejemplo, las leyes newtonianas del movimiento, permitiendo al estudiante explorar las consecuencias de los cambios en los parámetros de la simulación. [...]. Estos modelos exploratorios proporcionan una manera de preguntarse si los estudiantes pueden entender la manera de pensar de un experto sobre un problema.

La construcción de modelos (o modelación expresiva), por otro lado, proporciona a los estudiantes la oportunidad de expresar sus propios conceptos y aprender a través del proceso de representación de sus conceptos, definiendo relaciones y explorando las consecuencias de esas relaciones. [...]. Estos modelos expresivos [o modelación expresiva] proporcionan una manera de preguntarse si los estudiantes pueden entender su propia manera de pensar sobre un problema. Este es un cambio importante en la perspectiva de la actividad de explorar un modelo pre-construido, que necesariamente encarna los conceptos y estructuras de un experto. (Doerr, 1995, pp. 5-6) Araujo, Veit, y Moreira (2011, citando a Bliss y Ogborn, 1989) sitúan los conceptos de modelación exploratoria y expresiva en la modelación computacional, y hacen alusión a:

Dos modos básicos de usar las actividades de modelación computacional: el modo exploratorio y el modo expresivo. Las actividades exploratorias son caracterizadas por la observación, análisis e interacción del sujeto con modelos computacionales ya construidos, en el intento de permitir al alumno la percepción y la comprensión de las eventuales relaciones entre la matemática subyacente al modelo y el fenómeno físico en cuestión. En este tipo de actividad, el alumno tiene acceso a la estructura básica del modelo implementado, pudiendo modificarlo si desea. Las actividades de modelación computacional de tipo expresivo se caracterizan por el proceso de construcción del modelo desde su estructura matemática hasta el análisis de los resultados generados por él. El alumno puede interactuar totalmente con su modelo, pudiendo reconstruirlo tantas veces como le parezca necesario para la producción de resultados que le sean satisfactorios. (Araujo, Veit, & Moreira, 2011, pp. 205-206)

Después de haber expuesto estos conceptos en torno a la modelación y la modelación computacional, se puede establecer que los estudiantes que realizaron las actividades de experimentación, modelación computacional y construcción de un videojuego, realizaron un trabajo más cercano a una modelación computacional expresiva.

Este tipo de modelación es ampliamente practicada en la formación de ingenieros, y en la ingeniería en general, y debería ser parte de las actividades desarrolladas durante la formación de un estudiante de matemática aplicada, debido a que constituye un paso previo para la implementación de productos, procesos o sistemas, y dónde dichos estudiantes pueden experimentar con los modelos de sistemas sin ningún riesgo para ellos y para el sistema mismo.

2.3 Actividades inductoras de modelos

La metodología propuesta se compone de tres actividades orientadas a la construcción de videojuegos a través de la experimentación y la modelación computacional. Estas se diseñaron tomando en cuenta la teoría de Actividades Inductoras de Modelos (*Model-Eliciting Activities* o *MEAs*) propuesta por Lesh y Doerr (2003).

Son llamadas así, porque los productos que los estudiantes producen van más allá de respuestas cortas a preguntas específicas, e implican el uso de herramientas conceptuales compartibles, manipulables, modificables y reutilizables, para construir, describir, explicar, manipular, predecir o controlar sistemas matemáticamente significativos. (p.3)

En este caso, interesa particularmente los seis principios de diseño de las *MEAs*: realidad, construcción del modelo, documentación del modelo, autoevaluación, generalización del modelo, prototipo simple (Hamilton et al., 2008). Estos principios se volverán a abordar en la sección 3, en el apartado relacionado con el diseño de las actividades.

2.4 Aprendizaje basado en construcción de videojuegos

En referencia a la aplicación del construccionismo en el aprendizaje basado en la construcción de videojuegos, la presente propuesta se situará a partir de lo expuesto en (Kafai, 1994) respecto al *Game Design Project* (GDP), el cual se desarrolló siguiendo un enfoque construccionista, donde la idea principal era darle a los niños el rol de diseñadores y creadores de juegos de computadora, con la finalidad de favorecer su aprendizaje matemático y computacional. Dicho enfoque contrasta con el enfoque tradicional que se tenía del aprendizaje basado en videojuegos (*Game-Based Learning*) en la década de 1990, en donde los niños o estudiantes asumían el papel de consumidores pasivos.

El desarrollo de este proyecto de investigación significó, en su momento, romper con el paradigma de aprender a través de los juegos que otros construían y distribuían, para sustituirlo por el de aprender a través del proceso de diseño y construcción de los juegos.

En resumen, el proyecto GDP fue abordado desde un enfoque construccionista, donde la idea principal era transformar el aula en “un estudio de diseño de juegos” (Kafai, 1994), donde los estudiantes aprendieron programación, escribieron historias y diálogos (para el juego); construyeron representaciones relacionadas con números fraccionarios; diseñaron las interfaces gráficas y las estrategias de enseñanza que se utilizarían en el juego; e incluso diseñaron las “cajas comerciales” del “producto” y anuncios para el juego (como si fuera a comercializarse).

3 Componentes y procedimientos de la metodología experimentación - modelación - construcción de videojuegos

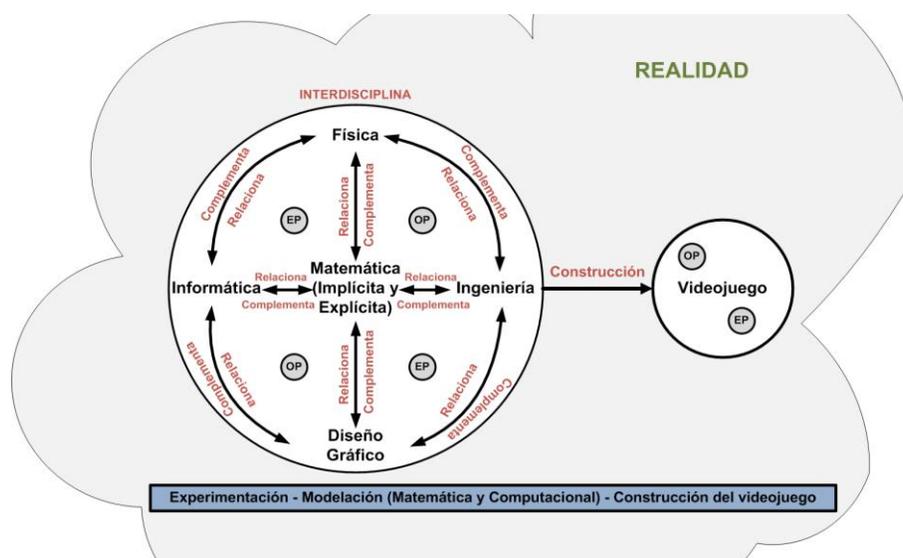
En esta ocasión, el estudio de la metodología *experimentación - modelación - construcción de videojuegos*, es producto de las actividades realizadas en el Taller Experimentación y modelación (matemática y computacional) para la construcción de videojuegos, llevado a cabo en la Facultad de Ciencias Exactas de la UJED (Durango, México) en abril de 2018. Este taller tuvo una duración de 15 horas, divididas en cinco sesiones de tres horas diarias.

