

Diseño de un módulo de entrenamiento para una habilidad técnica en estudiantes de
Especialización en Ortopedia y traumatología.

Tania Estephany Vergara Bernal

Trabajo De Grado Para Optar El Título De Ingeniera de Sistemas

Director

Gabriel Rodrigo Pedraza Ferreira

Ph D. en informática

Codirector

Luis Bautista Rojas

MSc. En ingeniería informática

Universidad Industrial De Santander

Facultad De Ingenierías Fisicomecánicas

Escuela De Ingeniería de Sistemas e Informática

Bucaramanga

2019

Dedicatoria

A la memoria de mi hermano *Juanjo*, quien con su repentina partida me enseñó a superar los obstáculos con una sonrisa, a él que vivió plenamente cada día de su vida siendo feliz a pesar del dolor y la enfermedad, luchando hasta el último segundo, en su nombre y a ese ejemplo que él mismo se encargó de mantener latente en mi cuando quise rendirme.

A la mujer que me dio la vida, por ser mamá y papá a la vez, por ser ejemplo de lucha y perseverancia, a ella y su amor de madre, amor que todo lo puede, que todo lo cura, amor sincero e infinito.

Agradecimientos

A Dios porque sin el nada de esto hubiera sido posible, gracias por guiarme, por ser fortaleza en los momentos de debilidad, por brindarme una vida llena de aprendizajes y experiencias, pero sobre todo felicidad.

A la Universidad Industrial de Santander por brindarme una educación de excelente calidad, igualmente a la Universidad Popular Autónoma del Estado de Puebla, México, por acogerme como estudiante de intercambio y brindarme una experiencia enriquecedora.

A mi mamá Sandra, por ser la promotora de que este sueño se hiciera realidad, por el apoyo incondicional, esfuerzo, aliento, paciencia, por los consejos, por confiar y creer en mi cuando nadie más lo hizo.

A mi papá Rodrigo, que, aunque no es mi padre biológico me ha brindado todo el amor y cariño que he necesitado para llegar hasta donde estoy hoy.

A mi hermana Julieta, por ser mi motivación para alcanzar esta meta, espero ser un ejemplo a seguir, gracias por tanto coqueta.

A mis abuelos Ismenia y Leopoldo, por sacrificar tiempo y educarme con reglas, valores y libertades que me ayudaron a elegir lo que estaba bien y mal.

A la familia Ruiz Bernal por brindarme amor y cariño durante estos años de trayectoria, por brindarmen un hogar y ayudarme a levantar cuando caía, a Damian Alejandro y Henry Santiago por su amor incondicional y por cada lagrima que secaron con sonrisas llenas de pureza y felicidad extrema.

A la familia García Bernal por el apoyo y preocupación. A Yeison García por escuchar mis alegrías y tristezas.

A Sharik Daniela, Karen, Carlos, Alexy, Andrés y Sergio, quienes compartieron conmigo estos años de desarrollo profesional y contribuyeron de manera importante para alcanzar este logro.

Al profesor Gabriel Pedraza y Luis Eduardo Bautista quienes marcaron mi camino con sus enseñanzas, les agradezco por la confianza, apoyo y dedicación entregada.

Gracias, este logro no es solo mío es NUESTRO.

Tabla de Contenido

Introducción.....	20
Cumplimiento de Objetivos.....	22
1. Objetivos.....	23
1.1 Objetivo General.....	23
1.2 Objetivos Específicos.....	23
2. Generalidades del Proyecto.....	24
2.1 Justificación.....	24
2.2 Planteamiento del Problema.....	26
3. Marco Referencial.....	28
3.1 Enfoque de desarrollo basado en modelos.....	28
3.1.1 Ingeniería dirigida por modelos.....	28
3.1.2 Desarrollo de Software Dirigido por Modelos -.....	29
3.1.3 Desarrollo de Interfaces de Usuario Basado en Modelos (MBUIDE).....	30
3.1.3.1 Descripción de interfaces de usuario basada en modelos según el marco de referencia Cameleon.....	31
3.1.3.1.1 Modelado de tareas.....	32
3.1.3.1.2 Modelos de presentación.....	35
3.2 Realidad aumentada.....	38
3.3 Herramientas.....	38

3.3.1 Sistema optitrack.....	38
3.3.2 Unity.....	39
3.3.3 Natnet.....	39
4. Fase I: Establecer la habilidad técnica indispensable para la colocación de los tornillos transpediculares.....	39
4.1 Diseño de la ecuación de búsqueda.....	40
4.1.1 Exploración no estructurada	40
4.1.2 Agrupación de términos semejantes.....	41
4.1.3 Combinación de términos	42
4.1.4 Selección de la ecuación de búsqueda.....	43
4.2 Ejecución de la ecuación de búsqueda.	43
4.3 Recolección de información	44
4.3.1 Información esencial.	45
4.3.2 Fuentes primarias	48
4.3.2.1 Tipo y técnica de muestreo.	48
4.3.2.2 Diseño del instrumento para la recolección de información.....	48
4.3.2.3 Validación del instrumento de recolección de información.	49
4.3.2.4 Análisis de resultados	49
4.3.3 Fuentes primarias.	52
4.4 Elección de la habilidad técnica a entrenar.....	52
5. Fase II: Diseñar ejercicios para el entrenamiento de la habilidad coordinación óculo-manual para conformar el módulo de aprendizaje.....	53
5.1 ¿Qué es coordinación?.....	53

5.1.1 Tipos de coordinación.	54
5.2 ¿Qué es coordinación óculo-manual?	55
5.2.1 Fases de coordinación óculo- ojo	55
5.3 Diseño de ejercicios.	56
6. Fase III: Diseño de la herramienta	60
6.1 Primera etapa: Núcleo funcional.....	61
6.1.1 Diagramas de casos de uso.	61
6.1.2 Arquitectura del software	62
6.2 Interfaz de usuario	63
6.2.1 Modelos de tareas.	64
6.2.2 Modelo de dominio de los ejercicios.	69
6.2.3 Interfaz de usuario abstracta -AUI	71
6.2.3.1 Definición de categorías.	71
6.2.3.2 Reglas de transformación de modelo de tarea a AUI.	72
6.2.3.3 Aplicación de las reglas de transformación.	74
6.2.3.4 Modelos de interfaces de usuario abstractas para el módulo de entrenamiento.	74
6.2.3.4.1 Plan de pruebas.	75
6.2.3.4.2 Ejecución y reporte de pruebas.	75
6.2.3.4.3 Modelos de interfaz de usuario abstractos	77
6.2.4 Interfaz de usuario concreta -CUI.....	79
6.2.4.1 Modelo de relaciones intermodelos (Mapping).	79

6.2.4.2 Mapeo entre el modelo de tareas y las instancias del metamodelo de ejercicios.....	80
6.2.4.3 Selección de CIO.....	80
6.4.4.4 Parametrización de los widgets 3D	81
6.2.4.5 Reglas de transformación de AUI a CUI.	83
6.2.4.6 Transformación de modelos de interfaz de usuario abstractas a modelos de interfaz de usuario concretas.	86
6.2.4.7 Modelos de interfaces de usuario concretas para el módulo de entrenamiento.	86
6.2.4.7.1 Plan de pruebas.....	86
6.2.4.7.2 Ejecución y reporte de pruebas.	86
6.2.4.7.3 Modelos de interfaz de usuario concreta	88
7. Fase IV: Desarrollo e implementación	91
7.1 Creación de objetos rígidos.....	91
7.2 Streaming.....	91
7.3 Funcionamiento de Natnet.....	92
7.4 Tratamiento de datos.	93
7.5 Interfaz de Usuario.....	93
7.6 Escenas jugables.....	94
7.7 Modelos finales de la interfaz de usuario.	95
8. Fase V: Pruebas	97
8.1 Uso de CPU.....	97
8.2 Memoria.....	98
9. Conclusiones	100

10. Recomendaciones. 102

Referencias Bibliográficas..... 103

Lista de Tablas

Tabla 1. Cumplimiento de Objetivos.	22
Tabla 2. Definición de los tipos de tareas CTT	34
Tabla 3. Operadores temporales definidos en notación CTT.	34
Tabla 4. Facetas AIC.	36
Tabla 5. Objetos de interacción concretos.	37
Tabla 6. Descripción de la metodología fase I.	40
Tabla 7. Número de archivos arrojados por cada base de datos.	43
Tabla 8. Resumen de la ejecución de la ecuación.	44
Tabla 9. Análisis de las habilidades técnicas halladas	45
Tabla 10. Definición de habilidades técnicas.	46
Tabla 11. Reporte de respuestas entregados por los encuestados.	49
Tabla 12. Análisis de frecuencias.	51
Tabla 13. Descripción de la metodología de la fase II.	53
Tabla 14. Detalles ejercicio 1.	57
Tabla 15. Detalles ejercicio 2	58

Tabla 16. Detalles ejercicio 3.	58
Tabla 17. Detalles ejercicio 4	59
Tabla 18. Descripción de la metodología de la primera parte de la fase III.....	61
Tabla 19. Descripción de la metodología de la segunda parte de la fase III.....	63
Tabla 20. Establecimiento de iconos para los AIO.....	72
Tabla 21. Establecimiento de iconos para las facetas de los AIC.....	73
Tabla 22. Reporte de pruebas interfaz de usuario abstracta.	75
Tabla 23. Parámetros widgets label 3D.	81
Tabla 24. Parámetros widgets componente visual 3D.	82
Tabla 25. Parámetros widgets botón 3D	82
Tabla 26. Parámetros widgets panel 3D.....	83
Tabla 27. Reporte de errores subinterfaz de usuario concreta 1	87
Tabla 28. Reporte de errores subinterfaz de usuario concreta 2.....	87
Tabla 29. Reporte de errores subinterfaz de usuario concreta 3.....	87
Tabla 30. Reporte de errores subinterfaz de usuario concreta 4.....	87

Lista de Figuras

Figura 1 Estructura marco de referencia Cameleon.....	32
Figura 2 Modelos de presentación	35
Figura 3 Objetos de interacción abstractos.....	36
Figura 4 Pasos para el establecimiento de la ecuación de búsqueda	40
Figura 5 Agrupación de términos semejantes.....	41
Figura 6 Clasificación de la información	44
Figura 7 Frecuencia relativa vrs. grupos de habilidades	51
Figura 8 Ranking de habilidades técnicas	52
Figura 9 Composición del módulo de entrenamiento.	57
Figura 10 Etapas de diseño.....	60
Figura 11 Diagrama de casos de uso.....	62
Figura 12 Arquitectura del módulo de entrenamiento	63
Figura 13 Composición de los ejercicios	64
Figura 14 Modelo de tareas para la interfaz de usuario de los ejercicios.....	65
Figura 15 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 1.....	67

