



**UNIVERSIDAD DEL BÍO-BÍO**  
**Facultad de ciencias empresariales**  
**Departamento de sistemas de información**

**17 de enero de 2018**

**Proyecto de título: “Estudio del impacto de incorporar videojuegos al proceso de aprendizaje de fracciones matemáticas”**

**Pablo Castillo Burgos**

**Profesor guía:  
Clemente Rubio Manzano**

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad de Bío-Bío en el proceso de titulación para la carrera de Ingeniería Civil Informática

## Resumen

Este proyecto se presenta para dar conformidad a los requisitos exigidos por la Universidad de Bío-Bío en el proceso de titulación para la carrera de Ingeniería Civil Informática. El proyecto se titula “Estudio del impacto de incorporar videojuegos al proceso de aprendizaje de fracciones matemáticas”. Forma parte de un proyecto interno de la Universidad del Bío-Bío llamado “Estudio del impacto de incorporar videojuegos al proceso de enseñanza-aprendizaje de fracciones en centros educacionales de la Octava Región” que tiene como objetivo utilizar los videojuegos con el fin de mejorar el aprendizaje y la comprensión de las fracciones. La intención es entender de qué modo varía la enseñanza de fracciones cuando es complementada con el uso de videojuegos. Para esto se diseñará un videojuego con la ayuda del profesor de la Universidad del Bío-Bío, Jairo Navarrete, experto en didáctica de las matemáticas y tres estudiantes en proceso de titulación, cuyo proyecto se asemeja lo suficiente a éste como para permitirnos cooperar en el diseño en un principio.

La investigación arroja resultados variados, esto es debido a que la manera de medir el impacto resultó ser indirecta, el videojuego ejercita mayoritariamente equivalencias entre fracciones y la forma de controlarlo fue ubicar fracciones en la recta geométrica, requiriendo a los alumnos poseer cierto conocimiento previo para poder vincular ambas habilidades. En los controles, los alumnos de 4° básico lograron reducir su margen de error al ubicar los números en la recta un 59% respecto a cómo estaban antes de probar el videojuego, pero los alumnos de 3° básico no experimentaron ninguna mejora en el test. Vale decir también, que con esta investigación se facilita la realización de futuros proyectos de la misma índole para no tener que comenzar desde cero.

## Abstract

This project is presented to comply with the requirements demanded by the University of Bío-Bío in the process of qualification for the career of Computer Engineering. The project is entitled "Study of the impact of incorporating videogames into the process of learning mathematical fractions". It is part of an internal project of the University of Bío Bío called "Study of the impact of incorporating videogames into the teaching-learning process of fractions in educational centers of the Eighth Region" that aims to use videogames in order to improve the learning and the comprehension of fractions. The intention is to understand how the teaching of fractions varies when is complemented by the use of video games. For this a video game will be designed with the help of the professor of the University of Bío-Bío, Jairo Navarrete, expert in mathematics didactics and three students in the process of titling, whose project is similar enough to this one to allow us to cooperate in the design initially.

The research yields varied results, this is due to the way to measure the impact turned out to be indirect, the video game exercises mostly equivalences between fractions and the way to control it was to locate fractions in the geometric line, requiring the students to possess certain prior knowledge for to be able to link both abilities. In the controls, the 4th grade students managed to reduce their margin of error by placing the numbers on the line by 59% compared to how they were before trying the game, but the students of 3rd grade did not experience any improvement in the test. It is also worth mentioning that this research facilitates the realization of future projects of the same nature by not having to start from scratch.

## Estructura del documento

Para la confección del presente informe se utilizó como base la plantilla proporcionada por la Universidad, pero debido a la naturaleza de éste proyecto se decidió modificarlo, a continuación se explica el contenido de cada uno de sus capítulos:

Primero en la sección de Introducción se describirá la problemática. Entre los temas que se abordarán se encuentran los resultados de PISA en la competencia matemática, la necesidad de buscar mecanismos que ayuden a aumentar la motivación mediante nuevas herramientas. Se revisará los videojuegos como una alternativa al aprendizaje de matemáticas analizando su historia y su impacto en los niños.

El capítulo dos está enfocado a explicar cuáles son los objetivos de esta investigación y que procedimientos y herramientas serán utilizadas para su desarrollo.

El capítulo tres explica cuál es el contexto en el que se lleva a cabo éste proyecto, es importante mencionar que el presente proyecto forma parte de otro, interno en la Universidad, por lo que amerita describirlo.

El capítulo cuatro describe la base pedagógica de las analogías matemáticas que fueron el foco del diseño de las mecánicas del videojuego.

El capítulo cinco comienza a describir cuales son los objetivos del videojuego al momento de diseñarlo. La ausencia de requerimientos formales se debe a que un videojuego, a diferencia de un sistema de información más convencional, no está tan bien estructurado en cuanto a sus características desde el principio. En su lugar, se diseñaron mecánicas enfocadas a las analogías del capítulo anterior, y luego se les dio un contexto atractivo para que un niño lo juegue.

El capítulo seis entrega un breve análisis de factibilidad, ya que el proyecto no supone gastos ni ganancias, el análisis es breve.

El capítulo siete explica cuál es el resultado del diseño del videojuego. Da una explicación general, luego explica en detalle todas sus mecánicas en formato de casos de uso, posteriormente explica cómo funciona el motor sobre el cual se trabajará y por último muestra cómo estarán distribuidos los elementos en el juego.

El capítulo ocho trata de la implementación propiamente tal. Explica cuál es el patrón de diseño del template utilizado y detalla el diagrama de clases que representa la estructura del código.

El capítulo nueve está dedicado a la investigación, primero explica el método que se ideó para controlar a los alumnos, luego se detalla cómo estas ideas fueron llevadas a la práctica, y finalmente se detallan y analizan los resultados obtenidos.

Los últimos capítulos corresponden a la realización de las pruebas de software en el capítulo diez. En el capítulo once se hace una estimación de las horas dedicadas y se

contrastan con la estimación inicial del proyecto. El capítulo doce trata sobre las conclusiones del proyecto, indicando sus resultados y el impacto académico y personal hacia el desarrollador. El capítulo trece corresponde a la bibliografía utilizada. Y finalmente el capítulo catorce muestra cual era la planificación inicial del proyecto y estima su tamaño en base a sus casos de uso.

# Índice General

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>INTRODUCCIÓN.....</b>                            | <b>9</b>  |
| 1.1      | DESCRIPCIÓN DE LA PROBLEMÁTICA .....                | 9         |
| 1.2      | ESTADO DEL ARTE .....                               | 9         |
| <b>2</b> | <b>DEFINICIÓN PROYECTO .....</b>                    | <b>11</b> |
| 2.1      | OBJETIVOS DEL PROYECTO .....                        | 11        |
| 2.2      | AMBIENTE DE INGENIERÍA DE SOFTWARE .....            | 11        |
| 2.3      | DEFINICIONES, SIGLAS Y ABREVIACIONES .....          | 12        |
| <b>3</b> | <b>DEFINICION DE LA EMPRESA O INSTITUCIÓN .....</b> | <b>13</b> |
| 3.1      | DESCRIPCIÓN DE LA EMPRESA .....                     | 13        |
| 3.2      | DESCRIPCIÓN DEL ÁREA DE ESTUDIO .....               | 14        |
| <b>4</b> | <b>BASE PEDAGÓGICA .....</b>                        | <b>16</b> |
| 4.1      | CONTEXTO .....                                      | 16        |
| 4.2      | DIAGRAMAS ANALÓGICOS .....                          | 17        |
| 4.2.1    | COMPARACIÓN .....                                   | 17        |
| 4.2.2    | EQUIVALENCIA .....                                  | 17        |
| 4.2.3    | SUMA .....  | 18        |
| 4.2.4    | RESTA .....   | 18        |
| <b>5</b> | <b>ESPECIFICACIÓN DE SOFTWARE .....</b>             | <b>19</b> |
| 5.1      | ALCANCES .....                                      | 19        |
| 5.2      | OBJETIVO DEL SOFTWARE .....                         | 19        |
| <b>6</b> | <b>ESTUDIO DE FACTIBILIDAD .....</b>                | <b>20</b> |
| 6.1      | FACTIBILIDAD TÉCNICA. ....                          | 20        |
| 6.2      | FACTIBILIDAD OPERATIVA. ....                        | 20        |
| 6.3      | FACTIBILIDAD ECONÓMICA. ....                        | 20        |
| 6.4      | CONCLUSIÓN DE LA FACTIBILIDAD .....                 | 21        |
| <b>7</b> | <b>DISEÑO.....</b>                                  | <b>22</b> |
| 7.1      | DESCRIPCIÓN GLOBAL DEL VIDEOJUEGO .....             | 22        |
| 7.2      | MECÁNICAS.....                                      | 23        |
| 7.2.1    | ACTORES .....                                       | 23        |
| 7.2.2    | CASOS DE USO Y DESCRIPCIÓN .....                    | 23        |
| 7.2.3    | ESPECIFICACIÓN DE LOS CASOS DE USO .....            | 23        |
| 7.3      | DISEÑO DE ARQUITECTURA FUNCIONAL .....              | 27        |
| 7.4      | DISEÑO INTERFAZ Y NAVEGACIÓN .....                  | 28        |
| 7.4.1    | EXPLICACIÓN DE LA SECCIÓN 1 .....                   | 28        |
| 7.4.2    | EXPLICACIÓN DE LA SECCIÓN 2 .....                   | 29        |
| 7.4.3    | EXPLICACIÓN DE LA SECCIÓN 3 .....                   | 30        |
| <b>8</b> | <b>IMPLEMENTACIÓN.....</b>                          | <b>32</b> |
| 8.1      | PATRÓN DE DISEÑO .....                              | 32        |
| 8.2      | DIAGRAMA DE CLASES.....                             | 33        |

|           |   |                  |
|-----------|---|------------------|
| 8.2.1     | ESPECIFICACIÓN CLASE MAIN .....                               | 34               |
| 8.2.2     | ESPECIFICACIÓN PAUSESTATE .....                               | 34               |
| 8.2.3     | ESPECIFICACIÓN GAMESTATE.....                                 | 35               |
| 8.2.4     | ESPECIFICACIÓN CONSEJO .....                                  | 36               |
| 8.2.5     | ESPECIFICACIÓN PREMIOS .....                                  | 36               |
| 8.2.6     | ESPECIFICACIÓN RULETA .....                                   | 37               |
| 8.2.7     | ESPECIFICACIÓN RECURSOS .....                                 | 37               |
| 8.2.8     | ESPECIFICACIÓN PISTA .....                                    | 38               |
| 8.2.9     | ESPECIFICACIÓN RESERVA.....                                   | 39               |
| 8.2.10    | ESPECIFICACIÓN ARRASTRADOR .....                              | 39               |
| 8.2.11    | ESPECIFICACIÓN SENSORCLICK .....                              | 40               |
| <b>9</b>  | <b><u>INVESTIGACIÓN.....</u></b>                              | <b><u>41</u></b> |
| 9.1       | DISEÑO DE LOS CONTROLES .....                                 | 41               |
| 9.2       | PROGRAMACIÓN Y DISEÑO GRÁFICO DE LOS CONTROLES .....          | 41               |
| 9.3       | RESULTADOS .....  | 45               |
| <b>10</b> | <b><u>PRUEBAS .....</u></b>                                   | <b><u>48</u></b> |
| 10.1      | ELEMENTOS DE PRUEBA.....                                      | 48               |
| 10.2      | ESPECIFICACIÓN DE LAS PRUEBAS.....                            | 48               |
| 10.3      | RESPONSABLES DE LAS PRUEBAS.....                              | 48               |
| 10.4      | CALENDARIO DE PRUEBAS .....                                   | 48               |
| 10.5      | DETALLE DE LAS PRUEBAS.....                                   | 49               |
| 10.5.1    | PU-01 FUNCIONALIDADES DEL VIDEOJUEGO .....                    | 49               |
| 10.5.2    | PU-02 FUNCIONALIDADES DE LOS CONTROLES .....                  | 50               |
| 10.6      | CONCLUSIONES DE PRUEBAS .....                                 | 51               |
| <b>11</b> | <b><u>RESUMEN ESFUERZO REQUERIDO.....</u></b>                 | <b><u>52</u></b> |
| <b>12</b> | <b><u>CONCLUSIONES .....</u></b>                              | <b><u>53</u></b> |
| <b>13</b> | <b><u>BIBLIOGRAFÍA .....</u></b>                              | <b><u>54</u></b> |
| <b>14</b> | <b><u>ANEXO: PLANIFICACION INICIAL DEL PROYECTO .....</u></b> | <b><u>56</u></b> |
| 14.1.1    | ESTIMACIÓN INICIAL DE TAMAÑO .....                            | 56               |

## Índice de Tablas

|  |    |
|--|----|
| Tabla 1. Especificación de pruebas .....             | 48 |
| Tabla 2. Prueba de funcionalidad videojuego .....    | 50 |
| Tabla 3. Prueba funcionalidad de los controles ..... | 51 |
| Tabla 4. Estimación de esfuerzo requerido.....       | 52 |
| Tabla 5. Estimación de tamaño por casos de uso ..... | 58 |

## Índice de Figuras

|   |    |
|---|----|
| Fig. 1 Organigrama UBB .....                    | 14 |
| Fig. 2 Organigrama FACE .....                   | 15 |
| Fig. 3 Analogía: comparación .....              | 17 |
| Fig. 4 Analogía: equivalencia .....             | 17 |
| Fig. 5 Analogía: suma .....                     | 18 |
| Fig. 6 Analogía: resta .....                    | 18 |
| Fig. 7 Casos de uso .....                       | 23 |
| Fig. 8 Arquitectura de Luxe .....               | 27 |
| Fig. 9 Diseño general .....                     | 28 |
| Fig. 10 Ejemplo fichas existentes .....         | 28 |
| Fig. 11 Ejemplo fichas agotadas .....           | 28 |
| Fig. 12 Ejemplo de reservas .....               | 29 |
| Fig. 13 Sección 2 .....                         | 29 |
| Fig. 14 Instrucciones .....                     | 30 |
| Fig. 15 Sección 3 .....                         | 30 |
| Fig. 16 Martillos .....                         | 31 |
| Fig. 17 Patrón de diseño .....                  | 32 |
| Fig. 18 Diagrama de Clases .....                | 33 |
| Fig. 19 Especificación Main .....               | 34 |
| Fig. 20 Especificación PauseState .....         | 34 |
| Fig. 21 Especificación GameState .....          | 35 |
| Fig. 22 Especificación Consejo .....            | 36 |
| Fig. 23 Especificación Premios .....            | 36 |
| Fig. 24 Especificación Ruleta .....             | 37 |
| Fig. 25 Especificación Recursos .....           | 37 |
| Fig. 26 Especificación Pista .....              | 38 |
| Fig. 27 Especificación Reserva .....            | 39 |
| Fig. 28 Especificación Arrastrador .....        | 39 |
| Fig. 29 Especificación SensorClick .....        | 40 |
| Fig. 30 Diseño test .....                       | 41 |
| Fig. 31 Diseño test lugar seleccionado .....    | 42 |
| Fig. 32 Base de datos .....                     | 43 |
| Fig. 33 Panel de resultados .....               | 43 |
| Fig. 34 Carta Gantt de pruebas .....            | 48 |
| Fig. 35 Carta gantt planificación inicial ..... | 56 |

---

## 1 INTRODUCCIÓN

---

### 1.1 Descripción de la problemática

Los países de América Latina han experimentado un retroceso significativo en los niveles educativos. En la actualidad, ocupan los últimos puestos de la lista del Informe PISA sobre Educación 2012, divulgado por la Organización para la Cooperación y el Desarrollo Económico (OCDE) (Pisa 2012 – <https://data.oecd.org/>). Una de las capacidades evaluadas más perjudicadas por estos informes, ha sido históricamente la matemática.