3.1 La metodología experimentación - modelación - construcción de videojuegos

La idea de situar a los estudiantes en el rol de constructores o creadores de videojuegos ha sido explorada en otros proyectos de investigación (*e.g.* Kafai, 1994; Harel, 1990; Hoyles & Noss, 2005). Sin embargo (como se muestra también en la revisión hecha por Kafai & Burke, 2015), muchos de los proyectos mencionados en la literatura están enfocados en estudiantes que cursan el nivel preescolar, primaria, secundaria o preparatoria. En contraste, la metodología constructora utilizada en este taller tiene sus antecedentes en el trabajo reportado en Pretelín-Ricárdez y Sacristán (2015) y fue diseñada para ser implementada con estudiantes universitarios, aunque esto no es algo restrictivo.

La metodología establece tres actividades en donde se espera que los estudiantes lleven a cabo ciclos de construcción para resolver uno o varios problemas de modelación: (i) Actividad de experimentación; (ii) Actividad de modelación (matemática y computacional); y (iii) Actividad de construcción del videojuego (Figura 4.1), en las que se expresa la matemática de maneras diferentes y se complementa y relaciona con otras disciplinas como la física, la ingeniería, la informática y el diseño gráfico.

Figura 4.1 La metodología Experimenta-Modela-Construye videojuego



Fuente: Elaboración propia

En la Figura 4.1 se representa cómo se espera que la matemática se relacione y complemente, de manera explícita e implícita con otras disciplinas (en este caso: física, ingeniería, informática y diseño gráfico) a través de la experimentación, construcción y reconstrucción de objetos para pensar (OP) y entidades públicas (EP): modelos matemáticos y computacionales. Todo esto orientado a la creación de un producto interdisciplinario (videojuego), el cual a su vez, se convertirá en un nuevo OP y EP.

Es importante mencionar que esta metodología fue diseñada para la realización de proyectos de larga duración (de cuatro a seis meses), debido a la naturaleza interdisciplinaria de los problemas que se pretende que los estudiantes resuelvan. No obstante, la implementación en un taller corto de 15 horas resultó muy enriquecedora, gracias al ajuste en el tipo de problemáticas (retos) a resolver (ver sección 3.4).

3.2 Participantes

En el estudio participaron catorce estudiantes del programa de Licenciatura en Matemáticas Aplicadas de la UJED. De estos catorce estudiantes, cinco eran de segundo semestre, uno de cuarto semestre, cuatro eran de séptimo semestre y cuatro de octavo semestre. Ninguno de los estudiantes tenía experiencia previa utilizando el motor de videojuegos *Unity 2D* (<https://unity3d.com/es>), sin embargo, dos estudiantes reportaron haber utilizado los siguientes motores de videojuegos: *Scumm* y *Unreal engine*; además, todos los estudiantes reportaron experiencia en alguno de los siguientes lenguajes de programación: *C*, *C++*, *C#*, *Java*, *PHP*, *Pascal*, *Html*, *Latex*, *Python*, *Basic*, *JavaScript*; y en los siguientes entornos digitales: *Matlab* y *Visual Basic*; lo que permitió intuir que sí contaban con competencias en el uso de tecnología.

3.3 Las herramientas tecnológicas

Se clasificaron las herramientas tecnológicas que utilizaron los estudiantes, en dos categorías: herramientas para la construcción y herramientas de apoyo. En el primer grupo incluimos al motor de videojuegos *Unity*, es decir, el software con que se llevan a cabo las actividades de experimentación, construcción de los modelos computacionales y videojuegos; y en el segundo grupo están los procesadores de texto, editores de imágenes o algún otro software que puede auxiliar el proceso de documentación y construcción de los productos requeridos para cada actividad.

Para este caso, se utilizaron las siguientes herramientas:

- Herramientas de construcción: el motor de videojuegos *Unity* (en este caso se utilizó en el modo 2D).
- Herramientas de apoyo (recursos): un conjunto de recursos prediseñados para la integración de personajes, escenarios, música y sonidos, los cuales son de licencia libre. En el caso del motor de videojuegos *Unity*, estos recursos son conocidos como *Assets* y pueden descargarse directamente desde el motor de videojuegos, o bien, desde una búsqueda directamente por los estudiantes desde un navegador.

Con relación a la herramienta (*Unity*) utilizada para la construcción de los experimentos, las simulaciones y videojuegos, ésta cumple con los siguientes principios o características:

- Es una herramienta de propósito específico, que permite una rápida y óptima implementación de un videojuego, pero también de un modelo computacional (ver Tabla 4.1).
- Permite la construcción de modelos computacionales y videojuegos través de la programación basada en código (uso de scripts de lenguajes de propósito general: *C#* y *Java Script*), o bien, sin programar una sola línea de código (esta característica fue la que se utilizó para poder garantizar el “bajo umbral y alto techo” de las actividades). También, cuenta con su propia biblioteca de comandos y recursos (*e.g.* motor físico; editor de imágenes de fondos y *sprites* (imágenes de personajes en un videojuego); editor de música y sonido; etc.).
- El motor de videojuegos incorpora un motor físico 2D y 3D (en este caso, se utilizó sólo el motor físico en 2D: *Box2D*), para asegurar que las simulaciones físicas dentro del videojuego puedan llevarse a cabo. Esta característica es muy importante debido a que si se quisiera replicar lo expuesto en este capítulo con otro motor, se debe asegurar que el motor de videojuegos incorpore un motor físico.
- La herramienta permite experimentar a través de la construcción, y favorece que los estudiantes aprendan y ganen experiencia respecto a los modelos y la modelación computacional.