Figura 16 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 2.....	68
Figura 17 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 3.....	68
Figura 18 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 4.....	69
Figura 19 Modelo del dominio de los ejercicios.....	70
Figura 20 Categorías	71
Figura 21 Enfoque sistemático de transformación de modelos de tareas a AUI.....	74
Figura 22 Resultados prueba estática 1	76
Figura 23 Subinterfaz de usuario abstracta 4.....	77
Figura 24 Subinterfaz de usuario abstracta 4.....	78
Figura 25 Subinterfaz de usuario abstracta 3.....	78
Figura 26 Subinterfaz de usuario abstracta 4.....	79
Figura 27 Mapping.....	80
Figura 28 Mapeo entre los modelos de tareas y las instancias del modelo de dominio.....	81
Figura 29 Enfoque sistémico de transformación AUI a CUI	86
Figura 30 Resultado de prueba estática 2.....	88
Figura 31 Barras de calibración	91
Figura 32 Plano de calibración	91
Figura 33 Ejemplo de Streaming	92
Figura 34 Sistema de coordenadas.....	93

Figura 35 Objeto	94
Figura 36 Objetivo	94
Figura 37 Herramienta	95
Figura 38 Subinterfaz 1	95
Figura 39 Subinterfaz 2	96
Figura 40 Subinterfaz 3	96
Figura 41 Subinterfaz 4	96
Figura 42 Uso de CPU	97
Figura 43 Uso de CPU por porcentaje de uso.....	98
Figura 44 Uso de memoria.	99

Lista de Apéndices

**(Ver apéndices adjuntos en el CD y pueden visualizarlos en la Base de Datos de la
Biblioteca UIS)**

Apéndice A. Formato de encuesta

Apéndice B. Especificación de casos de uso

Apéndice C. Documento de especificación de requisitos

Apéndice D. Modelos de tareas

Apéndice E. Plan de pruebas estáticas

Apéndice F. Reporte de pruebas estáticas

Apéndice G. Interfaces de usuario concretas- gráficamente

Apéndice H. Plan de pruebas técnicas

Resumen

TÍTULO: DISEÑO DE UN MÓDULO DE ENTRENAMIENTO PARA UNA HABILIDAD TÉCNICA EN ESTUDIANTES DE ESPECIALIZACIÓN EN ORTOPEDIA Y TRAUMATOLOGÍA *

AUTOR: VERGARA BERNAL, Tania Estephany **

PALABRAS CLAVES: Habilidad técnica, coordinación mano- ojo, entrenamiento, realidad aumentada.

DESCRIPCIÓN:

La fijación transpedicular es una técnica que consiste en la introducción de tornillos a través de un punto ubicado en la vértebra, su incorrecta colocación puede fracturar el pedículo y por consiguiente se efectúa una ruptura del canal medular que conlleva a una fuga de líquido cefalorraquídeo. Esta técnica es ejecutada por un grupo de especialistas, liderados por un ortopedista o neurocirujano que posee un conjunto de habilidades técnicas. Las habilidades técnicas se desarrollan mediante la práctica y el entrenamiento, y por lo general involucran esfuerzos coordinados del cerebro y los músculos.

Por consiguiente, se orientó este trabajo a estudiar este grupo de habilidades técnicas mencionadas en la literatura, seleccionando la habilidad coordinación óculo-manual como una destreza indispensable al momento de realizar un proceso quirúrgico donde se requiera efectuar una fijación con tornillos en la columna vertebral. Se diseñó un módulo de entrenamiento; utilizando un enfoque basado en modelos para la creación de la interfaz de usuario, flexible a cambios y posible rediseño de nuevos ejercicios. Este módulo permite entrenar y evaluar la habilidad para reducir las acciones fallidas mediante el uso de realidad aumentada (RA). La RA permite superponer imágenes virtuales para crear escenarios simulados, que garantizan una experiencia guiada en tiempo real. El módulo propone de cuatro (4) ejercicios que conforman un circuito de entrenamiento y apoyo, donde residentes de ortopedia y/o estudiantes de medicina, pueden desarrollar la habilidad de coordinación óculo-manual.

*Trabajo de grado

**Facultad de Ingenierías Físico – Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: PhD Gabriel Rodrigo Pedraza Ferreira. Codirector: MSc. Luis Eduardo Bautista Rojas.

Abstract

Title: DESIGN OF A TRAINING MODULE FOR A TECHNICAL SKILL IN STUDENTS OF ORTHOPEDICS AND TRAUMATOLOGY SPECIALIZATION *

Author: VERGARA BERNAL, Tania Estephany **

Key words: Technical skills, hand-eye coordination, training, increased reality

Description:

Transpedicular fixation is a technique that involves the introduction of screws through a spot located in the vertebra, it's incorrect placement can fracture the pedicle and consequently can caused a rupture of the medullary canal, leading to a leak of cerebrospinal fluid. This technique is performed by a group of specialists, led by an orthopedist or neurosurgeon who owns a set of technical skills. Technical skills are developed through practice and training, and usually involve a coordinated work between the brain and the muscles.

Therefore, this work was oriented to study this group of technical skills mentioned in the literature, selecting the hand-eye coordination skill as an indispensable skill at the time of performing a surgical process where it is required to make a fixation with screws in the spine. A training module was designed; using a model-based approach for the creation of the user interface, flexible to changes and possible redesign of new exercises. This module allows to train and evaluate the ability to reduce failed actions through the use of increased reality (IR). The IR allows you to overlap virtual images to create simulated scenarios, which guarantee a guided experience in real time. The module proposes four (4) exercises that configue a training and support circuit, where residents of orthopedics and / or medical students can develop the eye-hand coordination skill.

*Thesis

**Faculty of Physics – Mechanic Engineering. School of Engineering and Computer Systems. Advisor: PhD Gabriel Rodrigo Pedraza Ferreira, Co-Advisor MSc. Luis Eduardo Bautista.

Introducción

El éxito de numerosos procesos quirúrgicos muchas veces se debe a la experticia del médico líder o médico cirujano, quien a lo largo de su vida a desarrollado unas habilidades técnicas que serán vitales para efectuar procedimientos que adquieren en su formación profesional para servir a los propósitos de la medicina. Según la revista colombiana de cirugía “para ser un buen cirujano se requiere de cualidades especiales como disciplina, creatividad, inteligencia, conocimiento profundo en anatomía, patología y sobre todo poseer una cualidad innata conocida como destreza manual, la cual no se aprende, sino que se perfecciona con la práctica” (Pestana Tirado, Moreno Ballesteros, & González Di Filippo, n.d.).

Sin embargo, el método de enseñanza- aprendizaje utilizado en las instituciones de educación superior actualmente estimula muy poco el desarrollo de habilidades como destreza manual, que hace parte de un grupo denominado habilidades técnicas, por el contrario, enfatiza en el conocimiento teórico o habilidades no técnicas incluyendo en sus planes de estudio actividades de experiencia donde se le permitan al estudiante practicar las veces que sea necesario un procedimiento.

Es así como aparece el concepto de simulador médico, una propuesta innovadora que se viene implementado a lo largo de la historia para sustituir la realidad por escenarios que finjan un fenómeno, en el cual, está permitido equivocarse para comprender y realimentar lo que hacen bien y mal. Estos escenarios varían dependiendo la habilidad técnica a entrenar y el enfoque que se le pretenda dar al relacionarla con una especialidad médica. De este modo, se puede afirmar que existe un gran número de especialidades médicas que contribuyen a la construcción de

simuladores, generando una educación tipo excluyente centrada en el conocimiento conforme a una especialidad; de esta forma los estudiantes no requieren conocer y/o comprender todas las anomalías presentes en el ser humano.

Para limitar el proyecto, nos enfocamos en los procedimientos de fijación transpedicular que está incluida en la especialidad quirúrgica de cirugía ortopédica y/o neurocirugía, y es la técnica más manipulada para tratar problemas degenerativos óseos de columna vertebral (Martínez-Poncede León et al., 2012), que radica en inmovilizar dos o varios segmentos de vertebras empleando tornillos que son conectados a barras metálicas. Este tipo de cirugía requiere un nivel de precisión muy alto ya que se pueden presentar complicaciones como infecciones, lesiones de vasos sanguíneos, invasión del canal medular, entre otras.

García Gonzáles et al., 2013. afirma que “la colocación errada de los tornillos según estudios se encuentra alrededor del 6% al 30%” pero estos errores se pueden disminuir incluyendo tecnología.

Por consiguiente, en el desarrollo de este proyecto se expone el diseño de un módulo de entrenamiento para la habilidad técnica coordinación mano- ojo u óculo- manual en procedimientos de fijación transpedicular en columna vertebral, que admita entrenar a estudiantes inscritos en los programas académicos de medicina y especializaciones médico quirúrgicas de la Universidad Industrial de Santander -UIS, haciendo uso de realidad aumentada, que como bien se sabe es una tecnología que permite la combinación de información digital y física media dispositivos tecnológicos (Rigueros Bello, 2017) y que ha sido usada en el entrenamiento de diversas habilidades, atestiguando que brindará ventajas positivas al módulo de entrenamiento puesto que será interactivo y en tiempo real permitiendo además la evaluación objetiva y no requerirá de un experto que lo guíe en el procedimiento del módulo.

Cumplimiento de Objetivos

Tabla 1.
Cumplimiento de Objetivos.

Objetivo	Cumplimiento	Pág.
Establecer la habilidad técnica indispensable para la colocación de los tornillos transpediculares en ambiente quirúrgico mediante revisión de literatura.	Capítulo	4
Aplicar un enfoque de desarrollo basado en modelos en el diseño de la herramienta.	Capítulo	6
Codificar un módulo de aprendizaje para adiestramiento de la habilidad técnica.	Capítulo	7
Diseñar y ejecutar un plan de pruebas técnicas.	Capítulo	8

Nota: Guía para encontrar dentro del documento el desarrollo de cada uno de los objetivos propuestos.

1. Objetivos

1.1 Objetivo General.

Diseñar una herramienta para el entrenamiento de una habilidad técnica en la colocación de tornillos transpediculares en procedimientos de Columna Vertebral.

1.2 Objetivos Específicos.

- Establecer la habilidad técnica indispensable para la colocación de los tornillos transpediculares en ambiente quirúrgico mediante revisión de literatura.
- Aplicar un enfoque de desarrollo basado en modelos en el diseño de la interfaz de usuario de la herramienta.
- Codificar un módulo de aprendizaje para adiestramiento de la habilidad técnica.
- Diseñar y ejecutar un plan de pruebas técnicas.