Una de las posibles causas que se pueden identificar, entre otras muchas, es la falta de motivación de los alumnos para enfrentarse a problemas abstractos, en una realidad cada vez más tecnológica, con alta carga audiovisual, de cambio constante y con un gran aumento de las fuentes de distracción extra escolares. Esta nueva realidad requiere realizar un cambio en las estrategias y en los procesos de enseñanza-aprendizaje. En particular, se requiere el diseño e implementación de alternativas que permitan reducir estas dificultades y medir su impacto en terreno, es decir, en el aula.

El presente proyecto de investigación tiene como objetivo trabajar en este importante reto. En particular, la meta es mejorar la motivación de los alumnos y por ende, los resultados obtenidos en la evaluación de la capacidad matemática. Con más precisión, se pretende ofrecer una alternativa científica y rigurosa mediante el diseño y la implementación de videojuegos educativos para el marco educativo chileno, focalizándonos en la enseñanza de las fracciones ya que es un tema de investigación actual (Moss and Case 1999).

### 1.2 Estado del arte

Existen varios informes que demuestran que los videojuegos pueden utilizarse de forma efectiva en el aula para mejorar el aprendizaje y la motivación en clase: (European Schoolnet 2009); (Futurelab 2007); (ISSA Press Release 2007); (BECTA 2006); (Teem 2002). Además de libros de textos donde se destaca la importancia de un cambio de paradigma en la educación: (Gee, J. P. 2008); (Prensky, M. 2006); (Shaffer, D. W. 2008). Además, se ha incrementado desde el año 2000 aproximadamente de material disponible publicado sobre el uso de los videojuegos. La presente revisión de la literatura trata de sintetizar los trabajos claves de la literatura actual formada por casi 200 artículos. Las principales categorías encontradas son: artículos sobre el potencial de los video juegos en la educación, artículos de investigación de video juegos diseñados para temas educativos específicos, como finanzas, idiomas modernos, etc; artículos que investigan los efectos de los video juegos en los individuos (por ejemplo, en su metabolismo y características físicas); investigación sobre cuestiones de género en relación al uso de los video juegos; estudios de la relación entre el uso de los juegos y el estilo de vida (incluyendo el uso doméstico); reseñas de investigación anteriores y meta-análisis (pocos de los cuales encontraron referencia a los estudios sobre el aprendizaje). Las siguientes cuestiones de investigación guían este estado del arte: ¿Cuál es el impacto del uso de los videojuegos en la juventud? ¿Por qué usar video juegos para enseñar y aprender? ¿Cómo se ha utilizado los videojuegos para aprender? ¿Cuáles son las

experiencias y preferencias de los alumnos con los videojuegos para aprender y para el ocio? ¿Cuáles son las recomendaciones para la planificación y el diseño de video juegos educativos?

En la década de 1970 los videojuegos se convirtieron en una de las principales opciones de entretenimiento para los jóvenes (es decir, los niños y adultos jóvenes). Hoy en día se ha extendido a todo tipo de plataformas, incluyendo los dispositivos móviles (Kirriemuir 2002). Esto hace accesible los juegos a las personas que no pueden poseer una televisión interactiva o un computador personal. Muchos jóvenes pasan una cantidad significativa de tiempo libre cada semana jugando a los videojuegos, a menudo 7 horas o más (Anderson y Bushman 2001; Walsh 2002).

Los videojuegos para muchos niños en su forma favorita de juego (VanDeventer y White JA 2002). Tanto alumnos como alumnas disfrutan con los videojuegos. Se ha comprobado que los alumnos juegan con más frecuencia a los videojuegos como un desafío y para impresionar a sus amigos. Las mujeres también juegan en menor medida, aunque también se han encontrado casos donde ellas requieren la aprobación de sus compañeros (Griffiths y Hunt 1995). Los hombres juegan con más regularidad (Griffiths y Hunt 1995; Griffiths, 1997b; Harris 1999;. Subrahmanyam et al 2001). Las mujeres no están motivadas por los juegos de violencia y prefieren juegos menos agresivos y menos exigentes (Griffiths, 1997b, página 234). Sin embargo, hay estudios (Cesarone 1998) que determinan tanto a niños como a niñas les gusta los videojuegos violentos en la misma proporción. Hay una relación causal entre el juego y el rendimiento académico (Emes 1997), encontrándose indicios de efectos positivos y negativos. Por un lado, los alumnos que más juegan suelen tener menos rendimiento (Colwell et al 1995;. Roe y Muijs 1998). Esto está relacionados con indicadores (Bruno 1995, página 101) que muestran que los alumnos menos aventajados muestran preferencias por actividades como ver la televisión o jugar a los video juegos. Jugar a menudo a los videojuegos reduce el tiempo disponible para hacer las tareas, lo que puede afectar negativamente al rendimiento académico (Roe y Muijs 1998). Se ha detectado los niños que tiene menos éxito académico pasan más tiempo jugando a los video juegos (3 horas o más) frente a los alumnos con un buen rendimiento (Benton 1995; Colwell et al., 1995). El uso excesivo de los videojuegos puede alentar una preferencia por la representación icónica, mientras que la gratificación instantánea puede disminuir la disposición a hacer el tipo de esfuerzo necesario para disfrutar de las recompensas de la lectura (Walsh 2002). En la sociedad actual, cada vez más gente necesita, aparte de las competencias básicas, entender y utilizar competencias matemáticas más avanzadas para razonar sobre cuestiones como la incertidumbre, el cambio, la evolución de los datos o las relaciones espaciales. Diversas aplicaciones tecnológicas han logrado que los estudiantes de secundaria entiendan mejor esos conceptos. Un ejemplo en el Proyecto (SimCalc Roschelle, J., Kaput, J., & Stroup, W. 2000) que ayuda a aprender conceptos de cálculo, como son los de porcentaje, acumulación, límite y valor medio.

Uno de los problemas de los videojuegos es que son de propósito general y no están adaptados a una realidad, no se suelen incorporar en el currículum, ni son diseñados en base a él. Proponer por tanto videojuegos para la enseñanza de las matemáticas que sean diseñados teniendo en cuenta las guías didácticas y la realidad del contexto es un problema abierto.

---

## 2 DEFINICIÓN PROYECTO

---

### 2.1 Objetivos del proyecto

Objetivo general:

Estudiar el impacto de actividades de aprendizaje de matemáticas complementadas con videojuegos, mediante al diseño e implementación de un videojuego para estudiantes de tercero y cuarto básico de establecimientos de la ciudad de Concepción

Objetivos específicos:

- Estudiar las actividades de aprendizaje de los diseños didácticos de los profesores.
- Caracterizar a los alumnos y el tipo de videojuegos.
- Diseñar e implementar un videojuego a partir de la planificación de los docentes para complementar las actividades de aprendizaje. Las planificaciones y las actividades se convierten en requisitos funcionales de los videojuegos.
- Evidenciar el impacto de estas actividades en el rendimiento de los estudiantes para el aprendizaje de las actividades fijadas en la planificación docente.
- Analizar el impacto de estas actividades en la motivación de los estudiantes para el aprendizaje de las actividades fijadas en la planificación docente.
- Controlar el impacto de estas actividades en el desarrollo del pensamiento lógico.

### 2.2 Ambiente de Ingeniería de Software

Se seguirá una metodología de investigación mixta que permita obtener mayor integración entre los métodos cuantitativos y los cualitativos desde el prisma educacional y desde el prisma del diseño. La metodología constará de cuatro fases i) estudiar textos MINEDUC, unidades didácticas en fracciones, hablar con el experto (Jairo Navarrete, docente de la Universidad del Bío-Bío), caracterización de estudiantes; ii) diseño e implementación de videojuegos basados en el conocimiento adquirido en las fases anteriores, las actividades de aprendizaje estudiadas, la caracterización de los estudiantes realizada, las entrevistas con el experto y los textos de MINEDUC; iii) análisis y evaluación del software desde el punto de vista de la ingeniería del software; iv) Análisis, monitorización y evaluación de los videojuegos desde el punto de vista didáctico y la cultura chilena.

Para el desarrollo del juego se seguirá la metodología incremental. Para esto se hará un diseño general del juego enfocándose en las mecánicas de juego y la estética. Posteriormente en la fase 2, que corresponde el desarrollo propiamente, tal se irán entregando incrementos funcionales, aproximadamente cada dos semanas para su retroalimentación con los profesores Jairo y Clemente, los cuales irán dictando si hay que arreglar algo o si hay algún cambio de planes con respecto a la planificación inicial.

Estándar de documentación:

IEEE Std 830 1998(Software Engineering Standards Comitee of the IEEE Computer Society, 1998) Para cumplir con el formato de la plantilla del proyecto de título de desarrollo de software de la carrera.

Tecnología de desarrollo:

- **Luxe Engine 1.0.0**, motor multiplataforma para el desarrollo de videojuegos.
- **Haxe 3.4.1**, lenguaje en el cual está escrito Luxe.
- **Firebase**, plataforma que ofrece una base de datos que se puede conectar con Luxe.

Software de desarrollo:

- **Sublime 3**, editor de texto.
- **Adobe Photoshop CC 2017.0.0**, para la edición y creación de imágenes

Software de apoyo:

- **Windows 7**, sistema operativo.
- **Firefox 56.0.1**, navegador web, para ejecutar el código.
- **Git 2.6.3**, sistema de control de versiones.
- **Draw.io**, para dibujo de casos de uso.
- **Tomsplanner**, para crear carta Gantt.
- **StarUml**, para desarrollo de diagrama de clases.

## 2.3 Definiciones, Siglas y Abreviaciones

|     |                                   |
|-----|-----------------------------------|
| CU  | Caso de uso.                      |
| POO | Programación orientada a objetos. |
| CPU | Caso de prueba unitaria.          |

---

### 3 DEFINICION DE LA EMPRESA O INSTITUCIÓN

---

#### 3.1 Descripción de la empresa

Antecedentes generales de la Empresa

- Nombre: Universidad del Bío-Bío.
- Rut: 60.911.006-6
- Representante legal: Héctor Guillermo Gaete Feres
- Dirección: Av. Casilla 5-C, Collao 1202, Concepción, Región del Bío Bío, Chile
- Rubro: Educacion

Entorno

En la región, los principales centros académicos del área de enseñanza de informática que compiten con la Universidad son la Universidad de Concepción, Universidad Técnica Federico Santa María, Universidad Católica de la Santísima Concepción, la Universidad del desarrollo, la Universidad san Sebastián, el instituto profesional DuocUC, el instituto profesional Virginio Gomez, entre otros.

#### Misión

Difundir el conocimiento científico, técnico y artístico generado en la Universidad, mediante el desarrollo de programas de extensión relevante, proyectos de extensión universitaria y presentaciones de sus grupos artísticos universitarios, contribuyendo al mejoramiento de la calidad de vida de cada habitante de la Región y a la difusión de múltiples expresiones culturales.

#### Visión

La visión de la Dirección de Extensión es la de consolidar la presencia de la Universidad del Bío-Bío como agente transformador y partícipe del desarrollo social en los diferentes sectores de la comunidad, logrando que el conocimiento generado en la academia, así como la actividad artística, cultural y recreativa, sea socialmente útil, y contribuya al desarrollo del país.

## Organigrama

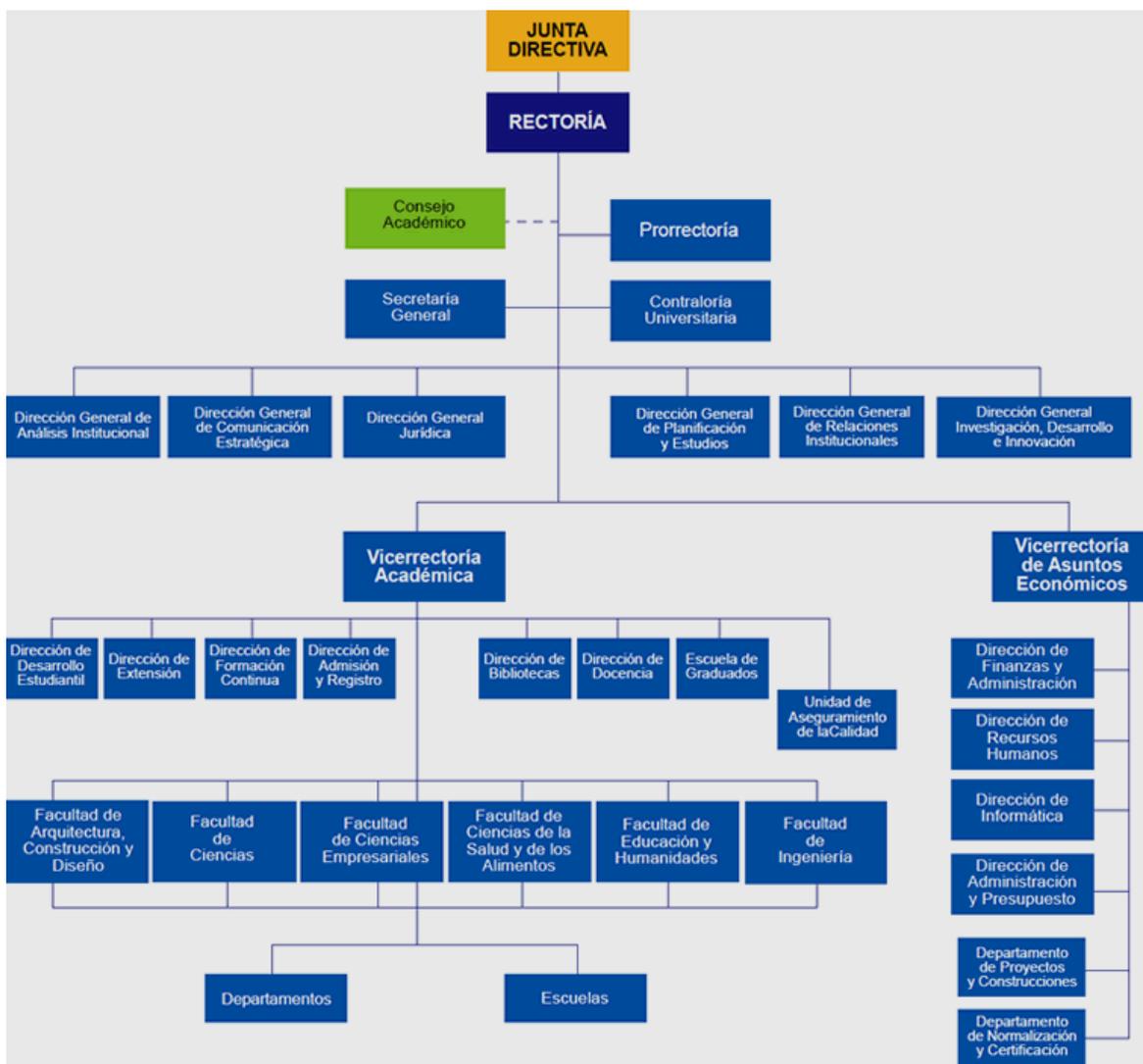


Fig. 1 Organigrama UBB

### 3.2 Descripción del área de estudio

El estudio se realizará en el departamento de Sistemas de Información de la facultad de Ciencias Empresariales.

El cliente y profesor guía del proyecto es Clemente Rubio, académico, ingeniero superior informático, magister en tecnologías informáticas avanzadas y doctor en informática

### Misión de la facultad

Somos una facultad comprometida en la formación continua de profesionales integrales de pre y postgrado; en la generación y difusión del conocimiento, impulsando el emprendimiento y la innovación para el desarrollo regional y nacional.

### Visión de la facultad

Ser una Facultad socialmente responsable, reconocida a nivel nacional por su excelencia académica con vinculación internacional en las áreas de las Ciencias Empresariales, de la Computación e Informática.