El propósito y justificación de por qué se escogieron estas herramientas, se muestran en la Tabla 4.1

Tabla 4.1 Propósito de cada herramienta tecnológica

Elemento	Propósito y justificación
Herramienta de construcción: Motor de videojuego (<i>Unity 2D</i>)	(1) Proveer a los participantes con las herramientas suficientes para implementar un modelo computacional en un videojuego. El motor de videojuegos cuenta con un motor físico que permite construir simulaciones que se rigen por las principales leyes físicas del mundo real (cuando esto sea necesario). (2) La elección del motor de videojuegos 2D se debió a que se buscaba que los estudiantes analizaran a profundidad los modelo a implementar. Se quería que tuvieran ciertas restricciones al momento de abstraer las ecuaciones o relaciones matemáticas y físicas en un ambiente de simulación 2D. También, se quería que el entorno digital fuera fácil de aprender a utilizar, respecto a otros motores de videojuegos especializados, o bien, un lenguaje de programación de propósito general. Con esto se esperaba que el estudiante se centrara en el desarrollo y la integración de la parte matemática (de manera implícita) con otras disciplinas y que no se perdiera en los detalles de aprender a utilizar o programar un entorno digital complejo.
Herramientas de apoyo: Recursos prediseñados y <i>Assets</i>	Lo que se busca con el uso de estos recursos, es que los estudiantes logren productos visualmente atractivos (simulaciones y videojuegos), pero que no tengan que consumir tiempo en la creación de los mismos.

Fuente: Elaboración propia

3.4 Diseño de las actividades

Como se ha mencionado, la metodología que se está abordando está compuesta por tres actividades: (i) Actividad de experimentación; (ii) Actividad de simulación (modelación computacional); y (iii) Actividad de construcción de un producto interdisciplinario (en este caso un videojuego).

Con base en lo anterior y en el hecho de que el taller donde se implementarían las actividades era un taller corto de quince horas, se decidió orientar todas las actividades a que los estudiantes asumieran el papel de un desarrollador de videojuegos desde el principio. De esta forma, la problemática a resolver era la de implementar “física” o sistemas físicos a un videojuego, es decir, implementar modelos computacionales de sistemas físicos en un videojuego a través de la aplicación de conocimientos interdisciplinarios.

Además, se necesitaba darle a las actividades el carácter de “bajo umbral” y “alto techo”. Para cumplir con la característica de “bajo umbral”, las actividades se tendrían que poder desarrollar por cualquier estudiante con conocimientos y experiencia básica en el uso de una computadora. Para poder solventar esto, se decidió que los estudiantes utilizarán el motor de videojuegos *Unity*, pero sin escribir una sola línea de código, lo cual es posible en dicho motor de videojuegos. Por otro lado, para cumplir con la característica de “alto techo”, las actividades tendrían que servir como medio para acceder a ideas poderosas (Papert & Resnick, 1996), así como un medio para “construir puentes” entre las disciplinas que son necesarias para poder producir un videojuego. Ahora bien, como la construcción de los videojuegos, en este caso, se centraría en la implementación de modelos computacionales, entonces se diseñaron las actividades tomando en consideración los seis principios de las Actividades Inductoras de Modelos o MEAs (*Model Eliciting Activities*) de Hamilton et al. (2008), las cuales se enlistan a continuación:

- Principio de realidad.
- Principio de construcción del modelo.
- Principio de documentación del modelo.
- Principio de autoevaluación.
- Principio de generalización del modelo.
- Principio de prototipo simple.

La forma en que se relacionan estos principios de diseño, con cada una de nuestras actividades, se muestran a continuación:

3.4.1 Principio de realidad

De acuerdo con Hamilton et al. (2008), este principio debe contestar las siguientes preguntas de diseño: "[1] ¿Puede pasar esto en la “vida real”?, [2] ¿Puede motivar a los estudiantes a dar sentido a la situación, con base en la extensión de sus propios conocimientos y experiencias personales?" (p. 7).

En lo que respecta a la primera pregunta, ésta fue el primer filtro para elegir y diseñar la problemática principal a resolver en nuestras actividades: Que los estudiantes tomaran el papel de un desarrollador de videojuegos que necesita implementar modelos computacionales de sistemas físicos en un videojuego a través de la aplicación de conocimientos interdisciplinarios.

La respuesta a la segunda pregunta se responde con parte de la respuesta anterior, pues se trabajaría con estudiantes de matemática aplicada, y las actividades que les pedimos realizar están orientadas a resolver una problemática que puede presentársele a un profesional de esa disciplina, es decir, que en su contexto sociocultural es una actividad que puede ser altamente significativa para ellos.

Resumiendo, queríamos "alejarnos" lo más posible de los problemas tradicionales que se encuentran en los libros de matemáticas –inclusive en los del nivel superior– y en donde la aplicación de las matemáticas se realiza de forma alejada de la realidad y otras disciplinas, lo cual es muy distinto de lo que sucede en la problemática que planteamos.

3.4.2 Principio de construcción del modelo

De acuerdo con Hamilton et al. (2008), este principio debe contestar las siguientes preguntas de diseño: "[1] ¿involucra la construcción, explicación, manipulación, predicción, o control de un sistema estructural significativo?, [2] ¿crea la tarea, la necesidad de que el modelo sea construido (o modificado, o extendido, o refinado)?" (p. 7). Respecto a la pregunta 1, durante la realización de las actividades por parte de los estudiantes, se espera que sus acciones expresen el desarrollo estructurado de la puesta en práctica de las matemáticas explícitas e implícitas desde un enfoque interdisciplinar.

En cuanto a la pregunta 2, se planteó la resolución de la problemática principal como un conjunto de retos que requerirán de la modelación y construcción de un videojuego, en donde se espera que los estudiantes se vean en la necesidad de realizar ciclos iterativos de construcción en donde la modelación computacional sea la parte esencial.