2. Generalidades del Proyecto

2.1 Justificación

La fijación transpedicular “consiste en la implantación de tornillos transpediculares dirigidos a través de un punto ubicado en la unión de la apófisis transversa con la apófisis articular superior” (Martínez-Poncede León et al., 2012), el cual atraviesa el pedículo para luego unirlos con barras; procedimiento empleada para la estabilización de la columna vertebral, que se efectúa en deformidades, hernias, tumores, entre otras.

Sin embargo, este proceso tiene un alto nivel de error que ha sido estudiado y registrado en diferentes estudios: Bergeson et al., 2008. demostraron que el “29% de los tornillos fijados por residentes ortopédicos, no estaban completamente dentro del pedículo, el 74% de estos procesos eran infracciones no críticas y el 26% era crítico”. Por otro lado, Wang, Chin, Lu, Smith, & Chou, 2010. afirmaron “nuestros resultados han demostrado que bajo supervisión, los tornillos pediculares torácicos colocados por residentes de neurocirugía tienen una tasa de violación cortical del 15%”, lo que demuestra que los estudiantes necesitan obtener experiencia práctica, que no puede ser adquirida por el método maestro-aprendiz, el cual propone que los pacientes sean herramientas de aprendizaje, donde se realicen procedimientos sin proporcionarle importancia al riesgo de cometer errores, que no se pueden justificar dentro de un proceso de formación (Centre Maribor & Merc, 2014).

Para disminuir estos errores, la medicina estableció una alianza con la tecnología, de la cual surgió la simulación. La simulación médica permite la enseñanza por medio de oportunidades

donde el estudiante puede realizar prácticas de destrezas mientras se familiariza con instrumentos y equipos, adquiriendo experiencias en el reconocimiento de problemas y el desarrollo de toma de decisiones, así como en el perfeccionamiento de técnicas y procedimientos que pueden presentarse en casos poco frecuentes (Carlos Serna-Ojeda, Borunda-Nava, Domínguez-Cherit, Médicas Nutrición Salvador Zubirán, & Domínguez Cherit, 2012), bajo la creación de situaciones hipotéticas similares a las que se podrían enfrentar; trayendo consigo ventajas en las cuales el estudiante “puede someterse a situaciones provocadas en un ambiente seguro donde el error está permitido, y es posible aprender de los errores sin dañar al paciente” (Lopez Araoz, 2014). Debido a que la simulación se desarrolla basada en la tecnología, se presentan dos circunstancias. La primera, es que los estudiantes o especialistas deben emigrar del país a centros de formación especializados que usen esta tecnología educativa y, por otro lado, también se presenta la importación de esta tecnología, donde los costos de adaptación son bastante altos y no se tiene el personal capacitado para ejercer el rol de formador (Sader, Yee, & Hodaie, 2017).

No obstante, universidades nacionales como la UIS están apuntando al segundo caso, donde se les garantiza a los estudiantes un espacio dentro del campus universitario, en el cual se simulan escenarios de atención en diferentes niveles de complicación sin arriesgar la vida de pacientes, y pueden desarrollar habilidades técnicas y no técnicas. Actualmente se dispone de 47 simuladores en el espacio denominado laboratorio de simulación clínica de la facultad de salud (CENTIC, n.d.), entre los que se destaca simuladores virtuales y simuladores de bajo y alto nivel tecnológico, obtenidos por medio de intermediarios de empresas internacionales que se dedican a comercializarlos para diferentes especialidades médicas y especialidades médico- quirúrgicas.

La motivación para desarrollar este proyecto, en primera instancia nace del interés personal que se ha tenido durante el proceso de formación académico, al cuestionar ¿Cómo se puede contribuir al país diseñando una tecnología que genere valor agregado? y en segunda instancia fue el resultado del proyecto de investigación financiado por Colciencias titulado “Sistema de orientación usando realidad aumentada para la perforación de túneles tibial y femoral en reconstrucción de ligamento cruzado anterior”, donde se halló que la precisión obtenida por este tipo de equipos, depende ampliamente de las habilidades del cirujano. A través de estas relaciones, se toma la decisión de investigar la existencia de procesos médicos que aún no cuentan con tecnologías que entrelacen el entrenamiento de habilidades técnicas y no técnicas en la universidad, para avalar que los próximos médicos cuenten con un adiestramiento que garantice la ejecución de procedimientos sin errores brindándole seguridad al paciente.

2.2 Planteamiento del Problema

Crear productos innovadores, como maniqués o aparatos compuestos por dispositivos informáticos y/o electrónicos denominados *simuladores médicos*, ha sido la finalidad de empresas como Medical-X y Symbionix Cop, entre otras; que se dedican a crearlos y posteriormente comercializarlos para solucionar las restricciones éticas impuestas en el entrenamiento quirúrgico en seres humanos y animales.

Sin embargo, se puede observar la existencia de carencias en temas correspondientes al costo de adquisición y mantenimiento de estos simuladores, que resultan de las diferencias existentes entre países.

Un claro ejemplo son los robots SpineAssist, y el AQRate que asisten al cirujano en fijaciones transpediculares en columna vertebral, a pesar de que estos no entrenan habilidades técnicas, y se restringe al conocimiento del procedimiento, su costo supera los 100.000 dólares (Muñoz Rodríguez, Albán, & Muñoz Rodríguez, n.d.), lo cual hace que sea imposible su compra por universidades u hospitales de enseñanza del país.

El más reciente documento de la UIS en cifras, presenta a la comunidad las cifras más relevantes durante el periodo del 2016; donde se evidencia que cerca de “700 personas se están matriculando por semestre al programa de medicina y 60 personas al programa de especialización medico quirúrgica” (Sierra Bueno, Yánez Canal, & Monroy Allado, 2016b), donde cada uno de estos estudiantes recibe diariamente un conjunto de saberes combinados con espacios de práctica, que se adoptan a esta sociedad que demanda un gran número de anomalías; y se “están egresando 102 médicos y 54 especialistas” (Sierra Bueno, Yánez Canal, & Monroy Allado, 2016a), a pesar, de no contar con un dispositivo que entrene habilidades técnicas y no técnicas para el procedimiento de fijación transpedicular en la UIS o universidades que estén ubicadas en su área aledaña, es decir, actualmente se están graduando médicos cirujanos que jamás realizaron este procedimiento en sus años de formación profesional, pero según el otorgamiento de su título cuenta con el conocimiento y la experiencia para efectuarlo en la sociedad colombiana. Esto es preocupante ya que no les están brindando oportunidades donde puedan perfeccionar las habilidades técnicas y/o no técnicas en la ejecución de este procedimiento médico donde el error puede causar daños irreparables.

Por ello, surge la siguiente pregunta: ¿Cómo se podría crear un módulo de entrenamiento para la habilidad técnica coordinación mano- ojo para estudiantes de medicina y especialización ortopedia?

3. Marco Referencial

Se enunciarán los conceptos claves para construir el diseño del módulo de entrenamiento para la habilidad técnica coordinación óculo-manual que ayudarán a la comprensión del documento.

3.1 Enfoque de desarrollo basado en modelos.

Uno de los aspectos que dificulta el desarrollo de software es la amplia separación conceptual existente entre el dominio del problema y el de la implementación (Frankel, 2004) para ayudar a disminuir esta dificultad, han surgido varias opciones, entre las cuales están, MDE, MDSD, MBUIDE.

3.1.1 Ingeniería dirigida por modelos - MDE (en inglés Model-Driven Engineering). Es el término que se emplea para “describir los enfoques de desarrollo de software, partiendo de la definición de modelos abstractos, hasta lograr implementaciones concretas al aplicarles transformaciones sistemáticas” (France & Rumpe, 2007), sus principales características son, “fomentar el papel de los modelos y las actividades de modelado en las etapas del proceso de desarrollo y aumentar el nivel de automatización, mediante la construcción de herramientas de soporte, en cualquier actividad relacionada con el desarrollo” (Bollati, Leone, Gonnet, & Cuenca Pletsch, n.d.).

En la actualidad, “MDE se focaliza especialmente en el desarrollo de técnicas, métodos, procesos y herramientas de soporte que reduzcan efectivamente la brecha existente entre el problema y la implementación” (Frankel, 2004).

3.1.2 Desarrollo de Software Dirigido por Modelos -MDSO (en inglés Model Driven Software Development). El enfoque trazado por MDE al punto de vista de la ingeniería de Software se conoce como desarrollo de software dirigido por modelos -MDSO, el cual “cubre aspectos propios de esta ingeniería mediante la especificación de técnicas, métodos, herramientas y lenguajes obteniendo una separación conceptual entre el espacio de modelado y el espacio de implementación” (Giraldo Orozco, 2010). El objetivo principal de MDSO es la “construcción de herramientas prácticas y los conceptos que conduzcan a la automatización del proceso de software por medio de transformaciones de modelos” (Chaparro Lemus & Gómez Estupiñan, 2012).

Este enfoque según Giraldo Orozco, 2010. pretende apoyar a los desarrolladores en algunas funciones técnicas que son generadoras de errores, pues se supone que al aplicar sistemáticamente MDSO, la calidad de los sistemas de software, el grado de reutilización y, por tanto, implícitamente, la eficiencia de desarrollo, va a mejorar.

Mientras que MDSO se ocupa del uso de los modelos para el desarrollo de software, la arquitectura dirigida por modelos -MDA se enfoca en definir los estándares para desarrollar software haciendo uso intensivo de modelos. Se puede decir que MDA es la estandarización de MDSO (Chaparro Lemus & Gómez Estupiñan, 2012).

Pero, ¿Qué es MDA? La arquitectura dirigida por modelos - *MDA* (en inglés Model driven architecture) son un conjunto de estándares desarrollados por el grupo de objetos -OMG (en

inglés Object Management Group) que describen la generación de sistemas de software ejecutable a partir de modelos formales, que cumple con tres aspectos importantes en la construcción de software, utilización de modelos en todo el proceso de desarrollo, definición de un modelo para cada nivel de abstracción y transformaciones entre modelos (Chaparro Lemus & Gómez Estupiñan, 2012).

MDA propone una separación de modelos del sistema en tres capas, el modelo independiente de la computación (CIM), el modelo independiente de la plataforma (PIM) y el modelo específico de la plataforma (PSM) (Pastor, Aquino, España, & Panach, 2008).

3.1.3 Desarrollo de Interfaces de Usuario Basado en Modelos (MBUIDE). El desarrollo de interfaces de usuario basado en modelos MBUIDE (en inglés Model-Based User Interface Development Environment), “se basa en el modelo de desarrollo de software dirigido por modelos MDSD para el desarrollo de la interfaz de usuario” (Giraldo Orozco, 2010). Consiste en la especificación de la interfaz de usuario manipulando modelos declarativos que describen las distintas facetas y artefactos involucrados en el desarrollo de la misma, estos modelos declarativos son información contextual, que proporciona los requisitos a tener en cuenta en la generación de UI.