### Organigrama

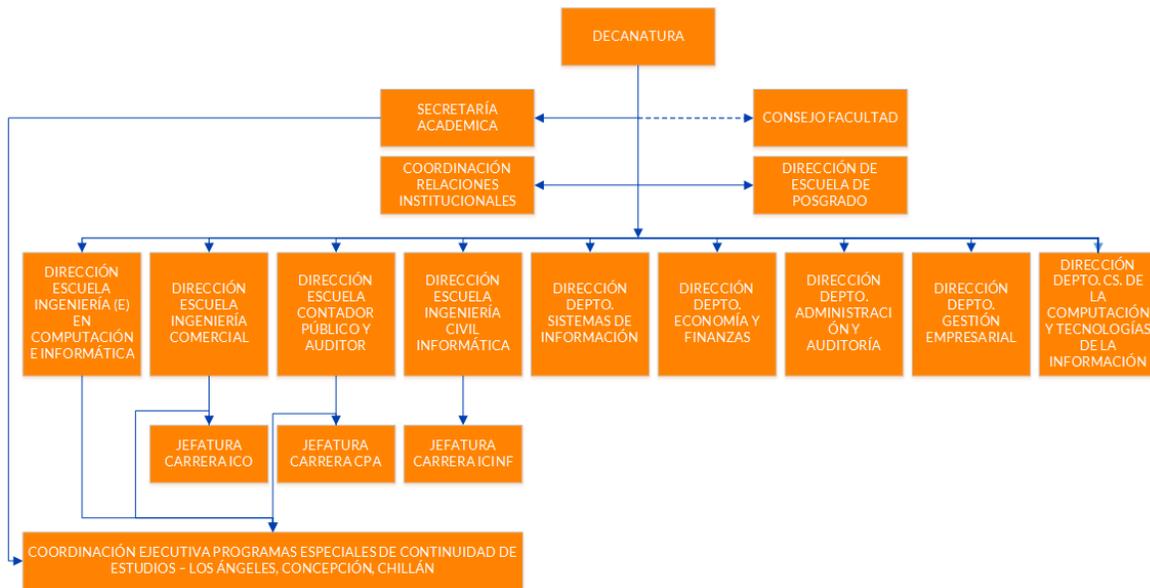


Fig. 2 Organigrama FACE

---

## 4 BASE PEDAGÓGICA

---

### 4.1 Contexto

Con el fin de ubicar un nivel de enseñanza donde el juego provoque el efecto deseado en el aprendizaje de fracciones, se realizó un análisis de los planes y programas del ministerio de educación centrando en el contenido específico de fracciones desde sus comienzos en la enseñanza básica.

Según los Planes y Programas del Ministerio de Educación (2012) el concepto de fracción aparece por primera vez a nivel de tercer año básico, donde se pide entre otros objetivos que los estudiantes comprendan las fracciones de uso común como  $1/4$ ,  $1/3$ ,  $1/2$ , explicando que una fracción representa la parte de un todo de manera concreta, pictórica y simbólica y comparando fracciones de un mismo denominador utilizando material concreto.

Luego a nivel de cuarto año básico estos objetivos se amplían dando paso a la comprensión de fracciones con denominadores mayores, tales como: 16, 12, 10 y 8, explicando que una fracción representa la parte de un todo o de un grupo de elementos y un lugar en la recta numérica, así como también comparando y ordenando fracciones con material concreto y pictórico. Así mismo se introduce la resolución de adiciones y sustracciones de fracciones con igual denominador de manera concreta y pictórica en el contexto de la resolución de problemas.

En los siguientes niveles escolares se sigue desarrollando el concepto de fracción basándose en los contenidos enseñados en los primeros años pero aumentando su dificultad y de este modo extendiendo el conjunto numérico a los racionales.

Apoyados en el análisis realizado y la teoría de respaldo se puede apreciar que generalmente la fracción se enseña mediante la metáfora parte de un entero, lo cual genera que los alumnos pierdan el sentido numérico de ella. Los autores Chaffe-Stengel y Nodding (1982), creen que esta manera de abordar el concepto de fracción en la enseñanza escolar es guiado por un modelo conceptual parte-todo, surgiendo el concepto de fracción como parte de cosas que no son números, perdiendo así el sentido numérico. Es aquí donde surge un obstáculo para los niños, debido a que es difícil visualizar operaciones con fracciones de distinto denominador cuando la metáfora fracción como parte no ofrece los elementos necesarios para esto, lo que conlleva a que los niños cometan errores derivados de esta metáfora de enseñanza.

A partir de lo anterior surge la necesidad de dar un nuevo sentido a la enseñanza de las fracciones, donde los estudiantes sean capaces de interpretarlas en diversos contextos logrando de esta manera sortear los obstáculos que se generan con la noción parte de un todo. Una forma de enfrentar este obstáculo desde sus inicios en enseñanza básica es enseñar las fracciones a través de la recta numérica facilitando así que los estudiantes conserven las propiedades numéricas de los números racionales.

Con el fin de dar uso a la recta numérica y enseñar las propiedades de las fracciones a través de relaciones representadas en una situación más sencilla sin uso de lenguaje

matemático riguroso, se hace uso de una analogía que permitirá a los estudiantes establecer una conexión entre la situación planteada y lo que deseamos enseñar.

## 4.2 Diagramas analógicos

### 4.2.1 Comparación

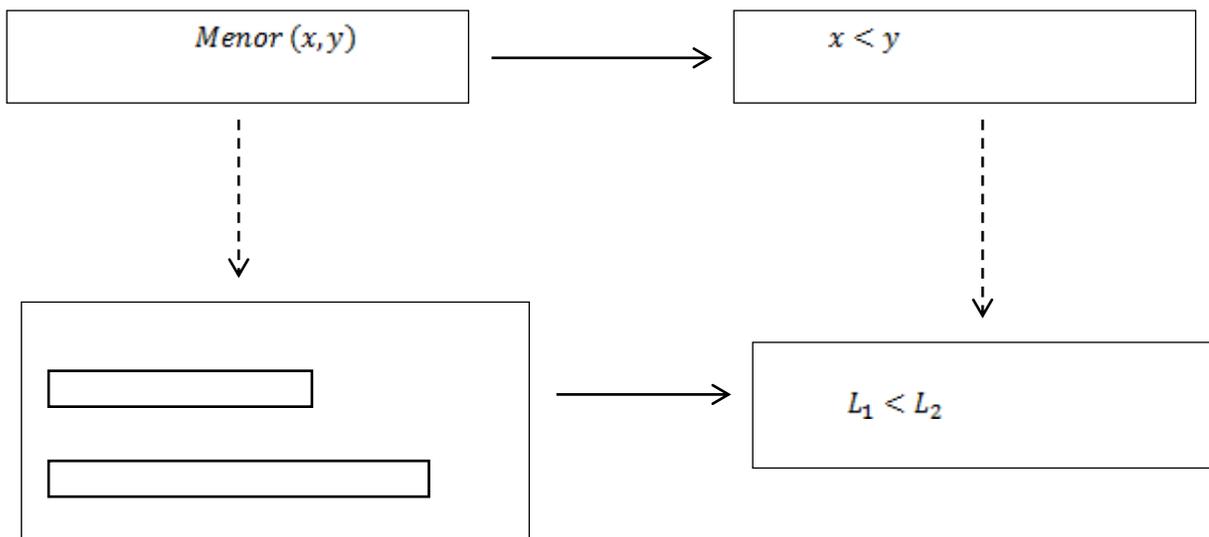


Fig. 3 Analogía: comparación

### 4.2.2 Equivalencia

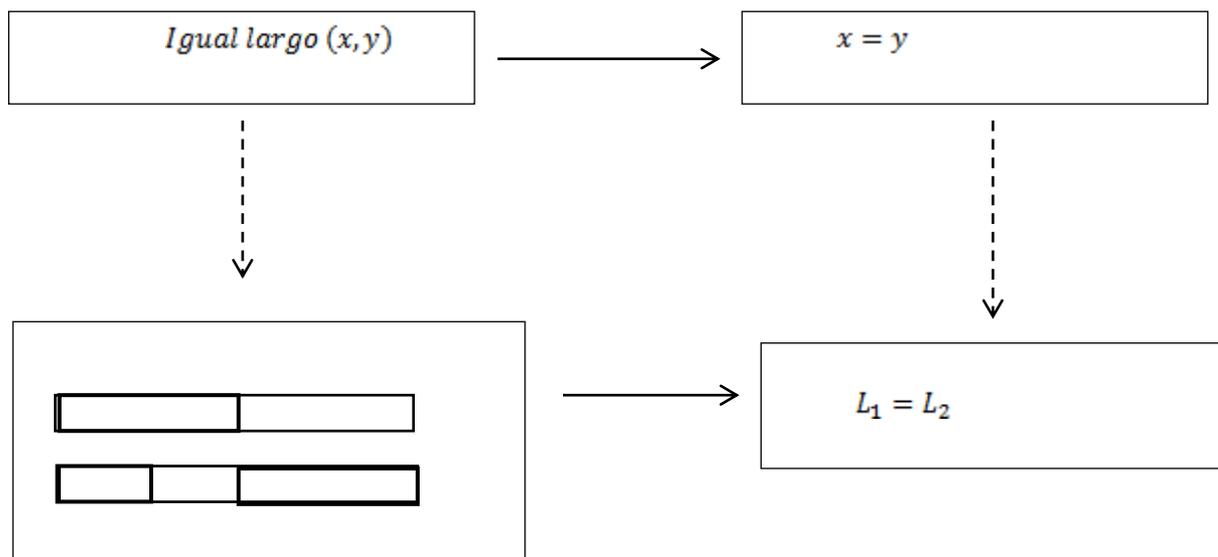


Fig. 4 Analogía: equivalencia

### 4.2.3 Suma

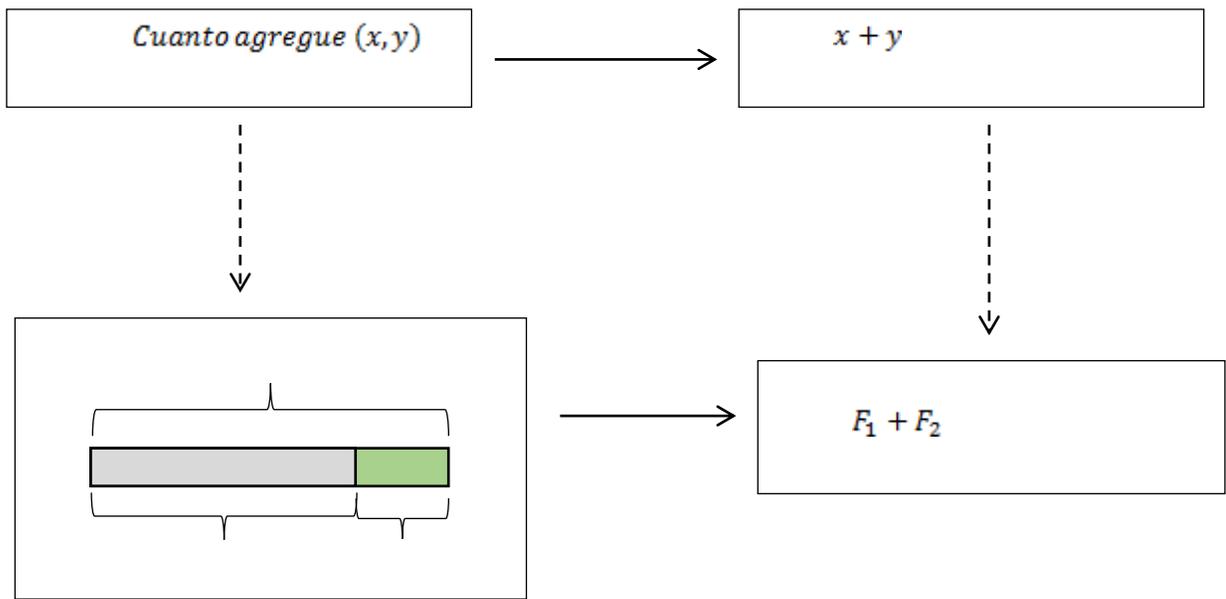


Fig. 5 Analogía: suma

### 4.2.4 Resta

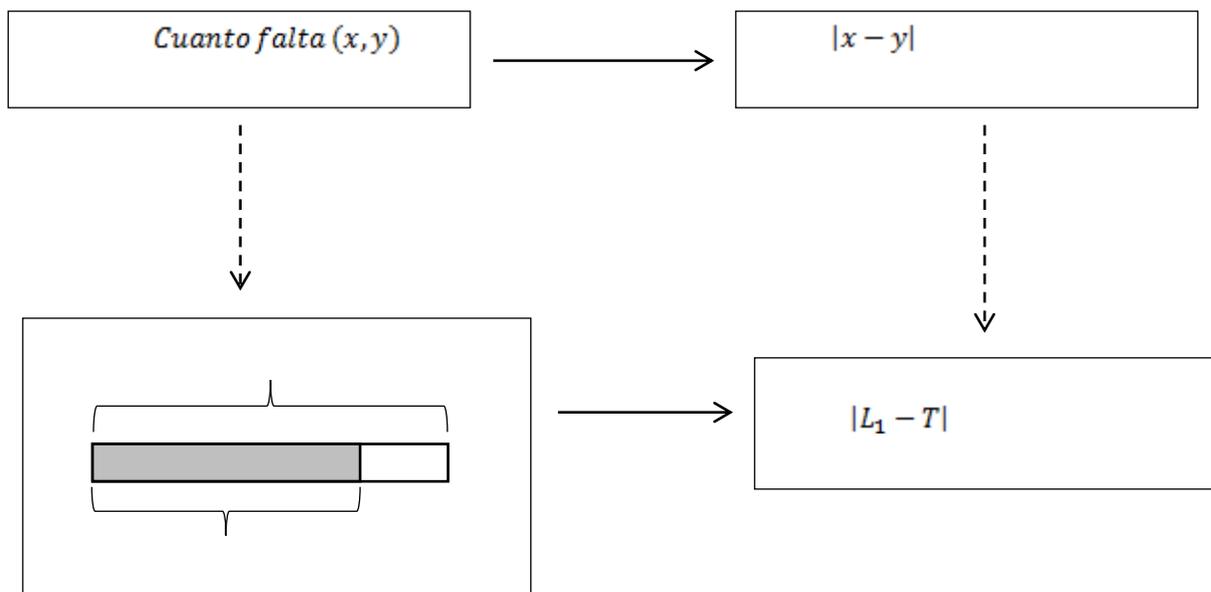


Fig. 6 Analogía: resta

---

## 5 ESPECIFICACIÓN DE SOFTWARE

---

### 5.1 Alcances

Se pretende desarrollar un videojuego que permita que el jugador se familiarice con las fracciones. Esto se lleva a cabo con el apoyo de una analogía que permita representar la recta geométrica, que es un concepto matemático abstracto, a un concepto que resulte más familiar: la construcción de un edificio que se va completando con fichas de diferentes tamaños, cada una con su respectivo valor fraccional, haciendo uso de las analogías mencionadas en el capítulo anterior.

Sobre esta base pedagógica se diseñó el juego de manera tal que dos jugadores compitan por completar el edificio primero, recompensándolos por reemplazar fichas pequeñas en fichas más grandes de igual valor. Con esta mecánica los jugadores intuitivamente sumaran las fichas más pequeñas e identificarán la equivalencia entre fracciones pequeñas y una grande.

El juego estará alojado en un dominio web público y será desarrollado en el motor para videojuegos Luxe. Ambos jugadores deberán jugarlo en el mismo navegador, no habrá una conexión remota entre ellos. Está pensado para ser un complemento al método de enseñanza convencional y de ningún modo pretende reemplazarlo.