3.4.3 Principio de documentación del modelo

Este principio debe contestar las siguientes preguntas de diseño: "[1] ¿Pueden los estudiantes revelar de forma explícita la manera en que están pensando respecto a la situación o problemática (datos, metas, rutas de solución posible)? [2] ¿En qué tipo de sistemas (objetos matemáticos, relaciones, operaciones, patrones, regularidades) están ellos pensando?" (Hamilton et al., 2008, p. 7).

La primera pregunta tiene que ver con los instrumentos (cuestionarios) que respondieron los estudiantes, en los cuales se les pide que describan la forma en la que llevan a cabo algunas de las experimentaciones, la modelación computacional, así como el desarrollo del videojuego. También, los mismos programas de las simulaciones y videojuegos sirven para documentar el proceso de modelación y revelar las formas de pensar de los alumnos acerca de la situación o problemática.

Desde el punto de vista de esta investigación, se aprovecha ese proceso de documentación de los alumnos, al igual que, sus programas (simulaciones y videojuegos), como medios importantes de información de sus procesos creativos. Esto se complementa con videograbaciones de las sesiones. Para la segunda pregunta, se clasificó la forma en que lo estudiantes aplicaban y utilizaban las matemáticas en: matemática explícita e implícita. Además, se clasificó la actividad (acciones observadas. Ver sección 4) realizada por los estudiantes, y que se relacionan con la experimentación, modelación y construcción de los videojuegos. Estas acciones observadas no revelan en qué están pensando los estudiantes, sin embargo, sí permiten intuirlo.

3.4.4 Principio de autoevaluación

Este principio debe contestar las siguientes preguntas de diseño: "[1] ¿Sugiere fuertemente el planteamiento del problema, los criterios que son apropiados para la evaluación de la utilidad de respuestas alternativas? [2] ¿Pueden los estudiantes juzgarse (evaluarse) a sí mismos cuando sus respuestas son lo suficientemente buenas? [3] ¿Es claro el propósito que abordan los resultados?, [4] ¿para quién?, ¿cuándo?" (Hamilton et al., 2008, p. 7).

Con base en este principio y sus cuatro preguntas de diseño, se esperaba que el proceso de autoevaluación por parte de los estudiantes se pudiera observar en varios momentos de cada una de las actividades. Por ejemplo, durante los ciclos o proceso de construcción, donde se esperaba también poder observar y documentar las acciones realizadas por ellos durante cada una de las actividades.

Por experiencias previas se esperaba que, en cada una de estas acciones, los estudiantes tuvieran que comprobar, modificar, experimentar, validar, verificar y ajustar sus modelos computacionales, variables, parámetros, etc. Además, que al llevar a cabo estas acciones, los estudiantes deberían realizar autoevaluaciones respecto a lo que estaban construyendo, para decidir si continuar o detener la etapa iterativa del ciclo de construcción en el que estarían inmersos.

3.4.5 Principio de generalización del modelo

Las preguntas de diseño que se deben contestar en este principio son: "¿Es el modelo, no sólo poderoso (para la situación específica), sino también compartible (con otros) y reutilizable (en otras situaciones)?" (Hamilton et al., 2008, p. 7).

En esta pregunta, se esperaba que en el momento en que los estudiantes plantearan un modelo computacional (o simulación) de la problemática a resolver, generaran las restricciones y relaciones necesarias en el motor de videojuegos, que permitieran la reutilización de dicho modelo en otras situaciones, otros videojuegos e inclusive —con algunos cambios— en otras herramientas tecnológicas.

Además, en algunos casos se espera que, aunque las actividades se realizaran de manera individual, no existiera una restricción sobre el hecho de que los estudiantes pudieran compartir ideas de cómo construir algunos modelos computacionales o videojuegos.

3.4.6 Principio del prototipo simple

Las preguntas relacionadas con este principio son: "[1] ¿Es la situación lo más simple posible, creando al mismo tiempo la necesidad de un modelo significativo? [2] ¿La solución proporcionará un prototipo útil (o metáfora) para interpretar otras situaciones estructuralmente similares?" (Hamilton et al., 2008, p. 7).

Se esperaba que los estudiantes representaran de manera preliminar las características generales de los modelos a implementar a través de un "tanteo dirigido" de los valores de las propiedades físicas de los objetos virtuales involucrados. Es decir, que los estudiantes experimentaran con la matemática y la física del modelo de manera implícita, pues esto es la base para que los estudiantes puedan acceder a la creación de un modelo más complejo en las actividades de modelación y construcción del videojuego.

3.5 Implementación de las actividades

Como se comentó en al inicio de la sección 3, el taller tuvo una duración de 15 horas, las cuales se dividieron en cinco sesiones de tres horas cada una. De tal forma que, la primera sesión estuvo destinada a la Actividad de experimentación, en la segunda y tercera sesiones se abordó Actividad de modelación, y en la cuarta y quinta sesiones la Actividad de construcción del videojuego.

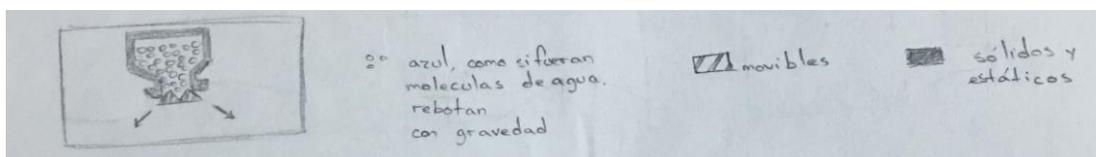
Durante la primera sesión se les dio a los estudiantes una breve plática acerca de la metodología que se llevaría a cabo (ver Figura 4.1, sección 3.1).

Después de esto, se pidió a los estudiantes que contestaran un cuestionario exploratorio que contenía seis secciones: (1) conocimientos técnicos, (2) sobre modelos matemáticos, (3) sobre simulación, (4) sobre modelado y simulación, (5) simulación y videojuegos y (6) emotiva y contextual. Este cuestionario sirvió para definir el perfil de los participantes, conocer sus competencias tecnológicas; su conocimiento de conceptos relacionados con los modelos matemáticos, modelado y simulación; así como, las diferencias entre simulación y videojuego; además de conocer sus motivaciones para asistir al taller.