MBUIDE busca aumentar el nivel de abstracción usado en el diseño, permitiendo una generación automática parcial o total de la UI (López Jaquero, 2005).

La propiedad que caracteriza este enfoque es la especificación de la UI manejando continuos niveles de abstracción, partiendo de un alto nivel de abstracción por parte del diseñador, trabajando únicamente con descripciones lógicas, y dejando que los detalles de implementación

sean proporcionados por el sistema, haciendo posible, además, una generación total o parcial de la UI de forma sencilla cuando los requisitos cambian. (Muñoz Arteaga et al., 2014).

Varios estudios coinciden en recomendar tres niveles de abstracción: el modelo de tareas donde se ubican las actividades a efectuar por parte del usuario, el nivel abstracto, que representa la interfaz de usuario independiente de la modalidad y de la plataforma y un nivel concreto, que genera una interfaz de usuario dependiente de la modalidad (Balme, Demeure, Barralon, Coutaz, & Calvary, 2004) .

3.1.3.1 Descripción de interfaces de usuario basada en modelos según el marco de referencia Cameleon. CAMELEON (en inglés Context Aware Modelling for Enabling and Leveraging Effective interactiON) es un proyecto financiado por la Unión Europea que tiene por objetivo crear métodos y entornos que respalden el diseño y desarrollo de sistemas de software interactivos para que sean útiles (“Página de inicio del proyecto CAMELEON,” n.d.); su marco de referencia fue el resultado de dos principios claves, un enfoque basado en modelos y una cobertura de las fases de diseño y tiempo de ejecución de interfaces que admiten múltiples objetivo o múltiples contextos de uso (Calvary et al., 2002), el cual estructura el ciclo de vida del desarrollo en cuatro niveles de abstracción: tareas y conceptos, interfaz de usuario abstracta, interfaz de usuario concreta e interfaz de usuario final (Calvary et al., 2003).

- Nivel superior de abstracción (tareas y conceptos): Contiene modelos de tareas y modelos de dominio.
- *Nivel abstracto (UI abstracta)*: Es un modelo de presentación independiente, que utiliza contenedores y componentes abstractos para limitar las tareas definidas.



Figura 1 Estructura marco de referencia Cameleon- tomada de (Muñoz Arteaga et al., 2014)

- *Nivel concreto (UI concreta)*: Contiene modelos de presentación con un nivel de abstracción bajo, donde se especifica la definición abstracta mediante widgets.
- *Nivel operativo (UI final)*: Representa el código fuente de la interfaz de usuario que se puede compilar, teniendo en cuenta la plataforma donde se adaptara y los recursos de software que se manejaran.

3.1.3.1.1 Modelado de tareas. El modelado de tareas es un proceso que se debe efectuar para avalar que las aplicaciones estén centradas en los usuarios, usando un método de análisis de tareas que permite describir las actividades que hace un usuario al interactuar con una interfaz, que al finalizar establece un modelo de tareas.

Para obtener un modelo se debe llevar a cabo un proceso de análisis de tareas efectuando un método exclusivo de este proceso, que además incluye actividades de obtención de información, tales como, entrevistas, simulaciones, observaciones, etc. para percibir, analizar e identificar las actividades que realiza un usuario para alcanzar un determinado objetivo y posteriormente modelarlo por medio de técnicas; los métodos identificados por Muñoz Arteaga et al., 2014 son:

- **Métodos de competencia o cognitivos.** Estos métodos identifican secuencias de comportamiento correctas, representando el tipo de conocimiento que debe poseer un usuario acerca del uso del sistema.
- **Métodos predictivos para la evaluación del rendimiento humano.** Describen secuencias de comportamiento y el conocimiento que necesita el usuario para su ejecución.
- **Métodos descriptivos.** Permiten obtener una descripción más o menos completa del sistema a partir de la información obtenida de las tareas, como, por ejemplo: CCT (Concur-

TaskTrees)

A continuación, se describe la notación a usar en el proyecto de investigación para construir los modelos de tareas.

a) Concurtasktrees (CTT)





CTT es una notación desarrollada por Fabio Paternó empleada para realizar modelos de tareas, sus principales características son:

- Sintaxis gráfica, lo cual facilita su interpretación.
- Estructura jerárquica en forma de árbol, que permite tener diferentes niveles de abstracción.
- Un conjunto de operadores que ayudan a expresar la relación existente entre las tareas.

Según la notación CTT, las tareas se declaran con iconos y se clasifican en 4 tipos en función del actor que las lleva a cabo como se definen en la tabla 2; al crear una tarea se debe definir su identificador, nombre y categoría.

Tabla 2.

Definición de los tipos de tareas CTT

Tarea	Clasificación de la tarea	Descripción
Tarea de usuario		Tareas realizadas completamente por el usuario. Generalmente son tareas cognitivas o físicas.
Tarea de sistema o aplicación		Tareas realizadas por la aplicación, requieren información interna del sistema o produce información hacia el usuario.
Tarea de interacción		Son tareas que realiza el usuario interactuando con la aplicación por medio de alguna interacción.
Tarea abstracta		Son tareas complejas que se descomponen en un conjunto de subtareas.

Nota: Adopción (Cuppens, Raymaekers, & Coninx, n.d.)

Las relaciones temporales existentes entre las tareas se expresan mediante un conjunto amplio de operadores existentes en LOTOS que se ha definido en la tabla 3.

Tabla 3.

Operadores temporales definidos en notación CTT.

Operador temporal	Notación	Descripción
Activar	$T1 \gg T2$	Cuando termina la T1 se activa la T2. Las dos tareas se realizan de forma secuencial
Activar con paso de información	$T1 [] \gg T2$	La tarea T1 proporciona información a la tarea T2 antes de ser activada.
Elección	$T1 [] T2$	Selección alternativa entre dos tareas. Una vez que se está realizando una de ellas la otra no se encuentra disponible hasta que termine la que está activa.
Desactivación	$T1 [> T2$	Desactivar. La primera tarea es desactivada cuando comienza la ejecución de la segunda.
Independencia de orden	$T1 = T2$	Ambas tareas pueden ser realizadas, pero una vez comenzada una debe finalizar antes de que comience con la otra
Iteración	$T1^*$	La tarea T1 se realiza de forma repetitiva. Se estará realizando hasta que otra tarea la desactive.
Tarea opcional	$[T1]$	La ejecución de la tarea es opcional.
Entrelazado	$T1 T2$	Las acciones de las dos tareas pueden realizarse en cualquier orden.
Sincronización	$T1 [] T2$	Las dos tareas tienen que sincronizarse en alguna de sus acciones para intercambiar información.
Suspender- Reanudar	$T1 > T2$	T2 tiene la posibilidad de interrumpir a T1 que podrá ser retomada cuando aquella finalice.

Nota: Adopción de la notación CTT (Nóbrega, Nunes, & Coelho, 2006)

3.1.3.1.2 Modelos de presentación. Contienen una descripción de la interfaz con la que el usuario interactuará. El contenido de los modelos de presentación depende de los niveles de abstracción definidos. En la figura 2, se especifican los modelos de presentación según el marco de referencia CAMELEON

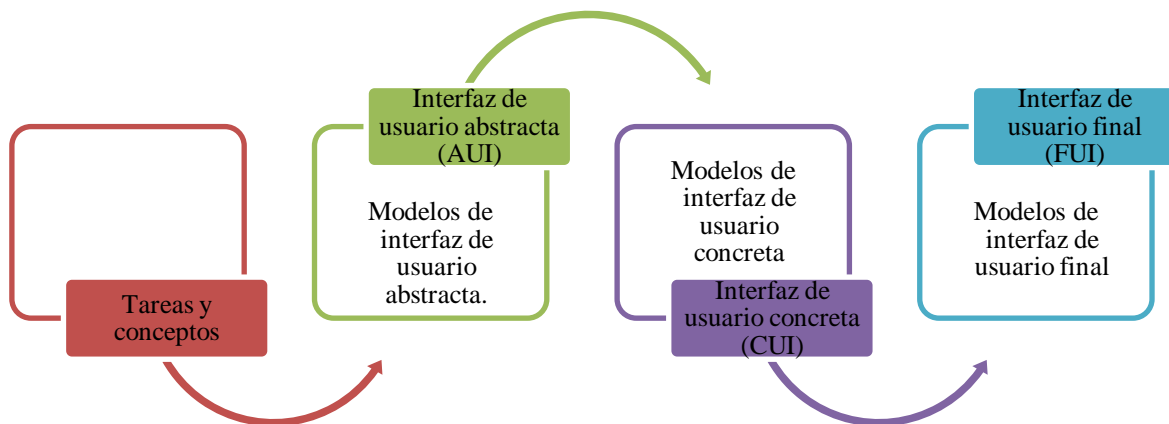


Figura 2 Modelos de presentación

- **Modelos de interfaz de usuario abstracta.** La Interfaz de usuario abstracta -AUI, es un modelo de presentación de UI usando un nivel de abstracción alto, generando detalles independientes de la plataforma donde se implementará.

La AUI se describe empleando objetos de interacción abstractos -AIO (en inglés Abstrac Interaction Object) que pueden ser de dos tipos: componentes de interacción abstractos -AIC (en inglés Abstract Individual Component) y contenedores abstractos -AC (en inglés Abstract Containers), figura 3 (Montero, López-Jaquero, Vanderdonckt, González, & Lozano, n.d.).

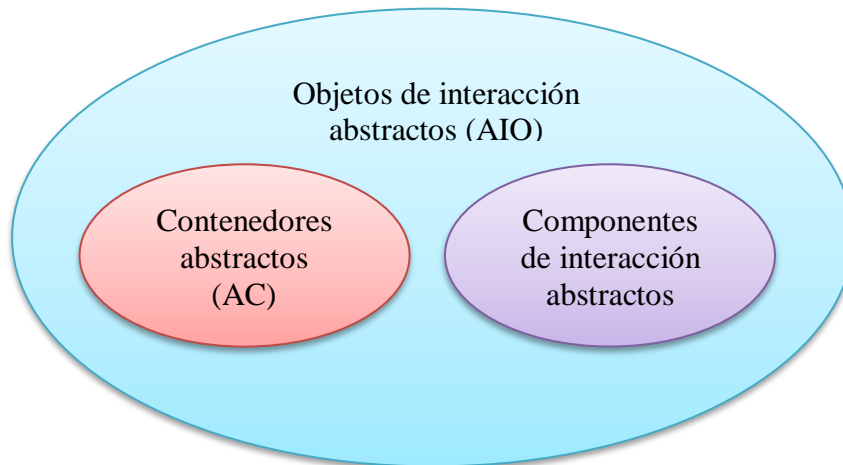


Figura 3 Objetos de interacción abstractos

UsiXML(en inglés USer Interface eXtensible Markup Language) es un lenguaje marcado compatible con XML que describe interfaces de usuario para múltiples contextos de uso (“UsiXML - Lenguaje de marcado extendido de la interfaz del usuario,” n.d.) proporcionando un conjunto reducido de elementos que permiten la descripción de interfaces de usuario abstractas en plataformas y modalidades independientes(Montero & López-Jaquero, 2007) . Los objetos abstractos que utiliza son AC y AIC, donde, un AC es una entidad que permite una agrupación lógica de otros contenedores abstractos o componentes individuales abstractos.