### 5.2 Objetivo del software

Objetivo general:

Apoyar el proceso educativo de estudiantes de 3° y 4° básico en el ámbito de la comprensión didáctica de fracciones, mediante al uso de analogías para poder establecer su impacto educacional.

Objetivos específicos:

- Ejercitar equivalencias entre fracciones mediante analogías basadas en un contexto amigable y fácil de entender
- Ofrecer herramientas para que el juego se pueda explicar a sí mismo de manera simple, esto con la finalidad de que el jugador pase menos tiempo aprendiendo las reglas del juego y se enfoque en la parte matemática.

---

## 6 ESTUDIO DE FACTIBILIDAD

---

### 6.1 Factibilidad técnica.

El motor para el desarrollo será Luxe, está escrito en Haxe. Cuando se compila un archivo escrito en Haxe se genera una aplicación nativa lista para ser ejecutada en la plataforma que se elija. En este caso se cuenta con servicio de hosting para alojar la aplicación, por lo que el código generado por Haxe será JavaScript.

Los entornos de desarrollo y de producción son los siguientes:

Entorno de desarrollo:

- Windows 7
- Luxe Engine 1.0.0
- Haxe 3.4.1
- Sublime 3
- Adobe Photoshop CC 2017.0.0
- Firefox 56.0.1
- Git 2.6.3

Entorno de producción:

- GNU/Linux
- Apache2

### 6.2 Factibilidad operativa.

El juego está pensado para funcionar con mecánicas simples para que los jugadores se enfoquen más en el área didáctica que entendiendo las reglas. De todas formas el juego es auto explicativo, pues tiene una explicación breve en donde enseñan todas las mecánicas y continuamente ofrece consejos para guiar al jugador.

### 6.3 Factibilidad económica.

Todo el software anteriormente descrito es de uso libre o se poseía con anterioridad, y el host también estaba contratado previamente. No existe ningún costo por el desarrollo de este software considerando a la mano de obra un ahorro.

De momento, este videojuego tampoco tiene ningún propósito comercial, ya que está construido únicamente para poder llevar a cabo esta investigación.

#### **6.4 Conclusión de la factibilidad**

En vista y considerando que el videojuego está desarrollado sobre tecnologías gratuitas o que ya se habían adquirido con anterioridad, no hay ningún impedimento mayor para el desarrollo de este proyecto.

---

## 7 DISEÑO

---

### 7.1 Descripción global del videojuego

A continuación se hará una descripción del videojuego a desarrollar según los siguientes criterios:

**Género:** Tomando en cuenta su propósito se puede definir como un videojuego educativo, pues posee un carácter lúdico y didáctico. Si en cambio se analiza según sus mecánicas, casi se puede considerar un juego de puzles, con la diferencia de que se juega compitiendo con otro jugador, por quien lo completa primero.

**Jugadores:** Se requieren 2 personas.

**Escenario:** El escenario consta de dos edificaciones en construcción, teniendo dos pisos cada una. El estilo es minimalista para no saturar a los jugadores de decoraciones, puesto a que el foco es hacer que los jugadores se concentren en las fracciones y en competir.

**Interfaz de usuario:** Los jugadores interactúan con el videojuego mediante el mouse. Existe un área en la pantalla para cada jugador de donde pueden sacar sus diferentes tipos de fichas y colocarlas en el edificio. Hay fichas de un octavo, un cuarto, un medio y un entero. También hay una recta numérica en cada edificio y 4 premios para cada jugador delimitados con un martillo.

**Objetivos:** El objetivo es colocar las dos fichas de un entero en la construcción antes que el oponente.

**Reglas:** En cada turno los jugadores podrán colocar una cantidad limitada de fichas de un octavo (entre uno y tres aleatoriamente). Para colocar fichas más grandes se debe reemplazar fichas más pequeñas que ya estén colocadas, colocando las grandes arriba de las pequeñas. Siempre que las equivalencias sean correctas, se pueden hacer todos los reemplazos que el usuario desee. La cantidad de fichas es limitada, cuando se realiza un reemplazo se recuperan las fichas reemplazadas. Estando en una zona premiada se puede hacer un reemplazo y se obtiene un turno extra.

**Diseño de niveles:** Los edificios serán siempre de dos pisos y siempre se tendrá la misma cantidad de fichas. Aleatoriamente se eligen las zonas premiadas y a qué jugador le toca primero.

**Requerimientos técnicos:** Para ejecutar este juego solo se necesita un navegador web e internet.

## 7.2 Mecánicas

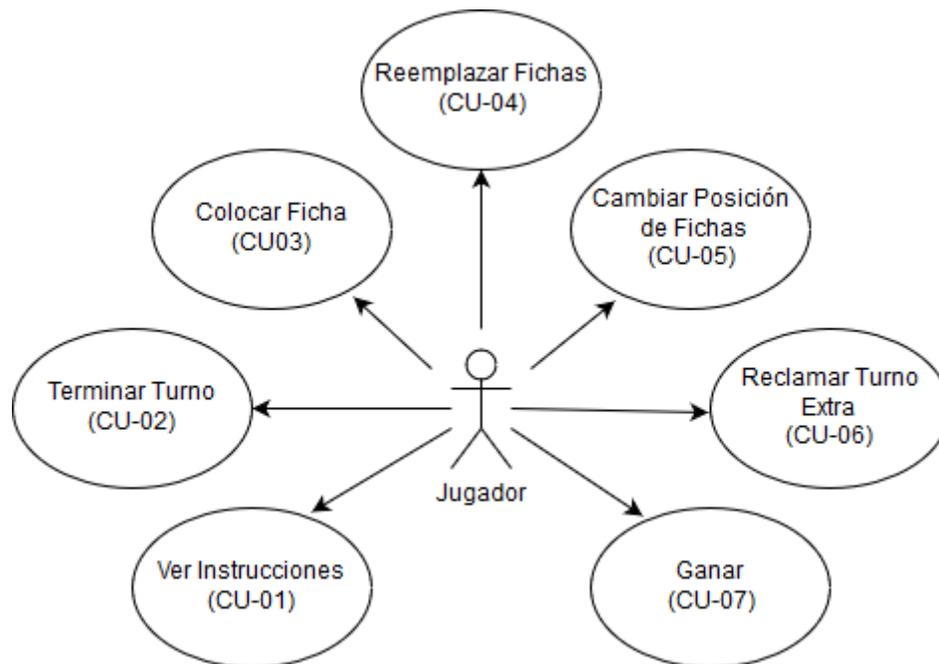
A continuación se detallarán las diferentes mecánicas del juego en el formato de casos de uso.

### 7.2.1 Actores

Jugador:

Se trata de los individuos que interactúan con el juego. Si bien se precisan dos jugadores para llevar a cabo el juego, en el diagrama solo existe un actor denominado “Jugador” que los representa a ambos, pues se tratan de los mismos casos de uso.

### 7.2.2 Casos de Uso y descripción



**Fig. 7 Casos de uso**

Los jugadores deben ingresar a la página <http://youractionsdefineyou.com/fracciones/> y decidir quién será el jugador de la izquierda y el de la derecha, hecho esto se puede comenzar a jugar y se llevarán a cabo los casos de uso.

### 7.2.3 Especificación de los Casos de Uso

#### 7.2.3.1 Caso de Uso: CU-01 Ver Instrucciones

- Descripción: Muestra al jugador una explicación del objetivo del juego, las mecánicas y como reclamar las recompensas.
- Pre-Condiciones: Las instrucciones se pueden ver en todo momento una vez comenzado el juego.

- Flujo de Eventos Básicos:

| Al actor  | El juego  |
|---|---|
| 1. El jugador pulsa el botón identificado como “Instrucciones”.       | 2. El juego muestra el objetivo del juego, las mecánicas y como reclamar las recompensas. Ofrece un botón para regresar al juego. |
| 3. Cuando el jugador lo desee, pulsa el botón para regresar al juego. | 4. El juego oculta las instrucciones y regresa al estado de juego.  |

- Post-Condiciones: El juego regresa al mismo estado que se encontraba antes del transcurso de este caso de uso.

### 7.2.3.2 Caso de Uso: CU-02 Terminar Turno

- Descripción: El jugador da a conocer que ya acabo su jugada y da pie a que le toque el otro jugador.
- Pre-Condiciones: Se puede dar solo después de que ya se asignó la cantidad de fichas que le tocó al jugador. Si en este turno el jugador obtuvo un turno extra, se reclamará un turno extra (CU-06) en lugar de llevarse a cabo este caso de uso.
- Flujo de Eventos Básicos:

| Al actor                                     | El juego   |
|--|--|
| 1. El jugador pulsa el botón terminar turno. | 2. La flecha de indicador de turno indica al otro jugador y empezará el sorteo del número de fichas que le tocará. |

- Post-Condiciones: El otro jugador podrá hacer sus jugadas dentro de las reglas.

### 7.2.3.3 Caso de Uso: CU-03 Colocar Ficha

- Descripción: El jugador coloca fichas de 1/8 en el edificio. Puede ser de sus recursos o de la reserva.
- Pre-Condiciones: Debe ser el turno del jugador. Para sacar una ficha de la reserva debe tener reservada alguna. Para sacar fichas de los recursos debe tener alguna en los recursos y debe tener fichas restantes asignadas para su turno.
- Flujo de Eventos Básicos:

| Al actor  | El juego                                   |
|---|--|
| 1. El jugador arrastra una ficha de 1/8 desde los recursos o de las reservas a su edificio. | 2. Se añade la ficha que tomó al edificio. |

- Flujo de Eventos Alternativo:

| Al actor  | El juego  |
|---|---|
| 1(a) El jugador intenta arrastrar una ficha de 1/8 sin tener fichas restantes este turno, o el jugador intenta colocar una ficha que no sea 1/8, o el edificio ya está lleno de fichas. | 3. Al momento de soltar la ficha, en lugar de añadirse al edificio, regresa al lugar desde donde se arrastró. |
| 1(b) El jugador arrastra una ficha de   | 4. Se añade la ficha que tomó a la  |

|                                      |          |
|--------------------------------------|----------|
| 1/8, desde los recursos a la reserva | reserva. |
|--------------------------------------|----------|

- Post-Condiciones: Se resta la ficha de donde estaba y se añade al destino. Si el origen era los recursos, se resta una ficha restante de este turno.

#### 7.2.3.4 Caso de Uso: CU-04 Reemplazar Fichas

- Descripción: El jugador hace equivalencias con una ficha de su reserva a fichas más pequeñas que en conjunto son equivalentes a la grande.
- Pre-Condiciones: Deben haber fichas en el edificio, deben haber fichas en los recursos, debe ser el turno del jugador y la equivalencia debe ser matemáticamente correcta.
- Flujo de Eventos Básicos:

| Al actor   | El juego   |
|--|--|
| 1. El jugador arrastra una ficha de los recursos sobre un grupo de fichas puestas en el edificio | 2. La ficha que se arrastró reemplaza a las que estaban puestas, dejando la grande en el edificio y las que fueron reemplazadas se recuperan, regresando a los recursos. |

- Flujo de Eventos Alternativo:

| Al actor   | El juego  |
|--|---|
| 1(a) El jugador intenta hacer una equivalencia matemáticamente incorrecta. | 3. Al soltar la ficha arrastrada, en vez de hacer el reemplazo, regresa a los recursos. |

- Post-Condiciones: El jugador recupera las fichas reemplazadas. Este proceso no quita las fichas restantes asignadas al turno.

#### 7.2.3.5 Caso de Uso: CU-05 Cambiar Posición de fichas

- Descripción: Se cambian las posiciones de las fichas que fueron puestas en su edificio.
- Pre-Condiciones: Deben haber al menos dos fichas colocadas en el edificio y debe ser el turno del jugador.
- Flujo de Eventos Básicos:

| Al actor   | El juego                                       |
|--|--|
| 1. El jugador hace clic sobre una ficha del edificio y luego sobre otra. | 2. Las fichas cambian de posición una con otra |

- Post-Condiciones: El juego solo cambia la posición de las fichas que fueron elegidas.

#### 7.2.3.6 Caso de Uso: CU-06 Reclamar Turno Extra

- Descripción: El jugador obtiene y reclama un turno extra.
- Pre-Condiciones: El jugador debe estar en una zona premiada.
- Flujo de Eventos Básicos:

| Al actor | El juego |
|----------|----------|
|----------|----------|

|  |   |
|--|---|
| <p>1. El jugador realiza un reemplazo válido.</p>  | <p>2. El juego le avisa al jugador que, si quiere, puede reclamar un turno extra. El botón “terminar turno” se transforma en “turno extra”.</p> |
| <p>3. Antes de reclamar el turno, el jugador puede dejar fichas desde los recursos en la reservas para ocuparlas luego, de manera que no desperdicie las fichas restantes del turno. El jugador reclama el turno extra, pulsando el botón “turno extra”.</p> | <p>4. El juego sortea nuevamente la cantidad de fichas restantes que el jugador posee este turno. Pierde las que poseía anteriormente.</p>      |

- Post-Condiciones: Corresponde de nuevo el turno al jugador.

#### 7.2.3.7 Caso de Uso: CU-07 Ganar

- Descripción: La partida finaliza, estableciendo como vencedor a uno de los jugadores.
- Pre-Condiciones: Debe ser el turno del jugador y el edificio debe estar lleno de fichas.
- Flujo de Eventos Básicos:

| Al actor   | El juego   |
|--|--|
| <p>1. El jugador debe situar las dos fichas de un entero en el edificio.</p> | <p>2. El juego establece como ganador al jugador, cambiando las fichas de ladrillos por la imagen de un edificio construido.</p> |

- Post-Condiciones: El juego termina, se ofrece la opción de empezar otra vez.

### 7.3 Diseño de arquitectura funcional

La arquitectura de Luxe es la siguiente:



**Fig. 8 Arquitectura de Luxe**

Como se puede observar, Luxe se apoya en tres elementos:

- Haxe: es un lenguaje de programación que genera código nativo en la plataforma que se desee.
- Flow: toma el código de Haxe y le da un contexto para poder ser usado rápidamente, por ejemplo, si se compila para web, flow crea un html en el cual esta embebido el código. También administra los assets y las librerías externas.
- Snow: Snow a su vez utiliza Flow, se trata de una herramienta que capta la entrada de teclado, mouse, audio, gamepad, etc... Luxe lo utiliza para la entrada y salida de datos.

Luxe también cuenta con el apoyo de una comunidad Snowkit, que se dedica a desarrollar librerías y frameworks para Haxe. Todas las tecnologías anteriormente descritas son libres y de código abierto.