Posteriormente, se comenzó a explicar el funcionamiento de los objetos virtuales y los objetos físicos virtuales en *Unity*. Además, se les comentó que las construcciones que se realizarían durante el taller no involucrarían la programación a través de código escrito. Después, se pidió a los estudiantes que comenzaran a experimentar, a base de tanteos dirigidos, con los parámetros físicos de los objetos físicos virtuales y que realizaran sus primeras simulaciones experimentales. En esta Actividad de experimentación los estudiantes no trabajaron de manera explícita con ecuaciones o fórmulas matemáticas sino que, al principio, ellos crearon objetos virtuales, que comenzaron a manipular para observar su comportamiento. De esta manera, es posible intuir que ellos crearon ciertas relaciones implícitas (sin usar ninguna fórmula o ecuación matemática) entre los parámetros físicos virtuales y las fórmulas o conceptos relacionados con dichos comportamientos. Debido a lo anterior, esta etapa resultó crucial para el entendimiento, por parte de los estudiantes, del funcionamiento del motor físico en general.

En la sesión dos y tres, ya que los estudiantes habían experimentado libremente, se les pidió que realizaran predicciones antes de construir una nueva simulación experimental. Se les propuso que dibujaran los escenarios y objetos que pretendían construir y predijeran el comportamiento de la simulación o modelo computacional que tendrían en *Unity*. Un ejemplo de estas predicciones puede observarse en la Figura 4.2 y 4.3.

Figura 4.2 Moléculas de agua en un contenedor. Predicción de Alonso



Fuente: Producción de los estudiantes

Este ejercicio de predecir a partir de dibujos o bosquejos, como en la Figura 4.2, se considera (en esta investigación) en términos muy básicos, un tipo de modelación matemática donde, tanto los conceptos matemáticos como físicos estaban implícitos, pues no contienen ningún tipo de ecuación o fórmula, sin embargo constituye el paso previo para construir un modelo computacional como el de la Figura 4.3.

Además, es un ejercicio que permitió que los estudiantes comenzaran a incubar ideas poderosas respecto a la herramienta digital que estaban manejando y en torno al modelado computacional como una actividad interdisciplinaria donde se complementa la matemática, la física, la informática y hasta el diseño gráfico, al momento de crear los escenarios que sirven como interfaces de usuario.

Figura 4.3 El modelo computacional de las moléculas de agua de Alonso



Fuente: Producción de los estudiantes

Al inicio de la cuarta sesión se pidió a los estudiantes que descargaran un *Asset* que contenía varios recursos (fondos, sonidos, escenarios y *sprites*) que facilitarían la construcción de un videojuego; luego, que construyeran un videojuego que incorporara, en su modo de juego, alguno de los modelos con los que habían estado experimentando.

Figura 4.4 El primer videojuego “físico” de Alonso

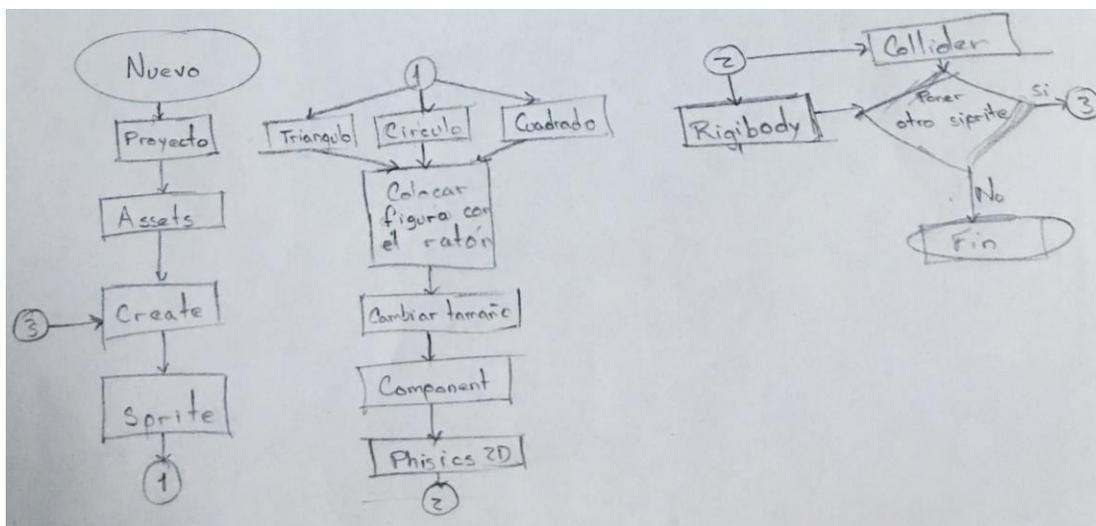


Fuente: Producción de los estudiantes

En la Figura 4.4, se muestra una captura de pantalla del primer videojuego de Alonso, en donde incorpora además de los elementos estéticos y un personaje, varios obstáculos en forma de “pelotas” que rebotan y caen por acción de la gravedad (virtual), impidiéndole el paso al personaje que es manipulado por el usuario.

En la quinta y última sesión se les pidió a los estudiantes que dibujaran un diagrama que representara la forma en la que habían llevado a cabo su ciclo de construcción de los modelos computacionales que, posteriormente, eran incorporados al videojuego. En la Figura 4.5, se muestra la representación del ciclo de construcción de Alonso.

Figura 4.5 El ciclo de construcción de Alonso

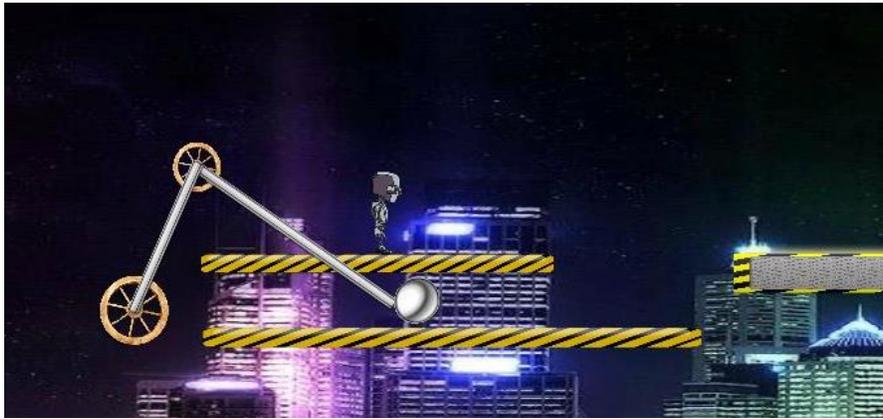


Fuente: Producción de los estudiantes

En el ciclo de construcción de la Figura 4.5, se observa que Alonso basa su ciclo de construcción en los elementos que se tienen que manipular o configurar desde *Unity* y los relaciona con la forma en la que configura la física y la apariencia de un objeto virtual.