Un AIC es una abstracción que permiten la descripción de objetos interactivos de manera independientes de la particularidad por la que son proporcionados en el mundo físico y puede estar compuesto de múltiples facetas (Muñoz Márquez, 2007), sus cuatro facetas (tabla 4) principales son: entrada, salida, navegación y control, cada una describe una función particular (Montero et al., n.d.).

Tabla 4.
Facetas AIC.

Faceta	Descripción
Entrada	Describe que existe una acción de entrada por parte del usuario que es soportada por un AIC.

Salida	Describe cualquier tipo de información (textual, auditiva, grafica, etc.) que se le imprime o muestra al usuario por medio de un AIC.
Navegación	Es La posible transición de contenedor por medio de un AIC
Control	Describe los enlaces existentes entre un AIC y funciones del sistema.

- **Modelos de interfaz de usuario concreta.** La interfaz de usuario concreta -CUI representa un nivel de abstracción menor y es independiente de la plataforma final donde se implementará. Se compone de objetos de interacción concretos CIO (en inglés Concrete Interaction Object) y relaciones concretas. Un CIO se define como una entidad que los usuarios pueden percibir y/o manipular. (Montero et al., n.d.)

En la tabla 5 se muestran algunos componentes individuales concretos encargados de dar forma a la especificación de la CUI en UsiXML y otros entornos creados para la elaboración de transformaciones de interfaces.

Tabla 5.
Objetos de interacción concretos.

Componente	Descripción
Panel, window	Típicamente este contenedor contendrá otros componentes de interacción como 'box', 'tables', button, listbox, etc
Box	Hace la función de contenedor de componentes gráficos.
Button	Es un componente de interacción que representa un elemento cliqueable.
RadioButton	Botón de radio, que describen un conjunto de opciones relacionadas.
ComboBox	Múltiples opciones desplegables en un cuadro.
CheckBox	Casilla de marcado.
ListBox	Lista con diversas opciones a seleccionar, con la propiedad 'multiple_selection' se permite seleccionar varias opciones.
Input text	Componente de introducción de texto por parte del usuario.
Table	Componente que representa tablas.
Label	Se encarga de mostrar texto que el usuario no puede modificar

ScrollBar	Componente mediante el cual se puede deslizar la interfaz de usuario hacia arriba o abajo.
Barra de menú	Alberga menús y elementos de menú
Caja de dialogo	Es un cuadro independiente que incluye de forma predeterminada un control de confirmación o cancelación.
Caja de diálogo con pestañas	Puede ser vista como un conjunto de ventanas apiladas cuyo acceso viene dado por un conjunto de pestañas.

Nota: Adopción de los CIO hallados en la literatura. (Vanderdonckt & Bodart, 1993) (Muñoz Márquez, 2007) (Giraldo Orozco, 2010).

3.2 Realidad aumentada

El concepto de realidad aumentada (en inglés Augmented Reality o AR) agrupa aquellas tecnologías que permiten la superposición, en tiempo real, de imágenes, marcadores o información generados virtualmente, sobre imágenes del mundo real. Se crea de esta manera un entorno en el que la información y los objetivos virtuales se fusionan con los objetos reales ofreciendo una apariencia tal para el usuario que puede llegar a pensar que forma parte de su realidad cotidiana olvidando incluso la tecnología que le da soporte. (“Realidad Aumentada : una nueva lente para ver el mundo,” n.d.)

3.3 Herramientas

3.3.1 Sistema optitrack. El sistema optitrack es un conjunto de cámaras y marcadores reflectivos, los cuales son rastreados por las cámaras y generan un modelo o esqueleto virtual el cual animado en tiempo real por los marcadores se mueve. (“OptiTrack - Motion Capture Systems,” n.d.).

3.3.2 Unity. Es un motor de desarrollo para implementar lógica de juegos y ejecución de alto rendimiento, que contiene un editor todo en uno, capaz de soportar el desarrollo 2D y 3D para necesidades específicas que el usuario requiera, además cuenta con un sistema que le permite crear interfaces de usuario intuitivas, así mismo permite la personificación de extensiones para cumplir con el objetivo que se busca. (“Unity - Download Archive,” n.d.).

3.3.3 Natnet. Es un SDK que contine librerías para enviar y recibir (cliente/servidor) información de NaturalPoint a través de la red. Natnet usa el protocolo UDP en conjunto con Ponit-To- Point unicast o IP Multicast para enviar información. Natnet empaqueta los datos de los objetos rígidos en formato XML y debido a su sencillez las librerías pueden fácilmente utilizarlas en cualquier lenguaje de programación. (“NatNet SDK 3.0 - NaturalPoint Product Documentation Ver 2.0,” n.d.).

4. Fase I: Establecer la habilidad técnica indispensable para la colocación de los tornillos transpediculares.

En esta fase principalmente se analizó en la literatura científica, las habilidades técnicas que un residente, médico especialista en ortopedia debe poseer; indicando sus nombres y definiciones que fueron determinantes para la elaboración de un instrumento que permitió la recolección de información y posterior análisis del nivel de importancia de entrenamiento de cada habilidad en residentes antes de enfrentarse en el quirófano a realizar una fijación transpedicular en el área de Bucaramanga, seleccionando la más importante. En la tabla 6, se presenta la descripción detalla de la metodología que se implementó para alcanzar el objetivo.

Tabla 6.

Descripción de la metodología fase I.

Actividades realizadas	Revisión detallada de las habilidades técnicas, a través del diseño y ejecución de una ecuación de búsqueda.	Investigación del grado de importancia de entrenamiento de cada una de las habilidades técnicas encontradas en la literatura.	Selección de la habilidad técnica a entrenar.
Resultado	-Ecuación de búsqueda. -Listado de habilidades técnicas.	-Formato de encuesta. -Análisis estadístico de la información obtenida.	-Nombre y descripción de la habilidad técnica seleccionada.

4.1 Diseño de la ecuación de búsqueda

Para llevar a cabo la exploración literaria de habilidades técnicas, fue necesario establecer una ecuación de búsqueda con la finalidad de encontrar información que proveerá un listado de habilidades técnicas que deben poseer los cirujanos ortopedistas, haciendo uso de operadores booleanos. Esta ecuación se obtuvo implementando 4 pasos definidos en la figura 4.

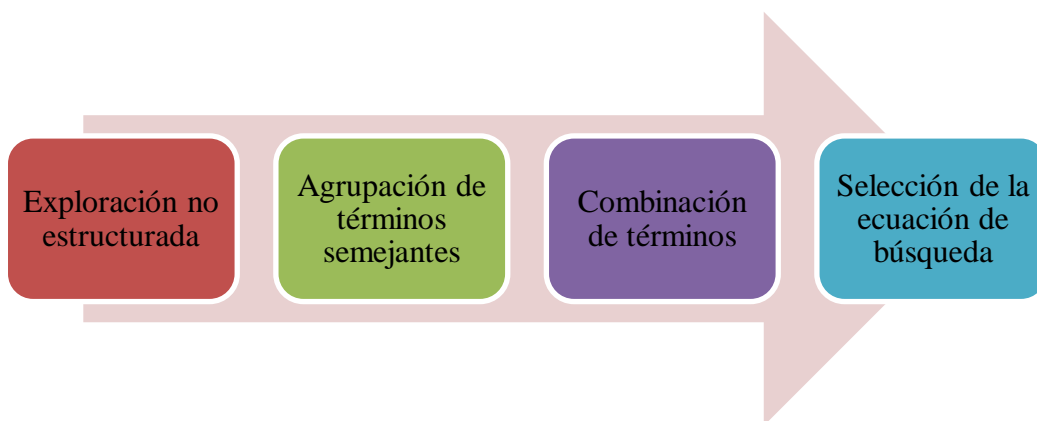


Figura 4 Pasos para el establecimiento de la ecuación de búsqueda

4.1.1 Exploración no estructurada. Se especificó un listado de palabras o términos relacionadas con habilidades técnicas en cirugía ortopédica. Para ello se definieron los siguientes términos: resident, virtual reality, simulation, skills, training, education, orthopedics, surgical,

surgeons, augmented reality, technical skill, trainee, teach, surgery, ability, learning, motor skills, physician.

4.1.2 Agrupación de términos semejantes. Especificado el listado, se procede hacer una agrupación de términos semejantes según su significado figura 5.