## 7.4 Diseño interfaz y navegación

El videojuego está dividido en tres secciones:

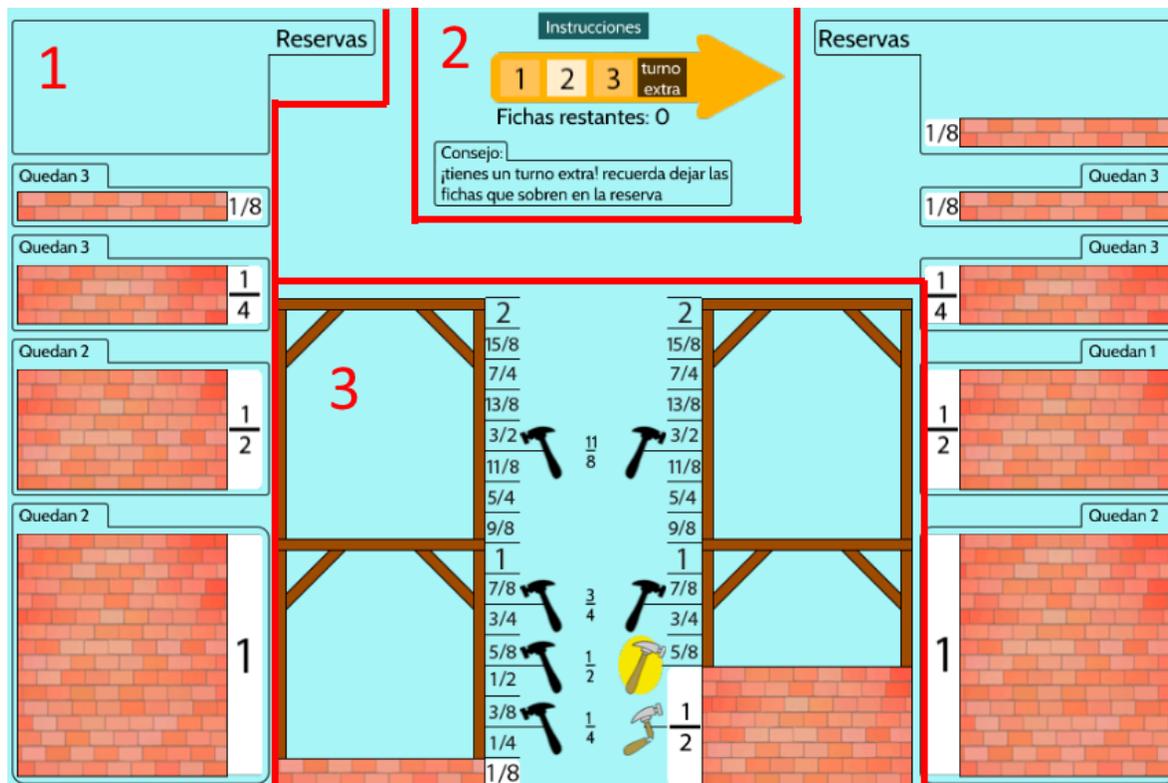


Fig. 9 Diseño general

### 7.4.1 Explicación de la sección 1

La sección 1 muestra los recursos que cada jugador tiene a su disposición.

Para cada tipo de ficha se muestra la cantidad y la imagen.

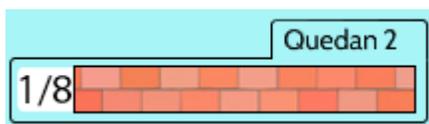


Fig. 10 Ejemplo fichas existentes

Si las fichas se acaban, la imagen no se muestra.

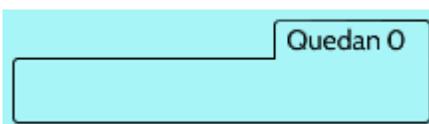
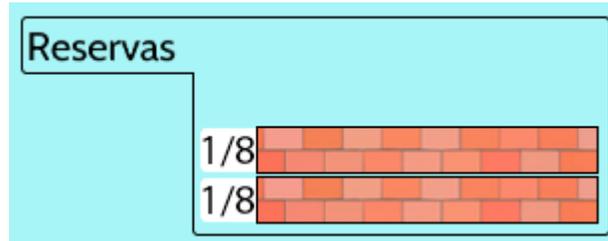


Fig. 11 Ejemplo fichas agotadas

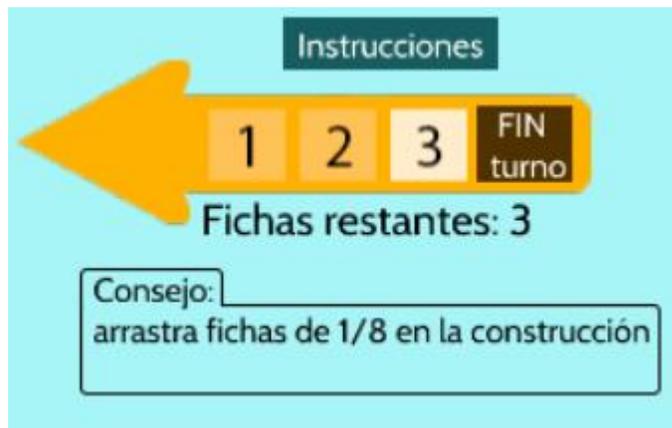
También muestra la cantidad de fichas en la reserva una sobre otra.



**Fig. 12 Ejemplo de reservas**

#### 7.4.2 Explicación de la sección 2

La sección 2 está orientada a mostrar información acerca del contexto del juego de manera simple.



**Fig. 13 Sección 2**

Posee un consejo, que cambia según el contexto del juego, no es necesario seguirlo pero provee una guía por si el jugador está indeciso o confundido.

También se encuentra la información relacionada con el turno, el indicador de turno (la flecha) seleccionará como una ruleta cuantas fichas se le asignarán al jugador.

Por último está el botón para ver las instrucciones, al presionarla los jugadores verán lo siguiente:

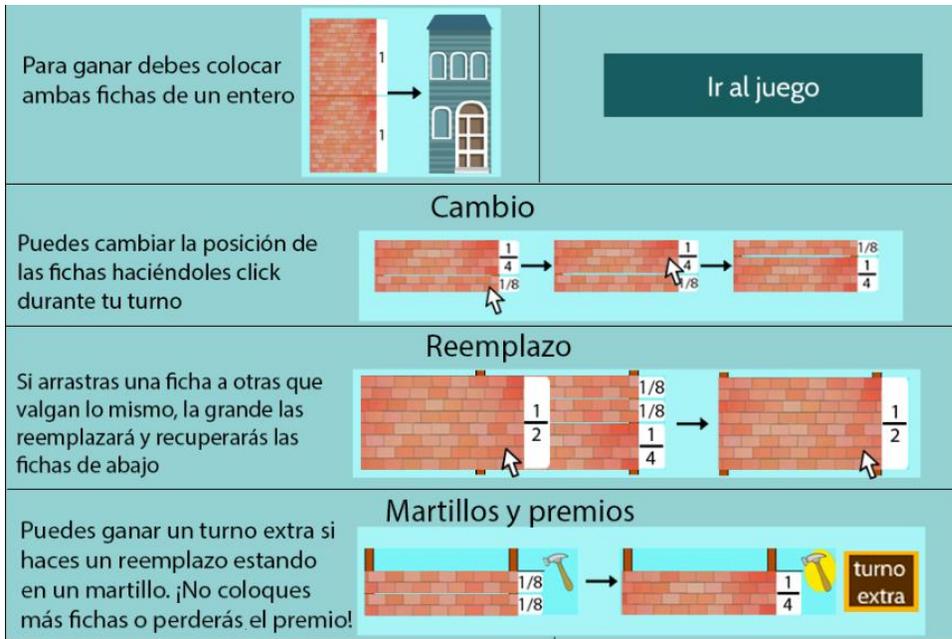


Fig. 14 Instrucciones

### 7.4.3 Explicación de la sección 3

Finalmente la sección 3 indica el estado del juego y las condiciones de victoria, muestra donde están las zonas premiadas, la cantidad y el tipo de fichas que están colocadas y una recta geométrica expresada en fracciones.

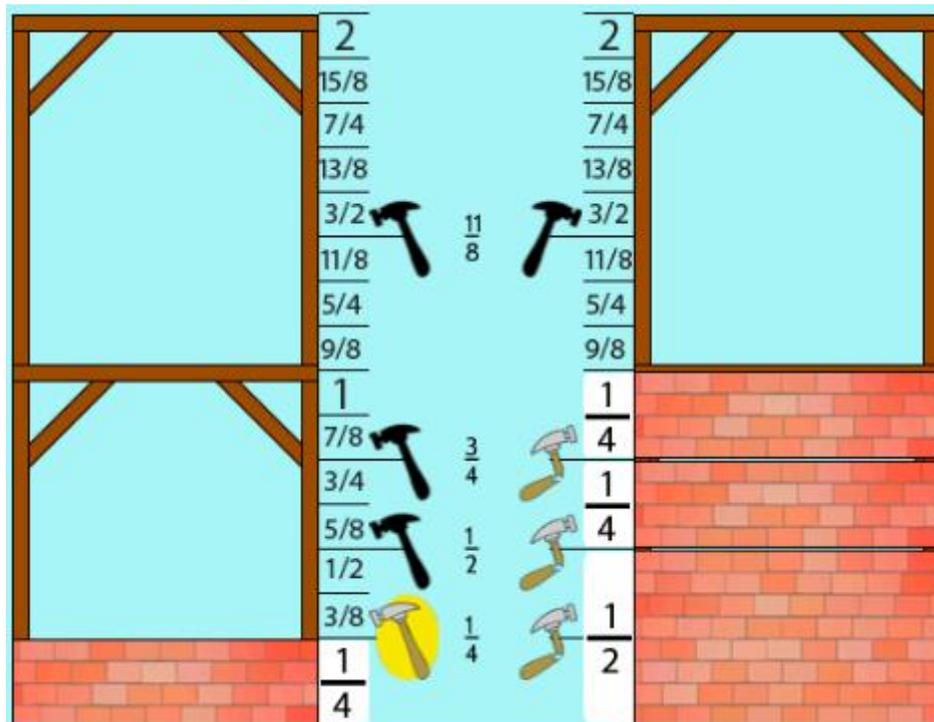


Fig. 15 Sección 3

Los martillos cambian su imagen dependiendo de qué tan cerca esté el jugador de ganarse el premio. Cuando solo se ve la silueta aún no se ha llegado al premio, si se ve el martillo normal se llegó pero aun no desbloquea el premio, cuando se desbloquea el martillo se ilumina, indicando que se puede reclamar un turno extra, y una vez reclamado el martillo se rompe. Estos son las imágenes de los martillos en orden.

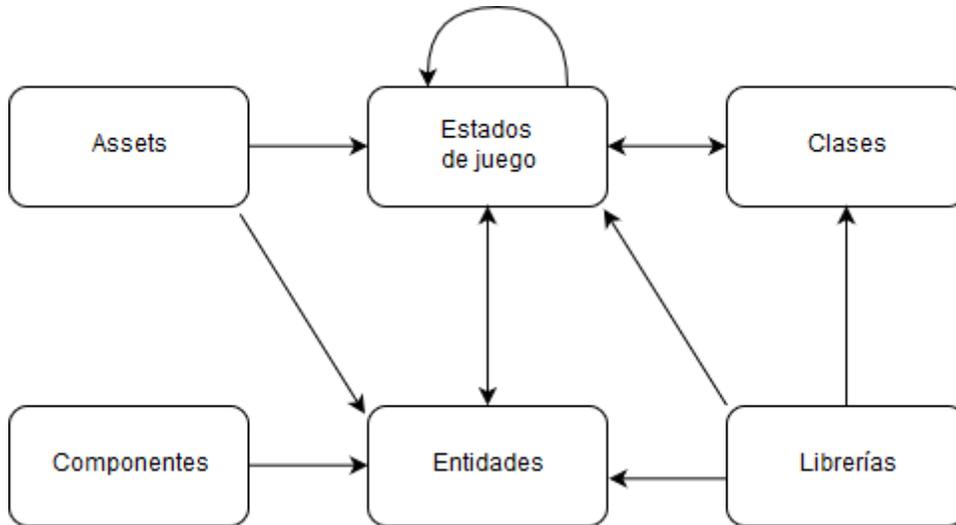


**Fig. 16 Martillos**

## 8 IMPLEMENTACIÓN

### 8.1 Patrón de diseño

El videojuego se construyó sobre un template de Luxe que hace uso del siguiente patrón de diseño:



**Fig. 17 Patrón de diseño**

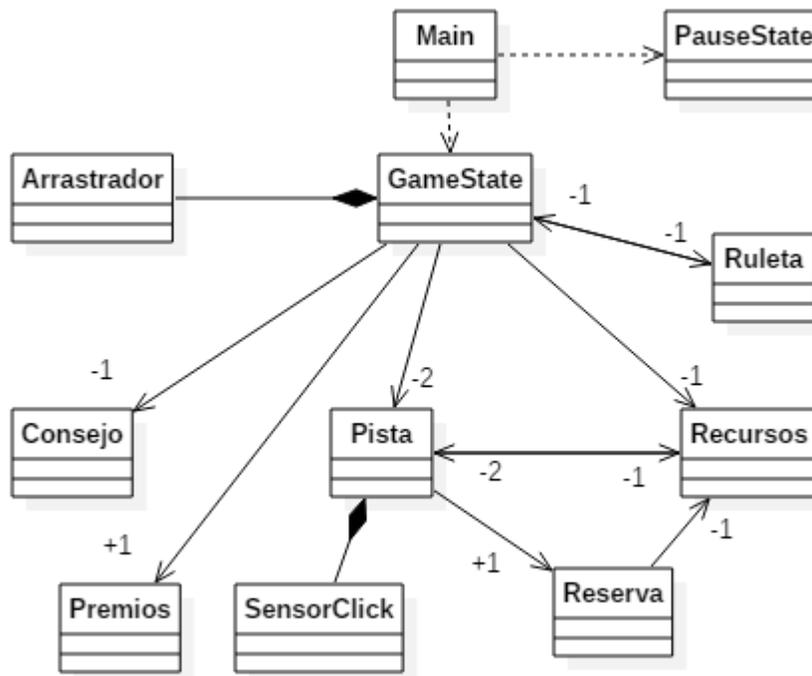
El motor primero carga los assets, que son los recursos gráficos y luego ordena cual es el estado de juego que comienza con la ejecución del juego.

Los estados de juego corresponden a las diferentes fases que el videojuego puede entrar, en este proyecto corresponden a la introducción, el menú, las instrucciones y el videojuego en sí. Cada estado provee un contexto en el cual los elementos van tomando vida, dentro de los estados se definen las entidades y la lógica.

Las entidades corresponden a los elementos que aparecen visualmente en cada estado que posean alguna propiedad lógica, posición y tamaño. Se les puede asignar componentes, que es una encapsulación de propiedades lógicas, por ejemplo si existen varias entidades que necesitan ser arrastrada, convendría crear un componente con esa propiedad y añadirlo a las entidades en cuestión.

Y por último existen las clases y librerías como es usual en otros lenguajes. Las clases son elementos conceptuales utilizados en la programación orientada a objetos y las librerías son herramientas que facilitan el desarrollo.

## 8.2 Diagrama de clases



**Fig. 18 Diagrama de Clases**

A partir de la clase Main se generan los estados de juego según lo desee el usuario. La clase principal sobre la que se trabajó es GameState, sobre esta se construyó toda la lógica requerida para este videojuego. Arrastrador y SensorClick son componentes, se utilizan para añadir lógica de manera reutilizable a elementos que por defecto no la poseen. Las demás clases están diseñadas para distribuir la lógica más organizadamente como es usual en la programación orientada a objetos.

A continuación una explicación más detallada de cada clase, del valor agregado con respecto al template y su extensión en líneas de código.

### 8.2.1 Especificación clase Main

| <b>Main</b>  |
|--|
| +machine: States   |
| -init_state_machine()<br>+config()<br>+onevent()<br>+ready()<br>+init_controls()<br>+first_state() |

**Fig. 19** Especificación Main

Esta clase configura los inputs de usuario, crea y llama los estados, y carga los elementos externos.

No sufrió ningún cambio en su estructura lógica con respecto al template que se utilizó. Si se configuró para que cargue las diferentes imágenes que componen el diseño gráfico del juego para poder ser utilizadas posteriormente, por lo demás quedó intacta. Posee 186 líneas de código.

### 8.2.2 Especificación PauseState

| <b>PauseState</b>  |
|--|
| +canvas: Canvas<br>+focus: ControllerFocus<br>+resume: Void<br>+fondo: Image |
| +new()<br>+onenter()<br>+onleave()<br>+update()                              |

**Fig. 20** Especificación PauseState

Este estado es llamado desde el GameState para pausar el juego y mostrar las instrucciones, no tiene mucho trasfondo lógico debido a que su propósito no es más que el anteriormente mencionado.

Esta clase no sufrió cambios en su lógica, puesto a que solo se le añadieron características gráficas. Posee 74 líneas de código.