Después, en la última sesión se entregó una hoja de “retos” a los estudiantes, donde se mostraban tres imágenes de modelos físicos: El primero era un mecanismo biela – manivela – corredera, el segundo eran las “orugas” de un tanque y el último era un puente colgante. La idea de darles estos retos fue que los construyeran y, posteriormente, los incorporaran en un videojuego. En la Figura 3.6, se muestra el modelo computacional de un mecanismo biela – manivela – corredera incorporado en el videojuego de Alonso.

Figura 4.6 Mecanismo biela – manivela – corredera en el videojuego de Alonso



Fuente: Producción de los estudiantes

La incorporación de estos sistemas físicos en los videojuegos enfrentó a los estudiantes con el reto de tener que construir sistemas con los que no están acostumbrados a “trabajar” debido a su formación, pero lo hicieron de forma correcta, ya que los sistemas operaban de manera adecuada. Al final, los estudiantes quedaron muy sorprendidos, porque al principio se habían declarado un poco escépticos con los alcances que podrían tener, construyendo modelos computacionales y videojuegos sin utilizar una sola línea de código. Al término del taller, habían diseñado y construido varios modelos computacionales de sistemas físicos y dos videojuegos, lo que reafirma el hecho de que esta metodología tiene un potencial alto si se requiere tener actividades con un bajo umbral y un alto techo.

3.6 Recolección de datos

La recolección de datos se llevó a cabo a través de:

- Videograbaciones de las sesiones.
- Cuestionarios hechos a los estudiantes (individuales), las cuales siempre se realizaban al principio de cada sesión. En ellos describieron sus avances, reflexiones y dificultades en la realización de las diferentes actividades.
- El programa de las simulaciones y videojuegos, el cual se consultó al momento de revisar a detalle alguno de los procedimientos descritos, o bien, alguno de los modelos computacionales implementados.

Aunque se cruzó la información recabada con cada uno de estos instrumentos, se puso especial énfasis en lo que los estudiantes respondieron en los cuestionarios personales y en las videograbaciones, en donde se pudo observar cómo habían desarrollado sus ciclos de construcción (qué habían hecho, cómo lo habían hecho y para qué lo habían hecho). Sin embargo, también se notó que algunas “acciones observadas” no fueron externadas (probablemente no fueron concientizadas) por los estudiantes –cuando contestaban sus cuestionarios– al describir sus ciclos de construcción.

4 Presentación y discusión de los resultados

Las “acciones realizadas” por los estudiantes en los ciclos o proceso de construcción de cada una de las simulaciones y videojuegos que realizaron, se presentan de manera condensada en tres tablas que las clasifican por tipo de actividad: Tabla 4.2 para la actividad de experimentación; Tabla 4.3 para la actividad de modelación computacional; y, Tabla 4.4 para la actividad de construcción del videojuego.

4.1 Experimentación

Este proceso de experimentación fue visto por los estudiantes como una etapa previa a la modelación matemática y computacional, donde a través de “tanteos dirigidos” se manipulaba o “jugaba” libremente con los objetos virtuales y sus propiedades físicas. A través de dichos procesos, los estudiantes podían comprender el comportamiento físico-matemático de los objetos virtuales de manera implícita; es decir, sin utilizar ecuaciones matemáticas o procedimientos analíticos de comprobación, sino sólo mediante la observación.

Las “acciones observadas” durante la actividad de experimentación, representan la forma en la que los estudiantes llevaron a cabo, o pusieron en práctica, la matemática de manera implícita. En conjunto, estas acciones muestran el proceso por medio del cual los estudiantes lograron abstraer la “información” o significados matemáticos (o físicos) que necesitaban para realizar la experimentación a través de la manipulación de objetos virtuales (Tabla 4.2).

Tabla 4.2 Acciones observadas en la experimentación

Actividad: Experimentación	Código de acción
Actividad 1: Experimentación y validación de propiedades físicas de los objetos a modelar en la problemática.	
Se definen los objetos (en el mundo real).	1.1
Se obtienen o crean recursos gráficos y estéticos para la experimentación.	1.2
Se crean objetos virtuales en <i>Unity 2D</i> .	1.3
Se definen propiedades físicas de los objetos virtuales.	1.4
Se prueba la simulación (modelación computacional).	1.5
Se experimenta y observa el comportamiento de los objetos.	1.6
Si el objeto en la simulación no se comporta de acuerdo con la realidad, se experimenta cambiando sus propiedades “físicas” virtuales	1.7
Se concluye el ciclo de refinamiento en la construcción de la simulación al comprobar su funcionamiento correcto	1.8

Fuente: Elaboración propia

4.2 Modelación computacional

En la Tabla 4.3, se enlistan las “acciones observadas” durante las actividades de modelación computacional desarrolladas por los estudiantes. Se considera que hay un conjunto de tres “acciones observadas” (2.2, 2.4 y 2.5) donde es posible inferir claramente que los estudiantes hicieron uso de una matemática implícita. La acción 2.2 es una de ellas, porque en el momento que los estudiantes hacen un bosquejo o dibujo acerca del modelo que deberían de plantear en una simulación (modelación computacional) y cómo deberían presentarlo, están abstrayendo información y/o significados matemáticos para poder generar cada una de las ideas. Respecto a las otras dos acciones, 2.4 y 2.5, el uso de la matemática implícita es aún más claro. La primera acción se refiere a la manipulación de objetos virtuales y sus propiedades para entender su funcionamiento en los “mundos físicos” creados en *Unity*. La segunda se refiere a la realización de experimentos con matemática implícita.