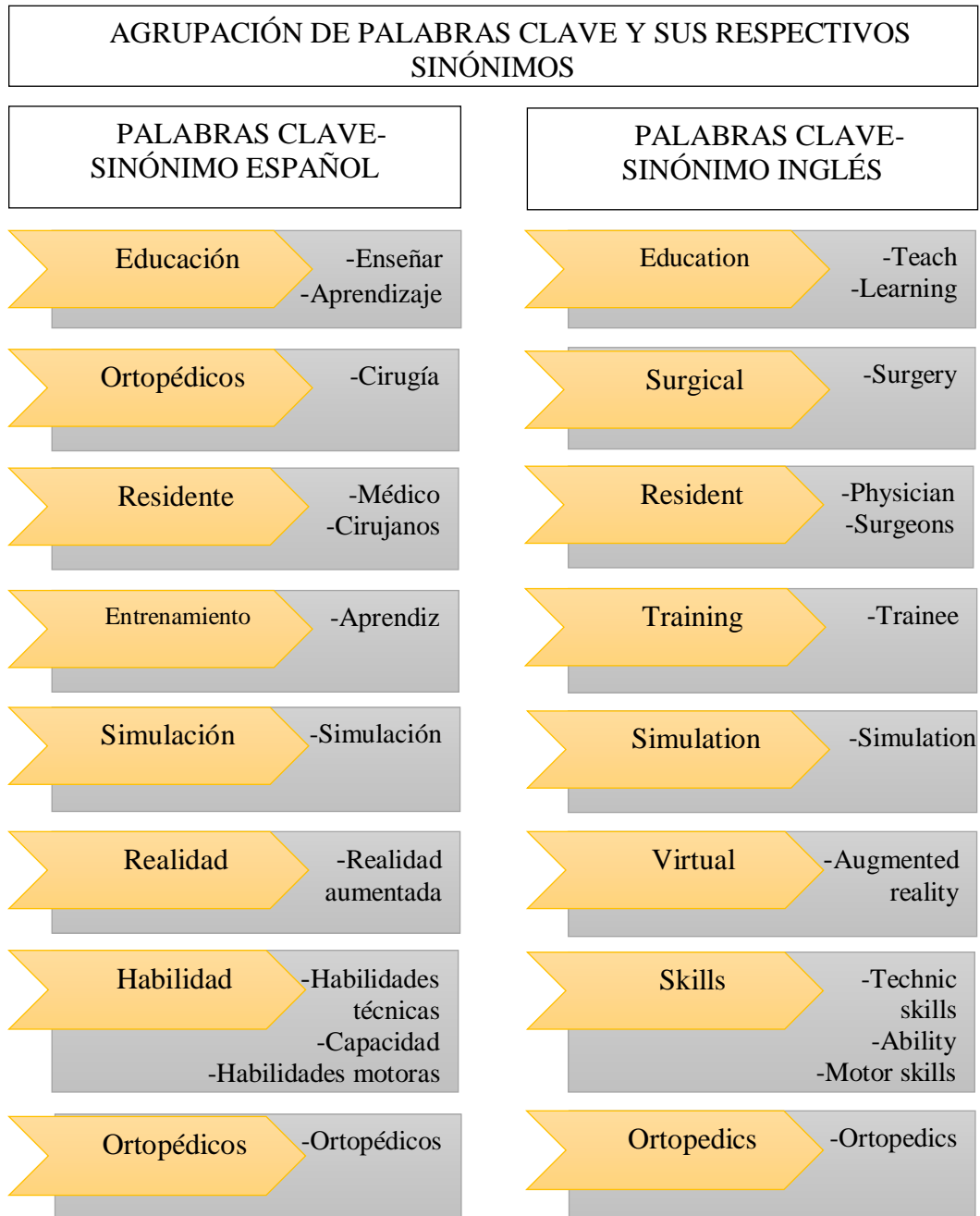


Figura 5 Agrupación de términos semejantes

4.1.3 Combinación de términos. Esta combinación se obtiene incluyendo los operadores booleanos AND, OR y NOT de la siguiente manera:

Para agrupar las palabras claves y los sinónimos se utilizó el conector OR, formando 8 grupos:

1. (Education OR teach OR learning)
2. (Orthopedics)
3. (resident OR surgeons OR physician)
4. (Simulation)
5. (skills OR technical skills OR ability OR motor skills)
6. (surgical OR surgery)
7. (Training OR trainee)
8. (Virtual reality OR Augmented reality)

Para entrelazar grupos se manipuló el conector AND, creando $P_8 = 8!$ que corresponde a $8 * 7 * 6 * 5 * 4 * 3 * 2 * 1 = 40.320$ posibles combinaciones, de las cuales se mostrarán 3.

- a) (resident OR surgeons OR physician) AND (skills OR technical skills OR ability OR motor skills) AND (Education OR teach OR learning) AND (surgical OR surgery) AND (Orthopedics) AND (Training OR trainee) AND (Simulation) AND (Virtual reality OR Augmented reality).
- b) (resident OR surgeons OR physician) AND (Virtual reality OR Augmented reality) AND (Simulation) AND (skills OR technical skills OR ability OR motor skills) AND (Training OR trainee) AND (Education OR teach OR learning) AND (Orthopedics) AND (surgical OR surgery).

- c) (surgical OR surgery) AND (Orthopedics) AND (Education OR teach OR learning) AND (Training OR trainee) AND (skills OR technical skills OR ability OR motor skills) AND (Simulation) AND (Virtual reality OR Augmented reality) AND (resident OR surgeons OR physician).

4.1.4 Selección de la ecuación de búsqueda. La ecuación seleccionada fue: (resident OR surgeons OR physician) AND (skills OR technical skills OR ability OR motor skills) AND (Education OR teach OR learning) AND (surgical OR surgery) AND (Orthopedics) AND (Training OR trainee) AND (Simulation) AND (Virtual reality OR Augmented reality), ya que arrojó el menor número de trabajos relacionados con habilidades técnicas.

4.2 Ejecución de la ecuación de búsqueda.

Haciendo uso de los recursos electrónicos que provee la universidad a través de la biblioteca, se procedió a ejecutar la ecuación de búsqueda en 4 bases de datos conocidas a nivel mundial, filtrando los documentos publicados desde el año 2008 al 2018, arrojando un número considerable de trabajos para estudiar documentados en la tabla 7.

Tabla 7.
Número de archivos arrojados por cada base de datos.

Base de datos	Número de artículos encontrados
Springer	233
IEEE	69
Scopus	636
ScienceDirect	1.370

Finalmente se llevó a cabo una inspección básica, estudiando y analizando los resultados de cada una de las bases de datos inspeccionando los títulos y verificando si satisfacía la información necesaria y se decidió trabar con los documentos entregados en la base de datos SPRINGER. Posteriormente se colocó en marcha la abstracción de los artículos que generaban valor agregado a nuestra investigación siguiendo la selección indica en la tabla 8.

Tabla 8.

Resumen de la ejecución de la ecuación.

Selección por ecuación de búsqueda	233 (incluyendo capítulos de libros, artículos, resúmenes conferencias)
Selección por lectura de títulos	16
Selección por resumen	8
Selección por conclusiones	6
Total, leídos	10

En conclusión 10 artículos fueron leídos, analizados e interpretados para extraer la información necesaria.

4.3 Recolección de información

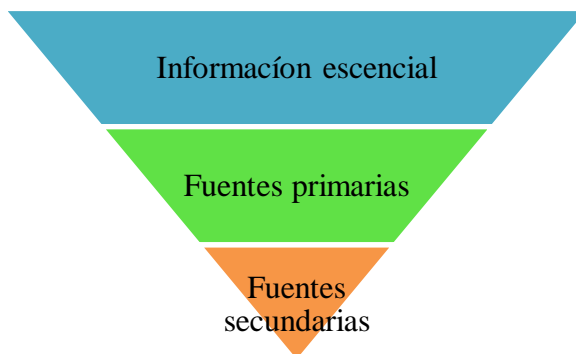


Figura 6 Clasificación de la información

4.3.1 Información esencial. Constituye una síntesis del listado de habilidades técnicas, tomadas de la ejecución de la ecuación de búsqueda.

La información mostrada en la tabla 9, corresponde al sumario de los artículos leídos y otras fuentes sacadas de las referencias, donde se definen los nombres de las habilidades técnicas, nombradas en investigaciones hechas por los respectivos autores.

Tabla 9.

Análisis de las habilidades técnicas halladas

Autor	Nombre	Listado de habilidades técnicas
(Safir et al., 2013)	What skills should simulation training in arthroscopy teach residents?	<ul style="list-style-type: none"> - Destreza manual - Percepción de profundidad - Percepción espacial - Sensación táctil - Triangulación
(Safir et al., 2013)	What skills should simulation training in arthroscopy teach residents? A focus on resident input.	<ul style="list-style-type: none"> - Destreza manual - Percepción espacial - Sensación táctil - Triangulación
(Stobbs & Kumar, 2014)	The role of simulation in developing surgical skills.	<ul style="list-style-type: none"> - Triangulación
(Bergamasca, Bardy, & Gopher, 2013)	Skill training in multimodal virtual environments.	<ul style="list-style-type: none"> - Acoplamiento proximal-distal - Acoplamiento respiratorio/movimiento - Balance y control de la postura - Control de flexibilidad y la gestión de la atención - Control de fuerza fina - Control prospectivo - Coordinación bimanual - Coordinación interpersonal - Coordinación mano- ojo - Estrategias de afrontamiento y esquemas de respuesta alternativas - Organización de la memoria, estructura y desarrollo de esquemas de conocimiento - Percepción observacional - Percepción por tacto

(Henrique & Karahan, 2017)	Motor skills training in orthopedic sports medicine.	<ul style="list-style-type: none"> - Aguante - Control de clasificación - Coordinación de múltiples miembros - Coordinación gruesa del cuerpo - Destreza de los dedos - Destreza manual - Equilibrio bruto del cuerpo - Firmeza del brazo - Flexibilidad de extensión - Flexibilidad dinámica - Fuerza del tronco - Fuerza dinámica - Fuerza estática - Fuerza explosiva - Orientación de respuesta - Precisión de control - Puntería - Tiempo de reacción - Velocidad de movimiento de la extremidad - Velocidad de muñeca
(Varas et al., 2012)	Significant transfer of surgical skills obtained with an advanced laparoscopic training program to a laparoscopic jejunostomy in a live porcine model: feasibility of learning advanced laparoscopy in a general surgery residency.	
(Karahan & Tuijthof, n.d.)	Arthroscopic Simulation in Skills Training: European Initiatives.	
(Stirling, Lewis, & Ferran, 2014)	Surgical skills simulation in trauma and orthopaedic training.	

Tabla 10.
Definición de habilidades técnicas.

Acrónimo	Nombre de habilidad	Definición
PDC	Acoplamiento proximal-distal	Se refiere a la organización del cuerpo, movimientos subyacentes del brazo, o la correlación entre las posturas del brazo y los movimientos de la mano.
RMC	Acoplamiento respiratorio/movimiento	La sincronización de la respiración y el movimiento (segmentos, músculos, articulaciones, etc.).
	Aguante	Resistencia cardiovascular o capacidad de mantener el esfuerzo.
	Anticipación	Suceso de anticipar eventos rápidamente y responder en consecuencia.