### 8.2.3 Especificación GameState

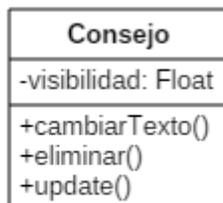
| GameState   |
|---|
| -state_machine: States<br>-canvas: Canvas<br>-focus: ControllerFocus<br>-block: Sprite<br>-fondo: Sprite<br>-victoria: Sprite<br>+etFichasRestantes: Text<br>+textoReserva1: Text<br>+textoReserva2: Text<br>-cambioTurno: Button<br>-instrucciones: Button<br>-posicionMouse: Vector<br>-consejo: Consejo<br>-pista1: Pista<br>-pista2: Pista<br>-recursos: Recursos<br>-ruleta: Ruleta<br>+premios: Premios<br>-arrastrador: Arrastrador<br>+turno: Int<br>-anchoFichas: Float<br>+fichasRestantes: Float<br>-tipoDeFichaArrastrada: Int<br>-juegoTerminado: Bool |
| +new()<br>+onenter()<br>+onmousedown()<br>+onmousemove()<br>+obtenerPista()<br>+cambiarTurno()<br>+primerTurno()<br>+siguienteTurno()<br>+reiniciar()<br>+pause()<br>+resume()<br>+onmouseup()<br>+update()   |

**Fig. 21 Especificación GameState**

Este es el estado principal del videojuego, alrededor de él ocurren todas las mecánicas y la lógica. Principalmente administra los turnos, decide que consejo entregar y procesa a que clase acudir cuando se toma o se suelta una ficha. También se encuentra la configuración gráfica de cómo están ordenados los elementos dentro de la pantalla.

Naturalmente ésta clase fue modificada casi en su totalidad. Posee 302 líneas de código.

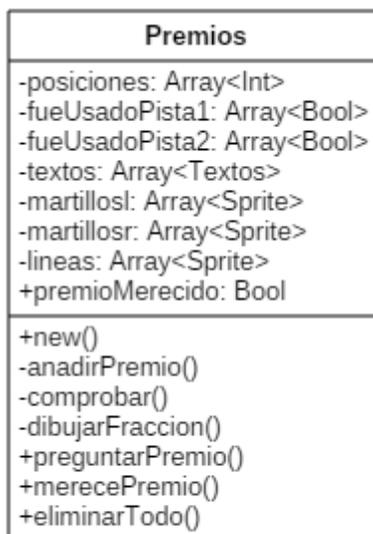
### 8.2.4 Especificación Consejo



**Fig. 22 Especificación Consejo**

Se trata de una clase simple que representa al texto del consejo. Su única función lógica es crear el efecto que cuando se cambia el consejo el texto desaparece y reaparece el nuevo lentamente. El contenido del consejo no se decide en esta clase, si no en GameState. Posee 33 líneas de código.

### 8.2.5 Especificación Premios



**Fig. 23 Especificación Premios**

Es la clase que genera los 4 premios para cada jugador aleatoriamente en la pista y los dibuja. Adicionalmente GameState le pregunta si el jugador activo esta sobre un premio, si tiene el premio desbloqueado y si el premio está reclamado, con lo que GameState pueda decidir qué consejo poner y si se merece un turno extra. Posee 185 líneas de código.

### 8.2.6 Especificación Ruleta

| Ruleta  |
|---|
| -indiceSeleccionado: Int<br>-fichasRestantes: Int<br>-periodo: Float<br>-tiempoAcumulado: Float<br>+encendido: Bool<br>-gameState: GameState<br>-orientacion: Int |
| +init()<br>+setGameState()<br>+seleccionarSiguiente()<br>+encender()<br>+apagar()<br>-update()  |

**Fig. 24 Especificación Ruleta**

Esta clase es la flecha que cambia según el turno, tiene el efecto gráfico de ruleta con los números, y asigna el número aleatorio en fichas restantes en GameState al principio de cada turno. Posee 67 líneas de código.

### 8.2.7 Especificación Recursos

| Recursos  |
|---|
| +spawn: Array<Array<Sprite>><br>+restantes: Array<Array<Text>><br>+quedan: Array<Array<Text>><br>-pista1: Pista<br>-pista2: Pista |
| +new()<br>+clickSpawn()<br>+restituir()<br>+identificarTipoAlto()<br>+eliminarTodo()  |

**Fig. 25 Especificación Recursos**

Representa las fichas que el usuario tiene pero que aún no están colocadas en las pistas. Cuando GameState toma una ficha, le pregunta a reservas y a recursos para

saber desde donde procede y a qué tipo de ficha pertenece y empezar a arrastrarla. También administra los textos que muestran cuantas fichas restantes de cada tipo quedan, y en el caso de que no quede ninguna la imagen de la ficha en los recursos desaparece hasta que se recuperen fichas de ese tipo. En las clases, la misma pregunta resta una ficha según corresponda. Posee 170 líneas de código.

### 8.2.8 Especificación Pista

| Pista  |
|--|
| +listaFichas: Array<Sprite><br>+acum: float<br>+reserva: Reserva<br>-anchoFichas: float<br>-recursos: Recursos<br>-turno: Int  |
| +init()<br>+setTurno()<br>+setReserva()<br>+setRecursos()<br>+colocar()<br>+estaLlena()<br>+identificarReemplazo()<br>+obtenerFichasAcumuladas()<br>+intercambiar()<br>+anadirFicha()<br>-ajustarFichas()<br>+identificarTipoAlto()<br>+eliminarTodo() |

**Fig. 26 Especificación Pista**

Representa a los edificios, administra gráficamente las fichas que se van colocando. Hace todo el trabajo de colocar fichas, que implica verificar si las puso dentro del edificio, si cabe en la pista, colocar un tipo distinto según cual sea la ficha, etc.. También se encarga del trabajo de comprobar si los reemplazos son matemáticamente correctos e intercambiar fichas. Posee 168 líneas de código.

### 8.2.9 Especificación Reserva



**Fig. 27 Especificación Reserva**

Representa las fichas que el usuario tiene guardadas en las reservas. Cuando GameState toma una ficha, le pregunta a reservas y a recursos para saber desde donde procede y a qué tipo de ficha pertenece y empezar a arrastrarla. A diferencia de los recursos que la imagen de las fichas es una para cada tipo de ficha, en las reserva hay tantas imágenes como fichas reservadas, se muestra una encima de otra, similar a cómo lo hacen en los edificios. Posee 57 líneas de código.

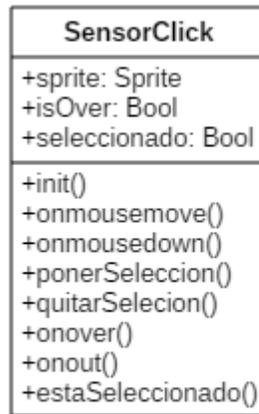
### 8.2.10 Especificación Arrastrador



**Fig. 28 Especificación Arrastrador**

Este componente es utilizado por GameState cuando arrastra una ficha. Cuando Recursos o a Reservas le dicen a GameState que se tomó una ficha válida, se crea una entidad con la imagen de la ficha del tipo apropiado y se le asigna este componente para que siga los movimientos del mouse. Posee 50 líneas de código.

### 8.2.11 Especificación SensorClick



**Fig. 29 Especificación SensorClick**

Es un componente utilizado por Pista. La razón por la cual no es utilizada por el resto de las clases es que este componente recuerda que la entidad a la que se le asignó fue clicada anteriormente, esto es útil para cuando el usuario quiere cambiar la posición de dos fichas, porque es necesario recordar cual ficha fue seleccionada primera cuando se selecciona la segunda. Posee 70 líneas d código.

## 9 INVESTIGACIÓN

### 9.1 Diseño de los controles

Para medir el impacto del videojuego en el aprendizaje de fracciones se tomó a un curso de 3° y 4° básico, se les aplicó un test de control antes de que jugaran, luego se les aplicó el mismo test, para luego poder comparar si existe alguna diferencia

La manera de evaluar fue propuesta por el profesor Jairo Navarrete y retocada en detalles por el profesor Clemente Rubio y el desarrollador. Se trata de mostrarle una serie de fracciones al alumno y él debe ubicarlos en la recta geométrica. Los números elegidos deben tener los mismos denominadores del juego, estos son un número dividido en ocho, cuatro o dos.

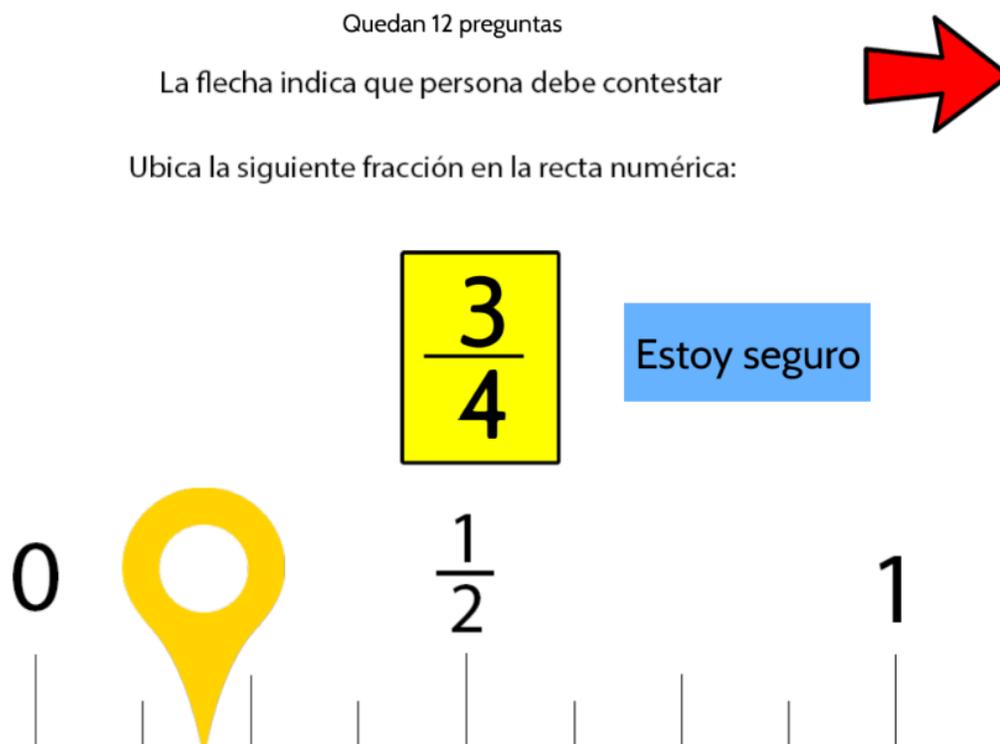
### 9.2 Programación y diseño gráfico de los controles

Los test se diseñaron para asemejarse lo más posible a la idea propuesta por el profesor Jairo, manteniendo su simplicidad y tomando en cuenta que dos alumnos deben realizar el test en la misma pantalla. El diseño es el siguiente:



Fig. 30 Diseño test

Aparece un número elegido aleatoriamente de una lista de un octavo hasta ocho octavos excluyendo el cuatro octavos, una vez se elige aparece simplificado, y el alumno al que apunte la flecha debe responder. No se pregunta dos veces por el mismo número al mismo alumno. Cuando el alumno elija un lugar en la recta, el juego se ve así:

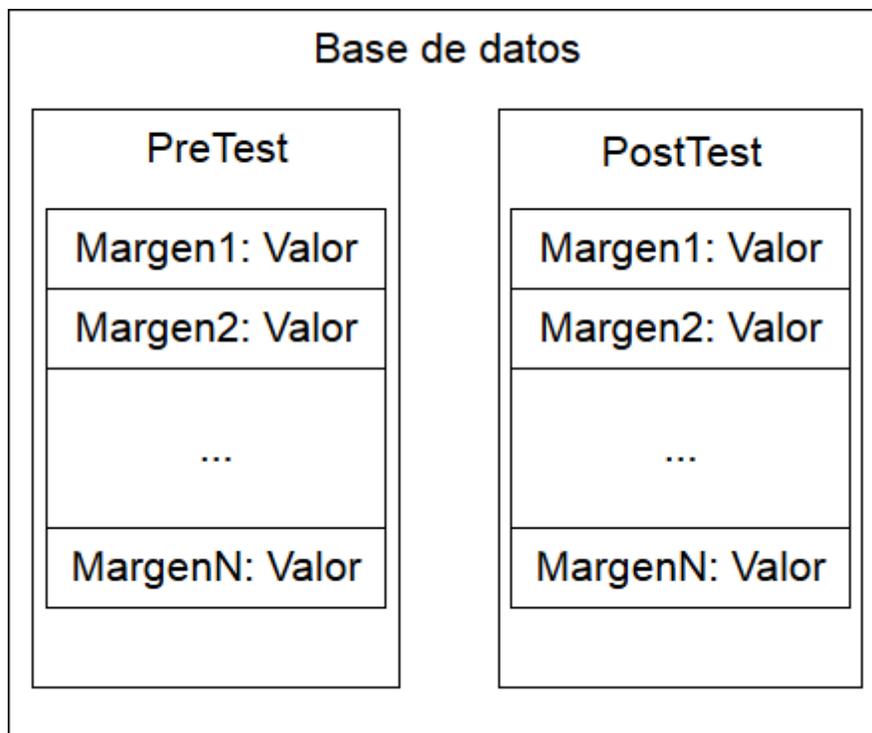


**Fig. 31 Diseño test lugar seleccionado**

Cuando el alumno elija un punto en la recta el botón de confirmar se activa y aparece esta aguja, puede moverla cuantas veces quiera antes de confirmar, cuando lo haga la flecha roja cambiará de lugar, el botón de confirmar se desactivará, aparecerá un nuevo número y se subirá la información de su respuesta a la base de datos.

La base de datos utilizada es firebase database, es una base de datos gratuita proporcionada por google. Se trata de una base de datos no relacional y permite construir aplicaciones que se actualizan en tiempo real.

En este caso solo se almacena el margen de error de cada respuesta medido en pixeles. La estructura de la base de datos es la siguiente:



**Fig. 32 Base de datos**

Cada dato es la cantidad de pixeles por la cual el jugador erró. No es necesario guardar información personal para identificar a cada alumno en particular, porque se tiene pensado comparar el promedio de margen de error de los resultados del pre test con los del post test. Esta información se puede comparar en un panel simple:

Reiniciar datos

**Promedio Pretest:**

**211.28571428571428 de 14 datos**

**Promedio Posttest:**

**193.92857142857142 de 14 datos**

**Fig. 33 Panel de resultados**

Los datos se actualizan en tiempo real, los datos de la imagen anterior son solo de ejemplo.

Las páginas para ingresar a la prueba de control y a la prueba de resultados son <http://youractionsdefineyou.com/pre> y <http://www.youractionsdefineyou.com/post/> respectivamente. El panel se aloja en <http://www.youractionsdefineyou.com/paneltest/>.

### 9.3 Resultados

La siguiente cita pertenece a la profesora que realizó las pruebas:

“El primer curso fue 222 de 14 datos en el pre y 130 de 25 datos en el post (un cuarto básico) Apretaron reiniciar datos entre medio de la clase por eso yo creo la diferencia de datos.”