Tabla 4.3 Acciones observadas en la modelación computacional

Actividad: Modelaciones computacional	Código de acción
Actividad 2: Construcción de la simulación (modelación computacional) de la problemática	
Se lleva a cabo una lluvia de ideas	2.1
Se hace un bosquejo o dibujo para representar el modelo.	2.2
Se familiarizan con comandos básicos y configuración del motor físico de <i>Unity 2D</i> .	2.3
Se crean y manipulan objetos virtuales para entender cómo funcionan en <i>Unity</i> ; se manipulan (se “juega” con) sus propiedades físicas.	2.4
Se llevan a cabo experimentos con matemática y física implícita.	2.5
Se obtienen o crean recursos gráficos y estéticos para la simulación. Se crean <i>sprites</i> .	2.6
Se definen objetos en <i>Unity</i> , así como sus propiedades físicas, para implementarlos en la simulación	2.7
Se asignan <i>sprites</i> a los objetos.	2.8
Se modifican los <i>sprites</i> para que sean estéticamente más funcionales [a juicio del estudiante].	2.9
Se prueba el programa (el modelo) de <i>Unity</i> para validar los parámetros (propiedades) físicos virtuales.	2.10
Se llevan a cabo ajustes en las propiedades físicas (gravedad, densidad, forma, fricción, etc.) y la forma de los objetos físicos virtuales.	2.11
Se concluye el ciclo de refinamiento en la construcción de la simulación, al comprobar visualmente su funcionamiento correcto	2.12

Fuente: Elaboración propia

4.3 Construcción del videojuego

Por último, en la Tabla 4.4 se muestran las “acciones observadas” durante las actividades de construcción del videojuego. En esta tabla se observa que dos de las acciones (3.2 y 3.3) corresponden al uso de una matemática implícita.

Cabe mencionar, que estas “acciones observadas” en los estudiantes (Tabla 4.2, 4.3 y 4.4) se desarrollaron en algunos casos de manera recursiva, debido a que los ajustes que los estudiantes hacían a ciertos parámetros físicos en los objetos se basaban en las ideas erróneas que tenían de algunos conceptos físicos. Por ejemplo, muchos de ellos relacionaban erróneamente el concepto de “peso” (fuerza gravitacional que actúa sobre un objeto) con el de “masa” (cantidad de materia de un objeto); también, relacionaban el concepto de “masa”, solamente con el “tamaño” (área) del objeto físico virtual, y no tomaban en cuenta la “densidad” del mismo, lo que en algunos casos los llevaba a realizar pruebas cambiando sólo las dimensiones del objeto y, por consiguiente, no obtenían los resultados esperados.

Tabla 4.4 Acciones observadas en la construcción del videojuego

Acciones observadas en los ciclos de construcción	Código de acción
Actividad 3: Construcción del videojuego	
Se aprenden funciones especializadas del motor físico.	3.1
Se realizan pruebas de funciones de propósito específico.	3.2
Se genera una lluvia de ideas para escoger una propuesta de videojuego.	3.3
Se crean o se reutilizan <i>sprites</i> (personajes, enemigos, etc.), así como los escenarios.	3.6
Se crean objetos y se les asignan <i>sprites</i> .	3.7
Se produce la validación y funcionamiento del juego.	3.8
Se lleva a cabo el ajuste del modelo en el programa con respecto a las propiedades físicas de los objetos virtuales (gravedad, densidad, amortiguamiento lineal y angular, coeficiente de restitución).	3.9
Se realiza la simulación y depuración del videojuego.	3.10
Se concluye el ciclo de refinamiento en la construcción del videojuego al comprobar su funcionamiento correcto.	3.11

Fuente: Elaboración propia

5 Agradecimiento

Esta investigación es producto de las actividades realizadas en el taller “Experimentación y modelación (matemática y computacional) para la construcción de videojuegos”, llevado a cabo y financiado en la Facultad de Ciencias Exactas de la UJED (Durango, México) en abril de 2018, dentro del ciclo “Variedades Matemáticas” en el marco del VIII Encuentro de Matemáticas y es parte también de los productos desarrollados y financiados en el Proyecto SIP2018083 en el Instituto Politécnico Nacional (México).

Por lo anterior, se extiende mi más sincero agradecimiento a la Facultad de Ciencias Exactas de la UJED y al Instituto Politécnico Nacional. Además quiero extender mi profundo agradecimiento a los estudiantes Dulce María Reyes Rojas, Eybette Mercado Favela, Alonso Eloy Ávila Dévora y Jafed A. Martínez Sánchez por haber compartido sus simulaciones y videojuegos desarrollados durante el taller.

6 Conclusiones

Sobre la interdisciplinariedad e integración del conocimiento

Implementar un modelo matemático en un videojuego puede ser una actividad nueva en el contexto de algunas especialidades. De hecho, algunos de los estudiantes comentaron que era algo que nunca habían hecho. Y, como se observó, fueron actividades que los enfrentaron a nuevas formas de representar sus conocimientos e ideas.

En consecuencia, se considera que este tipo de actividades pueden enriquecer significativamente el acervo de experiencias formativas de un alumno de educación superior.

Se considera esto como un aporte importante para los estudiantes de matemática aplicada, ya que la implementación de un modelo matemático en la construcción de un videojuego abre la posibilidad de que los estudiantes se enfrenten a nuevas experiencias de aprendizaje y creación a partir de la resolución de una problemática interdisciplinaria real.

Más específicamente, al tener que implementar un modelo matemático en un videojuego se tienen que poner en práctica matemáticas del mundo real, de manera que éstas estén articuladas e integradas con las otras disciplinas, y así, se fortalezca el conocimiento y la experiencia de cada disciplina en particular. En contraste, con lo que sucede desde una enseñanza de conocimiento fragmentado. ¿Por qué es importante la integración del conocimiento?, una respuesta puede ser: porque prepara a los estudiantes, para la vida profesional, poniéndolos en la zona de conflicto, donde la pregunta constante es, ¿cómo aplico todo el conocimiento teórico que tengo en una problemática real?

Sobre la matemática implícita

Como resultado de la implementación de las actividades, se ha observado entre otras cosas, que en muchas ocasiones la actividad matemática de los estudiantes no se manifiesta de manera explícita, es decir de una manera clara y directa a través, por ejemplo, del uso de ecuaciones o fórmulas para calcular o inferir un resultado; sino que, los estudiante abstraen la “información” o significados matemáticos (o a veces físicos o ingenieriles) que necesitan, de la experimentación o manipulación que realizan de ciertos objetos (en este caso virtuales) para construir una simulación o un videojuego.