	Control de clasificación	Posibilidad de hacer ajustes continuos de velocidad y dirección con precisión al rastrear.
FFC	Control de fuerza fina	La regulación de las fuerzas internas aplicadas en la superficie para alcanzar con éxito la meta (perforación, pegado, navegación, etc.).
PC	Control prospectivo	La anticipación de un futuro lugar de contacto y tiempo basado en información espacio-temporal.
BC	Coordinación bimanual	La sincronización funcional en el espacio y el tiempo de los brazos / manos / dedos.
	Coordinación de múltiples miembros	Capacidad de coordinar numerosos miembros.
	Coordinación gruesa del cuerpo	Capacidad de coordinar numerosos movimientos simultáneamente mientras el cuerpo está en movimiento.
IPC	Coordinación interpersonal	El acoplamiento entre dos o más personas. Surge de un enlace de componentes que incluyen la sociabilidad, los principios motores y las restricciones de neurociencia.
HEC	Coordinación óculo-manual	La sincronización del ojo / mano con referencia a la información detectada por el sentido de la vista.
	Destreza de los dedos	Capacidad de controlar manipulaciones de objetos pequeños principalmente mediante el uso de dedos.
	Destreza manual	Capacidad que tiene para producir, manipular, transformar objetos con las manos.
	Equilibrio bruto del cuerpo	Capacidad de mantener el equilibrio sin señales visuales.
CSRS	Estrategias de afrontamiento y esquemas de respuesta	Importancia o peso de atención calculado sobre las subelementos de una tarea que están asociados al logro de un objetivo.
	Exactitud	Habilidad ejecutada con precisión.
	Firmeza del brazo	Posibilidad de realizar movimientos precisos de posicionamiento de manos y brazos donde la participación de la fuerza y la velocidad son mínimas.
	Flexibilidad de extensión	Posibilidad de mover los músculos del tronco y la espalda a través de un rango de movimiento.
	Flexibilidad dinámica	Capacidad de hacer movimientos de flexión repetidos y rápidos.
	Fuerza del tronco	Fuerza dinámica de los músculos del tronco.
	Fuerza dinámica	Resistencia muscular o capacidad de ejercer fuerza repetidamente.
	Fuerza estática	Capacidad de generar la fuerza máxima contra el objeto externo.
	Fuerza explosiva	Habilidad para crear el máximo esfuerzo combinando fuerza y velocidad.
MO	Organización de la memoria, estructura y desarrollo de esquemas del conocimiento	Nivel de memoria multi jerarquizada y específica de la tarea formulada y organizada.
	Orientación de respuesta	Posibilidad de seleccionar rápidamente una respuesta de varias alternativas.
PO	Percepción observacional	Es la capacidad de detectar, muestrear y extraer información relevante para la tarea.
PbT	Percepción por tacto	El componente coetáneo (que existe al mismo tiempo que otra cosa) de la modalidad háptica.
	Precisión de control	Posibilidad de hacer ajustes de movimiento altamente controlados, especialmente aquellos que involucran grupos musculares grandes.
	Puntería	Capacidad de dirigir movimientos con la mano de forma rápida y precisa en un objeto pequeño.
	Sincronización	Tiempo preciso y orden secuencial correcto.
	Tiempo de reacción	Capacidad de iniciar rápidamente una respuesta a un estímulo.

Velocidad de movimiento de la extremidad	Capacidad de hacer un movimiento grueso de extremidades rápidas sin tener en cuenta el tiempo de reacción.
Velocidad de muñeca	Posibilidad de mover la muñeca y los dedos rápidamente.

Nota: Las abreviaturas se encuentran en inglés.

4.3.2 Fuentes primarias. La información recolectada del análisis literario provee una lista de habilidades con su respectiva definición tabla 10, pero esta es insuficiente para seleccionar la habilidad a entrenar en el módulo, lo cual forjó el diseño, evaluación y aplicación de una encuesta enfocada en la identificación del nivel de importancia que tiene cada una de las habilidades encontrada en la literatura a ser entrenadas en un residente que luego ejecutará una fijación transpedicular asumiendo las técnicas y tecnologías que se manipulan.

4.3.2.1 Tipo y técnica de muestreo. Se descarto la posibilidad de manejar técnicas de muestreo probabilistas ya que generalmente los tamaños de las muestras son grandes y los medios para acceder a esta es compleja.

Teniendo en cuenta que el tiempo de los ortopedistas es limitado, 3 de ellos accedieron a brindar información, por lo tanto, la técnica que se decidió emplear fue el muestreo no probabilístico por conveniencia, luego permite tramitar la aplicación de la mayor cantidad posible de encuestas para tener una muestra representativa.

4.3.2.2 Diseño del instrumento para la recolección de información. El enfoque cuantitativo fue el método usado como instrumento de investigación para recolectar y registrar información con el fin de conocer el nivel de importancia de entrenamiento de cada habilidad.

4.3.2.3 Validación del instrumento de recolección de información. Una vez diseñados los instrumentos se procedió a perpetrar la validación de estos con profesionales, la persona encargada de la evaluación fue el MSc. Luis Eduardo Bautista Rojas, investigador y docente de la universidad.

Se pudieron resaltar las siguientes observaciones:

- Examinar los significados y descartar la redundancia de habilidades con diferente nombre.
- Agrupar habilidades que se asemejan en grupos, lo cual facilita la lectura de la encuesta.
- Añadir la tabla 9 al instrumento, puesto que suministraría el significado de la habilidad al encuestado en caso de no conocerlo o recordarlo.

Estos aportes fueron estudiados y se reorganizó nuevamente el instrumento que finalmente fue utilizado (Apéndice A). Una vez validados los instrumentos, se procedió a la ejecución.

4.3.2.4 Análisis de resultados

Tabla 11.
Reporte de respuestas entregados por los encuestados.

<i>Grupo</i>	<i>Habilidad</i>	<i>Cantidad</i>
Acoplamiento	Acoplamiento proximal distal	3
	Acoplamiento respiratorio / movimiento	3
Control	Control prospectivo	3
	Control de fuerza fina	3
	Control de clasificación	3
	Control de flexibilidad de la gestión de la atención	2
Coordinación	Balance y control de la postura	2
	Coordinación bimanual	3
	Coordinación óculo-manual	3

	Coordinación interpersonal	3
	Coordinación de múltiples miembros	3
	Coordinación gruesa del cuerpo	3
Destreza	Destreza de los dedos	3
	Destreza manual	3
	Puntería	3
Equilibrio	Equilibrio bruto del cuerpo	2
Flexibilidad	Flexibilidad de extensión	3
	Flexibilidad dinámica	3
Fuerza	Fuerza estática	3
	Fuerza dinámica	3
	Fuerza explosiva	3
	Fuerza del tronco	3
Percepción	Percepción por tacto	3
	Percepción observacional	3
	Percepción espacial	3
	Percepción de profundidad/ triangulación	3
Planificación	Estrategias de afrontamiento y esquemas de respuesta alternativas	3
	Organización de la memoria, estructura y desarrollo del conocimiento de esquemas	3
Precisión	Exactitud	3
	Precisión de control	3
	Firmeza del brazo	3
Resistencia	Aguante	3
Respuesta	Orientación de respuesta	3
	Tiempo de reacción	3
	Anticipación	3
Sensación	Sensación táctil	3
Sincronización	Sincronización	3
Velocidad	Velocidad de movimiento de la extremidad	3
	Velocidad de muñeca	3

Todas las habilidades obtuvieron 3 respuesta excepto equilibrio bruto del cuerpo y control de flexibilidad de la gestión de la atención, donde solo se registraron 2 respuestas esto se debe, a

que uno alguno de los encuestados no dio respuesta al nivel de importancia de estas, como se refleja en la tabla 11.

Tabla 12.
Análisis de frecuencias.

Habilidad	Nivel de importancia
Coordinación	5
Destreza	5
Percepción	5
Planificación	5
Precisión	5
Sincronización	5

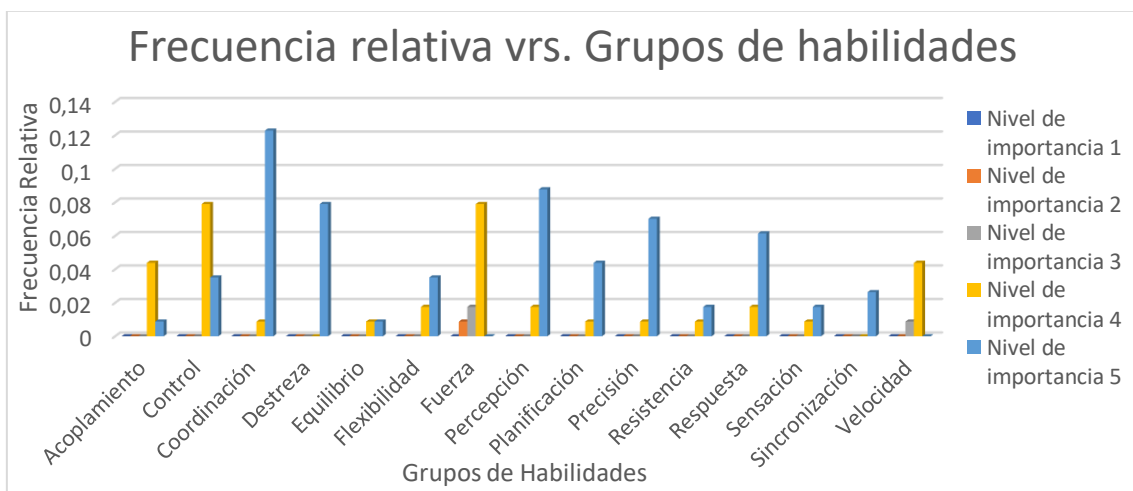


Figura 7 Frecuencia relativa vs. grupos de habilidades

Cada encuestado manifestó su opinión pertinente al nivel de importancia para cada habilidad y posteriormente se concibió un análisis de frecuencias que proporcionó el resultado expuesto en la tabla 12, donde se evidencia que 6 habilidades tenían el mismo nivel de importancia; por consiguientemente se planteó un análisis de frecuencias relativa y absoluta graficado en la figura 7, en el cual se observa que el grupo de habilidades que tiene un nivel de importancia mayor es

coordinación, esto se afirma al comparar las frecuencias relativas calculadas y confirmar que coordinación posee la mayor $f_i = 1,23$.

4.3.3 Fuentes primarias. Ranking de habilidades importantes en cirugías ortopédicas.

Rank	Surgeons [10] <i>n</i> = 101	Mean score (1–5)	Residents [9] <i>n</i> = 67	Mean score (1–5)	Surgeons- residents <i>n</i> = 195 [11]	Mean score (1–5)
1	Anatomical knowledge	3.86 ^a	Anatomical knowledge	4.4	Anatomical knowledge	4.63 ^b
2	Triangulation/depth perception	3.34 ^a	Spatial perception	4.3	Triangulation	4.43 ^b
3	Spatial perception	2.77 ^a	Triangulation/depth perception	4.2	Spatial perception	4.29 ^b
4	Manual dexterity	2.86 ^a	Manual dexterity	4.2	Tactile sensation	4.00 ^b
5	Tactile sensation	2.05 ^a	Tactile sensation	3.7	Manual dexterity	3.85 ^b

Figura 8 Ranking de habilidades técnicas - (Henrique & Karahan, 2017)

Henrique & Karahan, 2017 en su trabajo titulado “Motor skills training in Orthopedic sports medicine” recopiló tres estudios que documentan el nivel de importancia de algunas habilidades en ortopedia en cirujanos, residentes y cirujanos residentes, y los 3 concluyen el mismo ranking expuesto de la figura 8.