“El segundo curso fue un tercero básico con 227.15 en el pre test y 126 datos y 215.23 en el post test con 126 datos y en el cuarto básico 180.22 con 140 datos y luego 174.9 con 140 datos”

“Los niños y las niñas quedaron fascinados con el juego, aunque no comprendieron el pre y el post test, hubo detalles que uno no imagina que le parecen tan atractivos como la carita al final de cada etapa esa :D es una cosa muy sencilla que ellos señalaron como muy importante”

“Para realizar la actividad no realicé ninguna mediación previa en los dos primeros cursos, las únicas instrucciones que di fue abrir la carpeta (que había creado antes) con accesos directos con los números 1,2,3y 4, al finalizar cada etapa me llamaban y les autorizaba a pasar a la siguiente, pero nada más , no resolví dudas respecto del cómo contestar ni del contenido; sin embargo presentaron bastantes dificultades para poder responder razón por la que con el último curso di instrucciones previas antes de ir a la sala de computación solo respecto del cómo hacer los reemplazos, aun así hubo preguntas sobre los mismo durante el juego por lo que creo que las instrucciones no son lo suficientemente claras como para ser entendidas”

“Respecto del pre y el post test los niños y niñas no logran entender que la recta numérica que se les presenta abajo es un entero y esto es súper importante ya que en matemáticas se les enseña que el enunciado es más importante que el dibujo ya que el dibujo es una representación que no siempre es real (como en geometría con las medidas) que solo se usa para representar algo entonces se debe señalar no solo con los números que eso es un entero, además antes del pre test debería existir una instrucción una hoja previa que diga algo como a continuación van a responder una serie de preguntas sobre fracciones en la recta numérica, o algo así.”

“Respecto del juego en sí, las instrucciones no son nada de claras, deberían partir diciendo que los ladrillos con los que se debe construir son los de  $\frac{1}{8}$  sin embargo lo primero que dice es que hay que colocar los ladrillos de 1 entero entonces lo primero que hacen los y las peques es intentar construir el edificio con ladrillos de 1 entero, luego el juego es intuitivo (aunque no tanto) demoran como 5 minutos en darse cuenta que se parte con los ladrillos de  $\frac{1}{8}$  y los que demoraron más yo le dije como hacerlo y los invité a leer el cuadro con instrucciones para los turno, cuadro que pasa desapercibido para ellos.. quizás ponerlo en otro color, o bajo la flecha del turno correspondiente “

“Otra cosa que me parece no concuerda es que en el juego se presentan dos enteros pero en el pre y pos test 1 entero entonces los niños y niñas no logran hacer la relación entre uno y otro además que en el pre y pos test se presentan de forma horizontal y luego en el juego de forma vertical, cuando en alguna unidad uno quiere hacer un

traspaso de ese tipo (desde una representación más concreta como es el juego a una más abstracta como es la recta numérica) ésta demora como 4 clases en ser entendida por los peques y requiere que en algún momento las vean juntas”

“Ahora yo creo que la herramienta del juego es útil en el momento que se está enseñando el concepto de fracciones equivalentes y el pre y post test debería ser sobre eso y no sobre ubicarla en la recta numérica, porque en el juego se trabaja con equivalencia de fracciones que aunque parezcan contenidos similares no lo son y se enseñan distinto sólo cuando dominas uno puedes usarlo para ayudarte en el otro, una chica que sabe cuáles fracciones son equivalentes responderá más rápido la ubicación de ésta en una recta numérica, sólo si es que entiende que representa la recta numérica”

“En fin yo creo que con una buena mediación por parte de un docente el juego tiene potencial para ayudar a lograr el aprendizaje, pero sin ella creo que en 45 minutos de clase se puede hacer otro tipo de actividad que colabore más que el juego para el logro del mismo aprendizaje”

“Los chicos y chicas de estos cursos trabajan una vez por semana con el software khanacademy entonces no es una variable la dificultad o lejanía con el uso de software educativos en sala.”

Los resultados fueron inesperados a varios niveles, a continuación se realizará un análisis cualitativo, cuantitativo y una conclusión.

Se presentaron problemas a la hora de entender el juego y los tests. Por un lado el juego empezaba diciendo que el objetivo principal era colocar las fichas de un entero no dejaba en claro en el primer momento que sólo se podían poner fichas de  $1/8$ , cuando el juego se estaba desarrollando se asumió que los niños se apoyarían más en los consejos, pero en cambio la profesora indica que no le prestaron atención, al menos en un comienzo. Con respecto a los tests, fueron diseñados pensando en que los alumnos encontrarían intuitivo interactuar con la recta geométrica lo cual no fue efectivo. Por otro lado el juego tuvo buena recepción en el sentido de que les fue entretenido.

Si se analizan los datos, los alumnos de 3° básico no lograron un avance significativo luego de experimentar con el videojuego, esto se ve explicado por lo indicado por la profesora: “...la herramienta del juego es útil en el momento que se está enseñando el concepto de fracciones equivalentes y el pre y post test debería ser sobre eso y no sobre ubicarla en la recta numérica...”, los alumnos necesitan una base matemática un poco más sólida antes de poder hacerles este tipo de preguntas, de lo contrario les es difícil aplicar el conocimiento adquirido del juego en los tests. No ocurrió lo mismo con los alumnos de 4° básico que sí lograron una notoria mejora una vez que probaron el videojuego, su margen de error se redujo a un 59% aproximadamente de lo que era originalmente. Como otro dato, posteriormente realizados los tests, se tuvo la idea de revisar cuál era la cantidad de respuestas perfectas de los test, esto quiere decir que el alumno erró por menos de tres pixeles la respuesta correcta, y resulta que para el último grupo a pesar de aparentemente no mostrar casi mejoras, las respuestas perfectas aumentaron un 21,6% (de 37 a 45), esto sin embargo no es significativo, porque fue probada en solo 10 alumnos y puede deberse a la suerte, por lo que solo es un dato que puede ser prometedor en vez de ser uno concluyente. No se tienen datos de respuestas

perfectas en otros grupos debido a que ésta manera de medición fue ideada cuando los datos de los cursos posteriores ya había sido eliminada

Originalmente las mecánicas del videojuego fueron pensadas en torno a la analogía de completar una recta geométrica mediante reemplazos, luego se enfatizó tanto en los reemplazos que la recta quedó en un segundo plano. Esto quedó en evidencia en los resultados de las pruebas, que muestran que el videojuego solo tuvo efectos significativos en la medida que los alumnos se hayan familiarizado un poco con esta idea abstracta con anticipación. Lo anterior fue una sorpresa, puesto que la idea del videojuego en un principio era familiarizar a los niños a elementos matemáticos nuevos de manera lúdica e intuitiva, al final el punto fuerte terminó siendo mejorar las capacidades de alumnos que ya tenían una cierta base en estos ámbitos.

Como primer intento de realizar un videojuego de esta naturaleza los resultados son buenos, pero hay que mejorar 2 grandes ámbitos:

- debido a que no se presta mucha atención a los consejos, convendría simplificar las mecánicas del videojuego y explicarlo en su totalidad desde un principio, de todos modos no se descarta completamente el uso de consejos.
- rediseñar completamente los tests para que capten realmente el aprendizaje por que actualmente solo se mide como influye la facilidad de realizar equivalencias matemáticas a la facilidad de identificar un punto en la recta.

Otra idea útil para una continuación del proyecto es seguir creando videojuegos desde cero, ver qué hacen mejor y qué hacen peor y sobre ese conocimiento se darán las bases para ir generando ideas mucho más interesantes tanto a nivel pedagógico como a nivel de entretención por parte de los alumnos.

## 10 PRUEBAS

### 10.1 Elementos de prueba

Los elementos de prueba son:

- Funcionalidades del videojuego: se probará que todas las mecánicas del juego funcionen como es esperado.
- Funcionalidades de los controles: se probará que el control previo y el control posterior funcionen como es esperado y que se muestre los resultados consistentemente.

### 10.2 Especificación de las pruebas

| Características a probar       | Nivel de prueba | Objetivo de la Prueba  | Enfoque para la definición de casos de prueba | Técnicas para la definición de casos de prueba            | Actividades de prueba   | Criterios de cumplimiento   |
|--------------------------------|-----------------|--|---|---|---|---|
| Funcionalidades del videojuego | Unitaria        | Minimizar la ocurrencia de bugs  | Caja negra                                    | El input vía interacción de las mecánicas del videojuego. | Llevar el videojuego a un contexto en que se pueda probar la mecánica y ejecutarla. | El videojuego responde como es esperado y sin efectos secundarios.              |
| Funcionalidades de los test    | Unitaria        | Minimizar la ocurrencia de bugs y comprobar que se almacenen y muestren los datos reales | Caja negra                                    | El input vía interacción de las mecánicas de los tests    | Llevar los tests a un contexto en que se pueda probar la mecánica y ejecutarla      | Los tests responden como es esperado y la información entregada es consistente. |

Tabla 1. Especificación de pruebas

### 10.3 Responsables de las pruebas

La totalidad de las pruebas serán realizadas por el desarrollador

### 10.4 Calendario de pruebas

Las pruebas se separan en dos fases, la primera se enfoca en el correcto funcionamiento del videojuego en su conjunto y la segunda corresponde al correcto funcionamiento de los controles que irán antes y después de que los alumnos jueguen.

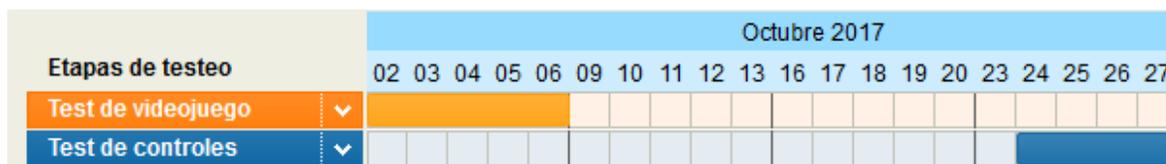


Fig. 34 Carta Gantt de pruebas

## 10.5 Detalle de las pruebas

### 10.5.1 PU-01 Funcionalidades del videojuego

En este ítem se especifican:

- Configuración: Las pruebas se realizan en el navegador Mozilla Firefox sin la necesidad de realizar ningún tipo de preparación.
- Pre-condiciones: Se debe preparar el juego para que cada funcionalidad se pueda dar.

| ID Caso De Prueba | Características a Probar   | Entrada  | Salida esperada  | Salida Obtenida  | Éxito / Fracaso | Observaciones     |
|-------------------|--|--|--|--|-----------------|-------------------|
| CPU-01            | Se muestran las instrucciones  | Clic en el botón de instrucciones  | Pantalla de instrucciones  | Pantalla de instrucciones  | Éxito           | Sin observaciones |
| CPU-02            | Se puede regresar al juego desde las instrucciones   | Clic en el botón de regresar al juego  | Pantalla de juego  | Pantalla de juego  | Éxito           | Sin observaciones |
| CPU-03            | Funcionan los cambios de turno, otorgando y quitando capacidades a los jugadores según corresponda | Clic en el botón "fin turno"   | Cada jugador tiene las capacidades que corresponden según su turno                     | Cada jugador tiene las capacidades que corresponden según su turno                     | Éxito           | Sin observaciones |
| CPU-04            | La ruleta funciona correctamente al iniciar los turnos   | Clic en el botón "fin turno"   | La ruleta realiza el efecto gráfico y asigna un número aleatorio de fichas disponibles | La ruleta realiza el efecto gráfico y asigna un número aleatorio de fichas disponibles | Éxito           | Sin observaciones |
| CPU-05            | Colocar fichas con fichas disponibles  | Arrastrar una ficha en el edificio o en las reservas teniendo fichas disponibles               | La ficha se sitúa en el edificio o las reservas y se resta una ficha disponible        | La ficha se sitúa en el edificio o las reservas y se resta una ficha disponible        | Éxito           | Sin observaciones |
| CPU-06            | Colocar fichas sin fichas disponibles  | Arrastrar una ficha en el edificio o en las reservas teniendo fichas disponibles               | La ficha regresa a los recursos  | La ficha regresa a los recursos  | Éxito           | Sin observaciones |
| CPU-07            | Colocar fichas desde la reserva  | Arrastrar una ficha desde la reserva al edificio   | La ficha se sitúa en el edificio   | La ficha se sitúa en el edificio   | Éxito           | Sin observaciones |
| CPU-08            | Reemplazo de fichas matemáticamente correcto   | Arrastrar una ficha hacia fichas que en su conjunto valgan lo mismo que la ficha arrastrada    | La ficha las reemplaza, y las antiguas regresan a los recursos                         | La ficha las reemplaza, y las antiguas regresan a los recursos                         | Éxito           | Sin observaciones |
| CPU-09            | Reemplazo de fichas matemáticamente incorrecto   | Arrastrar una ficha hacia fichas que en su conjunto no valgan lo mismo que la ficha arrastrada | La ficha regresa a los recursos  | La ficha regresa a los recursos  | Éxito           | Sin observaciones |
| CPU-10            | Cambio de posición de fichas   | Clic en dos fichas puestas   | Las fichas seleccionadas cambian de lugar  | Las fichas seleccionadas cambian de lugar  | Éxito           | Sin observaciones |
| CPU-11            | Estando en una zona premiada, se puede desbloquear un turno extra                                  | Hacer un reemplazo estando en una zona premiada  | El martillo y el consejo avisan que se posee un turno extra listo para reclamar, el    | El martillo y el consejo avisan que se posee un turno extra listo para reclamar, el    | Éxito           | Sin observaciones |

|        |   |  |  |  |       |                   |
|--------|---|--|--|--|-------|-------------------|
|        |   |  | botón de fin turno se transforma en turno extra  | botón de fin turno se transforma en turno extra  |       |                   |
| CPU-12 | Habiendo desbloqueado un turno extra, se puede reclamar | Hacer clic en turno extra  | Empieza normalmente un turno nuevo para el jugador que se ganó el turno                                    | Empieza normalmente un turno nuevo para el jugador que se ganó el turno                                    | Éxito | Sin observaciones |
| CPU-13 | Los martillos cambian de imagen según corresponda       | Generar las condiciones de transformación para cada fase           | Las imágenes de los martillos cambian como es esperado   | Las imágenes de los martillos cambian como es esperado   | Éxito | Sin observaciones |
| CPU-14 | Las condiciones de victoria hacen que el jugador gane   | Se colocan las dos fichas de un entero en el edificio              | Las fichas del edificio cambian por un edificio real y se da a conocer que el jugador ganó en el consejo   | Las fichas del edificio cambian por un edificio real y se da a conocer que el jugador ganó en el consejo   | Éxito | Sin observaciones |
| CPU-15 | Los consejos cambian según el contexto                  | Generar las condiciones en las que se diseñó que el consejo cambie | Los consejos cambian según lo esperado   | Los consejos cambian según lo esperado   | Éxito | Sin observaciones |
| CPU-16 | Los premios se generan aleatoriamente consistentemente  | Iniciar el juego y ver donde tocaron los premios                   | Los premios se generan aleatoriamente, separados por un margen y se simplifica su posición automáticamente | Los premios se generan aleatoriamente, separados por un margen y se simplifica su posición automáticamente | Éxito | Sin observaciones |

**Tabla 2. Prueba de funcionalidad videojuego**

### 10.5.2 PU-02 Funcionalidades de los controles

En este ítem se especifican:

- Configuración: Las pruebas se realizan en el navegador Mozilla Firefox sin la necesidad de realizar ningún tipo de preparación.
- Pre-condiciones: Se deben preparar los controles para que cada funcionalidad se pueda dar.

| ID Caso De Prueba | Características a Probar   | Entrada                    | Salida esperada   | Salida Obtenida   | Éxito / Fracaso | Observaciones     |
|-------------------|--|----------------------------|---|---|-----------------|-------------------|
| CPU-16            | Se puede pasar de un turno a otro                                      | Clic en el botón confirmar | La flecha apunta al otro estudiante y se reinicia todo lo hecho el turno anterior | La flecha apunta al otro estudiante y se reinicia todo lo hecho el turno anterior | Éxito           | Sin observaciones |
| CPU-17            | Los números preguntados no se repiten y se simplifican automáticamente | Empezar turnos             | Siempre se pregunta por números nuevos en un orden aleatorio                      | Siempre se pregunta por números nuevos en un orden aleatorio                      | Éxito           | Sin observaciones |
| CPU-18            | El indicador de posición   | Clic en la recta           | El indicador de   | El indicador de   | Éxito           | Sin               |

|        |   |  |   |   |       |                   |
|--------|---|--|---|---|-------|-------------------|
|        | se puede cambiar como el usuario desee antes de confirmar |  | posición se mueve libremente antes de confirmar | posición se mueve libremente antes de confirmar |       | observaciones     |
| CPU-19 | Al confirmar se sube la información consistentemente      | Elegir una posición en la recta y confirmar  | Se suben los datos reales de margen de error    | Se suben los datos reales de margen de error    | Éxito | Sin observaciones |
| CPU-20 | Calculo correcto de los promedios de pre-test y post-test | Subir respuestas desde el juego, calcular a mano y comparar con lo que dice el panel | La información entregada es consistente         | La información entregada es consistente         | Éxito | Sin observaciones |

**Tabla 3. Prueba funcionalidad de los controles**

## 10.6 Conclusiones de Pruebas

El proceso de pruebas ha sido realizado con éxito para cada funcionalidad especificada, garantizando que en la ejecución de cada software se ejecutarán todas las características correctamente y sin ningún efecto secundario no deseado.