Se infiere que dicha experimentación ha tenido como consecuencia que los estudiantes construyeran relaciones abstractas de las interacciones entre objetos virtuales, sus parámetros físicos y el mundo virtual, por ejemplo en el apartado 3.5 de este capítulo, se muestra cómo el estudiante Alonso construyó un mecanismo biela-manivela-corredera a partir de observar una imagen y varias animaciones del mecanismo, para después a partir de varios ciclos de experimentación (y manipulación de parámetros físicos), lograr que el mecanismo funcionara cumpliendo las leyes físicas del mundo virtual. También, se observó que los estudiantes manifestaban las relaciones a través de conceptos, diagramas o enunciados que describían el comportamiento de los objetos, sin utilizar ninguna ecuación matemática, fórmula física o código de programación. En estos casos, se consideró, por tanto, que ponían en práctica, o usaban la matemática de manera implícita.

Sobre el trabajo a futuro

Por otro lado se espera que esta propuesta no funcione como una “receta de cocina” la cual hay que seguir al pie de la letra o no se obtiene el sazón adecuado, sino al contrario, la idea es continuar probándola en diversos escenarios o contextos, con diferentes herramientas tecnológicas, y con estudiantes de otros niveles o áreas de estudio, para ver qué tanto puede refinarse, adaptarse o mejorar, para que los estudiantes puedan definir, refinar, transformar y extender los conocimientos teóricos que tiene hasta poderlos aplicar de manera integral a problemas de la vida real.

7 Referencias

- Ackermann, E. (2001). Piaget’s constructivism, Papert’s constructionism: What’s the difference. *Future of learning group publication*, 5(3), 438-449.
- Araujo, I.S., Veit, E.A., & Moreira, M.A. (2007). Um estudo exploratório sobre as potencialidades do diagrama AVM na aprendizagem significativa de tópicos de Física. *Indivisa, Boletín de Estudios e Investigación, Monografía VIII*, 503-514.
- Badilla, E. & Chacón, A. (2004). Construccinismo: Objetos para pensar, entidades públicas y micromundos. *Revista Electrónica Actualidades Investigativas en Educación*, 4(1), 1-12. Recuperado de: http://revista.inie.ucr.ac.cr/uploads/tx_magazine/construccinismo.pdf
- Dodge, Y. (2008). *The Concise Encyclopedia of Statistics*. New York: Springer.

- Bliss, J., Ogborn, J., Boohan, R., Briggs, J., Brosnan, T., Brough, D., Mellar, H., Miller, R., Nash, C., Rodgers, C., & Sakonidis, B. (1992). Reasoning supported by computational tools. *Computers in Education*, 18(1-3), 1-9.
- Bliss, J. & Ogborn, J. (1989). Tools for exploratory learning. *Journal of Computer Assisted Learning*, 5(1), 37-50.
- Doerr, H.M. (1995). An integrated approach to mathematical modeling: A classroom study. Paper presented at the Annual meeting of the American Educational Research Association, San Francisco, CA, April 18-22.
- Hamilton, E., Lesh, R., Lester, F., & Brilleslyper, M. (2008). Model-eliciting activities (MEAs) as a bridge between engineering education research and mathematics education research. *Advances in Engineering Education*, 1(2), 1-25.
- Harel, I. (1990). Children as Software Designers: A Constructionist Approach for Learning Mathematics. *Journal of Mathematical Behavior*, 9(1), 3-93.
- Hestenes, D. (2010). Modeling theory for math and science education. En R. Lesh, P. L. Galbraith, C. R. Haines, & A. Hurford (Eds.), *Modeling students' mathematical modeling competencies* (pp. 13-41). New York: Springer.
- Hoyle, C. & Noss, R. (2005). Crear reglas en el diseño de juegos de colaboración. En J. Siraj-Blatchford (Comp.), *Nuevas Tecnologías para la educación infantil y primaria* (pp. 72-91). Madrid: Ministerio de Educación y Ciencia.
- Ifenthaler, D. (2012). Computer Simulation Model. In N. Seel (Ed.), *Encyclopedia of the sciences of learning* (pp. 710-713). Springer US. doi:10.1007/978-1-44191428-6_500
- Kafai, Y. (1994). *Minds in Play: Computer Game Design as a Context for Children's Learning*. Mahwah, NJ: Lawrence Erlbaum.
- Kafai, Y. B. & Burke, Q. (2015). Constructionist gaming: Understanding the benefits of making games for learning. *Educational Psychologist*, 50(4), 313-334.
- Lesh, R. & Doerr, H. M., (2003). Foundations of a models and modelling perspective on mathematics teaching, learning, and problem solving. En R. Lesh & H. M. Doerr (Eds.), *Beyond constructivism. Models and modelling perspectives on mathematics problem solving, learning and teaching*, (pp. 3-33). Mahwah, NJ: Lawrence Erlbaum Associates.
- Lesh, R. & Harel, G. (2003). Problem solving, modeling, and local conceptual development. *Mathematical thinking and learning*, 5(2&3), 157-189.
- López, S., Veit, E., & Araujo, I. (2011). Modelación computacional apoyada en el uso del diagrama V de Gowin para el aprendizaje de conceptos de dinámica newtoniana. *Revista Electrónica de Enseñanza de las Ciencias*, V10(1), 202-226.
- Papert, S. (1981). *Desafío de la mente*. Buenos Aires, Argentina: Galápagos.
- Papert, S. & Harel, I. (1991). Situating Constructionism. En I. Harel & S. Papert (Eds.) *Constructionism*. Recuperado de <http://www.papert.org/articles/SituatingConstructionism.html>
- Papert, S. & Resnick, M. (1996). Rescuing the powerful ideas, en NSF Symposium, MIT.
- Piaget, J. (1968). *Educación e instrucción*. Buenos Aires, Argentina: Proteo.
- Pretelín-Ricárdez, A. y Sacristán, A. I. (2015). Videogame Construction by Engineering Students for Understanding Modelling Processes: The Case of Simulating Water Behaviour. *Informatics in education*, 14(2), 265-277. doi:10.15388/infedu.2015.15