---

## 11 RESUMEN ESFUERZO REQUERIDO

---

El tiempo estimado invertido en cada una de las fases fueron las siguientes:

| <b>Actividades/fases</b>                 | <b>Horas</b> |
|--|--------------|
| Capacitación en tecnologías nuevas       | 90           |
| Diseño conceptual de mecánicas del juego | 50           |
| Concepción y diseño gráfico              | 200          |
| Implementación del videojuego            | 250          |
| Rediseño de acuerdo a feedback           | 50           |
| Pruebas                                  | 30           |
| Diseño e implementación de los controles | 50           |
| Documentación                            | 100          |
| total                                    | 820          |

**Tabla 4. Estimación de esfuerzo requerido**

En un comienzo no se tomaron en cuenta ciertos aspectos del proyecto que resultaron consumir una gran parte del tiempo. La dificultad de agendar reuniones con las alumnas de pedagogía, el poco conocimiento de diseño gráfico y la dificultad para diseñar un videojuego, teniendo en cuenta las limitaciones pedagógicas, son algunos ejemplos de dificultades que no fueron previstas en la estimación inicial del proyecto.

---

## 12 CONCLUSIONES

---

El objetivo propuesto era la medición del impacto de los videojuegos en el aprendizaje de fracciones, se diseñó e implementó un videojuego enfocado a ayudar a alumnos en sus primeros pasos en el aprendizaje de fracciones. Al medir su impacto los resultados fueron positivos, pero no en el sentido que se esperaba, no se mostraron resultados positivos en los alumnos de 3° básico debido a que la manera de medir sus mejoras resultó ser tan indirecta que no pudieron aplicar el conocimiento supuestamente adquirido en las pruebas. Sin embargo se mostró una clara mejora en los alumnos de 4° básico, que tras experimentar el videojuego disminuyeron notoriamente el margen de error en los test a pesar ser un método de evaluación indirecto. Con los aciertos y errores de este proyecto se facilita la realización de futuros proyectos similares, tanto el diseño de un videojuego didáctico como la manera de medir su impacto.

Cada herramienta del entorno de desarrollo se desarrolló perfectamente y en ningún momento se sintió que sea preciso reemplazar alguna. Luxe fue bastante versátil e intuitivo de aprender y utilizar debido a que utiliza programación orientada a objetos, su sintaxis también es relativamente simple tanto en la parte gráfica como la lógica. Y se puede asegurar que Photoshop tiene su buena fama bien ganada ya que fue imprescindible para el dibujo de los elementos de éste videojuego.

A nivel académico este proyecto me presentó un gran desafío por tener un carácter tan multidisciplinario, casi la totalidad de las tecnologías requeridas para este proyecto las aprendí de manera autodidacta durante el transcurso del mismo, lo único que conocía con anticipación era la plataforma Firebase y JavaScript para desarrollar el pequeño panel, que de todas formas las aprendí autodidactamente. Aparte del manejo de nuevas tecnologías, el diseño de las mecánicas del videojuego y el diseño gráfico fueron retos aún más difíciles, ya que fueron actividades completamente nuevas para mí, por lo que se requirió improvisar sobre la marcha ciertos procedimientos para la elaboración de éstos contenidos, procedimientos diseñados a base de intento y error. A pesar de eso siento que lo que más me deja este proyecto no son las competencias académicas, las técnicas o el manejo de nuevas herramientas, sino habilidades personales, tales como conocer mis puntos fuertes y débiles, autocrítica, la resiliencia, y más.

Personalmente fue una muy buena experiencia llevar a cabo este proyecto a pesar de las dificultades que trajo, es inusual y apasionante que un proyecto exija tanta creatividad e improvisación además de la capacidad de análisis e ingenio que un proyecto de informática requiere.

---

## 13 BIBLIOGRAFÍA

---

Para descarga, instalación, guías, documentación API, y código fuente de Luxe

<<https://luxengine.com/>>

Librerías y ejemplos de proyectos en Luxe

<<https://github.com/anissen/awesome-snowkit/>>

Documentación API, y unidad para probar código online de Haxe

<<https://haxe.org/>>

Mision, visión y organigrama de la Universidad del Bio-Bio

<<https://www.ubiobio.cl/>>

Misión, visión y organigrama de la facultad de ciencias empresariales.

<<https://www.webface.ubiobio.cl/>>

Anderson CA & Bushman BJ (2001). Effects of violent video games on aggressive behavior, aggressive cognition, aggressive affect, physiological arousal and prosocial behavior: a meta-analytic review of the scientific literature. *Psychological Science*, 12(5), 353–359.

BECTA (2006). Computer Games in Education: Report.

Disponible en: <http://partners.becta.org.uk/index.php?section=rh&rid=13595>

BECTA (2006). The Becta Review: Evidence on the Progress of ICT in Education.

Disponible en: <http://publications.becta.org.uk/download.cfm?res ID=259486>

Benton P (1995). Conflicting cultures: reflections on the reading and viewing of secondary-school pupils. *Oxford Review of Education*, 21(4), 457–470.

Colwell J, Grady C, Rhaiti S (1995). Computer games, self-esteem and gratification of needs in adolescents. *Journal of Community and Applied Social Psychology*, 5, 195–206.

Emes CE (1997). Is Mr Pac Man eating our children? A review of the impact of video games on children. *Canadian Journal of Psychiatry*, 42(4), 409–414.

Disponible en: [www.cpa-apc.org/Publications/Archives/PDF/1997/May/EMES.pdf](http://www.cpa-apc.org/Publications/Archives/PDF/1997/May/EMES.pdf)

European Schoolnet (2009). Videojuegos en el aula XLIV .

Informe final disponible en: [http://games.eun.org/upload/gis-full\\_report\\_en.pdf](http://games.eun.org/upload/gis-full_report_en.pdf).

Informe resumido disponible en: [http://games.eun.org/upload/gis-synthesis\\_report\\_en.pdf](http://games.eun.org/upload/gis-synthesis_report_en.pdf)

Futurelab (2007). Teaching with Games.

Informe final disponible en:

[http://www.futurelab.org.uk/resources/documents/project\\_reports/teaching\\_with\\_games/TWG\\_report.pdf](http://www.futurelab.org.uk/resources/documents/project_reports/teaching_with_games/TWG_report.pdf)

Gee, J. P. (2005). Good video games and good learning. In Phi Kappa Phi Forum (Vol. 85, No. 2, p. 33). THE HONOR SOCIETY OF PHI KAPPA PHI.

Gee, J. P. (2008). What Digital Games Have to Teach Us About Learning and Literacy. New York & Basingstoke: Palgrave Macmillan.

ISSA Press Release (2007). Digital games May Offer Health Benefits, Experts Suggest. Disponible en: <http://www.issaonline.com/press-room/downloads/exertainment.pdf>

Kirriemuir J (2002). The relevance of video games and gaming consoles to the higher and further education learning experience. April 2002. Techwatch Report TSW 02.01. Disponible en: [www.jisc.ac.uk/index.cfm?name=techwatch\\_report\\_0201](http://www.jisc.ac.uk/index.cfm?name=techwatch_report_0201), accessed 14 April.

Prensky, M (2006). Don't Bother Me Mom – I'm Learning!. St. Paul, MN.: Paragon House Publishers.

Prensky, M (2007). Computer games and learning: Digital Game-based learning.

Roe K, Muijs D (1998). Children and computer games – a profile of the heavy user. European Journal of Communication, 13(2), 181–200.

Roschelle, J., Kaput, J., & Stroup, W. (2000). SimCalc: Accelerating students' engagement with the mathematics of change. Innovations in science and mathematics education: Advanced designs for technologies of learning, 47-75.

Shaffer, D. W., Halverson, R., Squire, K. R., & Gee, J. P. (2005). Video Games and the Future of Learning. WCER Working Paper No. 2005-4. Wisconsin Center for Education Research (NJ1).

Shaffer, D. W. (2008). How Computer Games Help Children Learn. New York & Basingstoke: Palgrave Macmillan.

Teem (2002). Report on the Educational Use of Digital games.

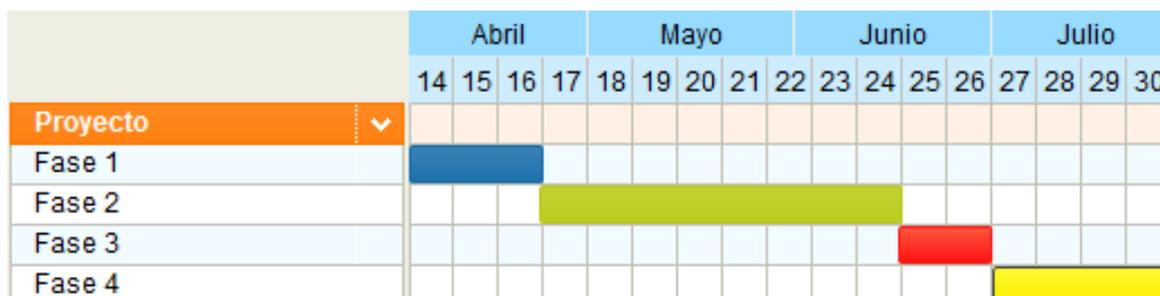
Disponible en: [http://www.teem.org.uk/publications/teem\\_gamesined\\_full.pdf](http://www.teem.org.uk/publications/teem_gamesined_full.pdf)

VanDeventer SS, White JA (2002). Expert behavior in children's video game play. Simulation and Gaming, 33(1), 28–48.

Walsh D (2002). Kids don't read because they CAN'T read. Education Digest, 67(5), 29–30.

## 14 ANEXO: PLANIFICACION INICIAL DEL PROYECTO

El desarrollo del software está dividido en 4 fases, inicialmente se asignaron los siguientes periodos de tiempo para cada una



**Fig. 35 Carta gantt planificación inicial**

Fase 1: Estudiar textos MINEDUC, unidades didácticas en matemáticas empleadas para la enseñanza de las fracciones, hablar con el profesor Jairo Navarrete. En esta fase se estudiará la información necesaria para construir el documento experto de diseño inicial de los videojuegos educativos de fracciones.

Fase 2. Diseño e implementación de videojuegos basadas en las actividades de aprendizaje, caracterización de los estudiantes, expertos, textos de MINEDUC.

Fase 3. Análisis y evaluación del software desde el punto de vista de la ingeniería del software. El objetivo de esta fase es el análisis y evaluación del videojuego como producto software.

Fase 4. Análisis, monitorización y Evaluación de los videojuegos desde el punto de vista didáctico. El objetivo de esta fase es el análisis, monitorización y evaluación del impacto de la innovación.

### 14.1.1 Estimación inicial de tamaño

| <b>Factor de peso de los actores sin ajustar (UAW)</b>   |   |                |             |                |
|--|---|----------------|-------------|----------------|
| Tipo de actor  | Descripción   | Factor de peso | Nro Actores | Subtotal (UAW) |
| Simple   | Otro sistema que actúa con el sistema a desarrollar mediante una interfaz de programación                       | 1              | 0           | 0              |
| Medio  | Otro sistema que interactúa con el sistema a desarrollar mediante a un protocolo o una interfaz basada en texto | 2              | 0           | 0              |
| Complejo   | Una persona que interactúa con el sistema mediante a una interfaz   | 3              | 2           | 3              |
| Total  |   |                |             | 6              |
| <b>Factor de peso de casos de uso sin ajustar (UUCW)</b> |   |                |             |                |

| Tipo de caso de uso                          | Descripción                                    | Factor de peso  | Nro de casos de uso | Subtotal (UUCW) |
|--|--|-----------------|---------------------|-----------------|
| Simple                                       | El caso de uso contiene de 1 a 3 transacciones | 5               | 7                   | 35              |
| Medio  | El caso de uso contiene de 4 a 7 transacciones | 10              | 0                   | 0               |
| Complejo                                     | El caso de uso contiene 8 o más transacciones  | 15              | 0                   | 0               |
| Total (UUCW)                                 |  |                 |                     | 35              |
| Total (UUPC)                                 |  |                 |                     | 41              |
| <b>Factores de complejidad técnica (TCF)</b> |  |                 |                     |                 |
| Factor                                       | Descripción                                    | Factor de peso  | Valor asignado      | Subtotal (TCF)  |
| T1   | Sistema Distribuido                            | 2               | 0                   | 0               |
| T2   | Objetivos de rendimiento                       | 1               | 3                   | 3               |
| T3   | Eficiencia respecto al usuario final           | 1               | 3                   | 3               |
| T4   | Procesamiento complejo                         | 1               | 1                   | 1               |
| T5   | Código reutilizable                            | 1               | 2                   | 2               |
| T6   | Instalación sencilla                           | 0,5             | 5                   | 2.5             |
| T7   | Fácil utilización                              | 0,5             | 4                   | 2               |
| T8   | Portabilidad                                   | 2               | 3                   | 6               |
| T9   | Fácil de cambiar                               | 1               | 3                   | 3               |
| T10  | Uso Concurrente                                | 1               | 5                   | 5               |
| T11  | Características de seguridad                   | 1               | 0                   | 0               |
| T12  | Accesible por terceros                         | 1               | 5                   | 5               |
| T13  | Se requiere formación especial                 | 1               | 0                   | 0               |
| Total  |  |                 |                     | 30              |
| <b>Factores de ambiente (EF)</b>             |  |                 |                     |                 |
| Factor                                       | Descripción                                    | Factor de peso  | Valor asignado      | Subtotal (EF)   |
| E1   | Familiar con el modelo de proyecto             | 1,5             | 3                   | 4.5             |
| E2   | Experiencia en la aplicación                   | 0,5             | 0                   | 0               |
| E3   | Experiencia con orientación a objetos          | 1,0             | 5                   | 5               |
| E4   | Capacidades de análisis                        | 0,5             | 4                   | 2               |
| E5   | Motivación                                     | 1,0             | 5                   | 4               |
| E6   | Requisitos estables                            | 2,0             | 1                   | 2               |
| E7   | Trabajadores a tiempo parcial                  | -1,0            | 0                   | 0               |
| E8   | Lenguaje complejo                              | -1,0            | 3                   | -3              |
| Total  |  |                 |                     | 14.5            |
| <b>Estimando los UCP</b>                     |  |                 |                     |                 |
| Parámetro                                    | Descripción                                    | Valor calculado |                     |                 |

|            |  |             |
|------------|--|-------------|
| UUCP       | Puntos de caso de uso no ajustados     | 41          |
| TCF        | Factores de complejidad técnica        | 0.9         |
| EF         | Factores de ambiente                   | 0,965       |
| <b>UCP</b> | <b>Puntos de caso de uso ajustados</b> | <b>35.6</b> |
|            | <b>Horas persona</b>                   | <b>10</b>   |
|            | <b>Horas hombre</b>                    | <b>356</b>  |

**Tabla 5. Estimación de tamaño por casos de uso**