4.4 Elección de la habilidad técnica a entrenar.

La encuesta concluyó que la habilidad técnica con un nivel de importancia mayor según los ortopedistas, son las pertenecientes al grupo de coordinación y el ranking expuesto por Henrique & Karahan, 2017. confirma que en la posición 4 se encuentra la habilidad destreza manual (ver figura 7).

Aunque las dos son habilidades técnicas diferentes, son complementarias, es decir, para poseer destreza manual se necesita de habilidades que involucran combinaciones de movimientos

músculo- esquelético, eficiencia y rapidez, por lo cual, se dice que de la habilidad coordinación óculo-manual depende la destreza manual.

Por ello se tomó la decisión de entrenar la habilidad óculo-manual, pues sus principales propósitos son; potenciar la destreza manual, controlar la fuerza, ajustar la coordinación de movimientos e independizar los movimientos de las manos y garantiza la respuesta rápida y precisa con las manos a percepciones visuales.

5. Fase II: Diseñar ejercicios para al entrenamiento de la habilidad coordinación óculo-manual para conformar el módulo de aprendizaje.

El objetivo principal de esta fase es establecer la definición de la habilidad seleccionada conociendo sus principios y propósitos, los cuales fijarán el diseño de los ejercicios que conformarán el módulo de entrenamiento.

Las actividades elaboradas en esta fase se especifican en la tabla 13.

Tabla 13.

Descripción de la metodología de la fase II.

Actividades	Buscar el concepto de la habilidad seleccionada.	Diseñar los ejercicios a implementar en el módulo de entrenamiento	Crear indicadores de desempeño que posibiliten la evaluación y admisión de los ejercicios.
Resultado	-Concepto de coordinación. -Concepto de coordinación óculo-manual.	-Diseño de ejercicios	-Indicadores de desempeño

5.1 ¿Qué es coordinación?

Coordinación, según “Real Academia Española,” n.d. “proviene del latín *coordinatio*, es una acción y efecto de coordinar o coordinarse”, admite una gran variedad de conceptos o significados según el contexto en el que se manipule, este trabajo concierne al contexto anatómico, donde:

Álvarez del Villar “señala que la coordinación es la capacidad neuromuscular de ajustar con precisión lo querido y pensado de acuerdo con la imagen fijada por la inteligencia motriz a la necesidad del movimiento” (Contreras Jordán, 1998).

Jiménez Ortega & Jiménez De La Calle, 2010. definen que coordinación “es aquella capacidad del cuerpo para asociar el trabajo de diversos músculos, con la intención de realizar unas determinadas acciones.”

5.1.1 Tipos de coordinación. Muñoz Rivera, n.d. después de estudiar propuestas de autores como Contreras, Escobar, Le Voulch, “concreto una clasificación general sobre la coordinación, en función de dos aspectos importantes”:

A. En función de si interviene el cuerpo en su totalidad, en la acción motriz o una parte determinada, podemos observar dos grandes tendencias:

- *Coordinación Dinámica general:* es el buen funcionamiento existente entre el sistema nervioso central y la musculatura esquelética en movimiento. Se caracteriza porque hay una gran participación muscular.
- *Coordinación Óculo-Segmentaria:* es el lazo entre el campo visual y la motricidad fina de cualquier segmento del cuerpo. Puede ser *óculo-manual* y *óculo-pédica*.

B. En función de la relación muscular, bien sea interna o externa, la coordinación puede ser:

- *Coordinación Intermuscular (externa)*: referida a la participación adecuada de todos los músculos que se encuentran involucrados en el movimiento.
- *Coordinación Intramuscular (interna)*: es la capacidad del propio músculo para contraerse eficazmente.

5.2 ¿Qué es coordinación óculo-manual?

Coordinación óculo – manual denominada también coordinación viso-manual o mano ojo, hace parte de la coordinación óculo-segmentaria, y tiene como finalidad desarrollar la posibilidad de reaccionar mediante manifestación a percepciones visuales.

La coordinación óculo- manual es la capacidad que posee un individuo de utilizar simultáneamente las manos y la visión con el objetivo de realizar una tarea o actividad, en otras palabras, consiste en integrar simultáneamente la información que proporciona los ojos, para guiar el movimiento de las manos (Jiménez Ortega & Alonso Obispo, 2007).

5.2.1 Fases de coordinación óculo- ojo. A continuación se describirán las 4 fases de la habilidad óculo- manual o óculo-manual identidades por Chavez Diego & Valdivia Fierro, 2015.

- **Detención de objetos.** Se localiza el objeto y se analiza, para poder realizar correctamente la trayectoria hacia el objeto.
- **Trayectoria.** Se desarrollan los programas de ejecución motriz que van a ser patrones motores que colocan la mano y dedos en la posición adecuada para coger las cosas.

Objeto grande: menor flexión de interfalángicas y mayor dorsiflexión de la muñeca.

Objeto pequeño: mayor flexión de interfalángicas y menos dorsiflexión de muñeca.

- **Reflejo de acomodación del cristalino.** El cristalino tiene una capacidad de abombarse más o menos dependiendo a qué distancia esté el objeto que mira el ojo. Cuanto más cerca, más se engrosa el cristalino. Cuanto más lejos más se alarga el cristalino.
- **Fenómeno de convergencia ocular.** Los ojos hacen movimientos, pero se convergen hacia el objeto que miran. Cuanto más cerca esté el objeto, se nos aproximan los dos ojos a la línea media. Si el objeto está más lejos estos están divergentes.

5.3 Diseño de ejercicios.

El módulo de entrenamiento para la habilidad coordinación mano- ojo tiene como finalidad capacitar a estudiante y residentes de medicina en el desarrollo de su motricidad fina mediante actividades de identificación y ejecución, desarrollando tareas coordinadas basadas en percepción de información visual alcanzadas con las manos.

- El módulo de entrenamiento está compuesto por tres (3) niveles, y cada uno de ellos constará de cuatro (4) ejercicios diferentes que se establecen en el nivel básico e irán sufriendo modificaciones en sus condiciones para aumentar su dificultad y agruparlos en un nivel intermedio y avanzado.

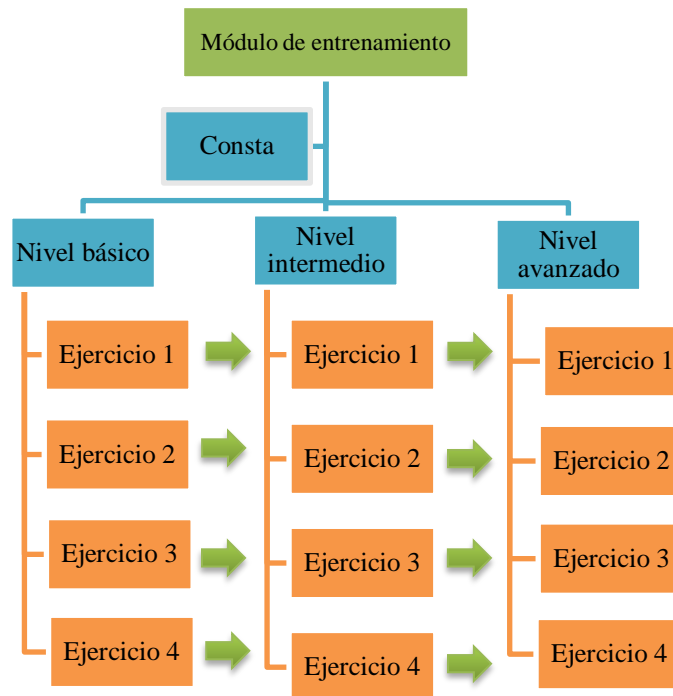


Figura 9 Composición del módulo de entrenamiento- las flechas diagonales, hacen alusión a las modificaciones que se llevan a cabo en cada ejercicio para aumentar su dificultad.

Tabla 14.
Detalles ejercicio 1.

Ejercicio 1				
Nombre	Ubicación de objetivos			
Objetivo	Identificar y ubicar los objetivos que tardan en aparecer y desaparecer un tiempo t mediante el sentido de la vista, con el fin de dirigir la atención y ayudar al cerebro a percibir dónde se sitúa el objetivo en el objeto.			
Descripción	Eliminar los objetivos identificados visualmente haciendo una colisión, donde utilice la exterminada superior para agarrar la herramienta y dirigirla al objetivo. Sobre un objeto geométrico tridimensional, aparecerá un objetivo geométrico, que deben ser identificado visualmente por el usuario para intentar eliminarlo realizando una colisión entre la herramienta y el objetivo, sin embargo, si el usuario (estudiante o residente) no es capaz de lograrlo el objetivo desaparecerá en un tiempo t y posteriormente aparecerá un nuevo objetivo.			
Instrucciones	Identificar el objeto Identificar el objetivo Identificar herramienta Realizar una colisión entre la herramienta y el objetivo			
Indicadores de desempeño				
Factor	Indicador	Unidad	Dimensión	Valor de aceptación

Número de errores	Mide el número de objetivos no eliminados	Porcentaje	$NE = \frac{\text{número de errores}}{\text{número de objetivos}} * 100\%$	$NE > 10\%$
Número de objetivos	Mide el número de objetivos eliminados	Veces	$NO = \text{número de objetivos eliminados}$	$NO = 10$

Tabla 15.
Detalles ejercicio 2

Ejercicio 2				
Nombre	Seguir objetivo			
Objetivo	Seguir el objetivo identificado visualmente con ayuda de la mano para desplazar la herramienta simultáneamente que se va desplazando el objetivo en el objeto.			
Descripción	Colisionar la herramienta con el objetivo ayudándose de la mano en todo el desplazamiento.			
Instrucciones	Sobre un objeto geométrico tridimensional, asomará un objetivo que el usuario debe identificar y acercando la herramienta con su extremidad superior efectuará una colisión continuando el movimiento que esté lleve en una determinada dirección.			
	Identificar el objeto Identificar el objetivo Identificar herramienta Realizar una colisión entre la herramienta y el objetivo Seguir movimiento con la colisión realizada.			
Indicadores de desempeño				
Factor	Indicador	Unidad	Dimensión	Valor de aceptación
Número de errores	Mide el número de objetivos no eliminados	Porcentaje	$NE = \frac{\text{número de errores}}{\text{número de objetivos}} * 100\%$	$NE > 10\%$
Número de desplazamientos	Mide el número de desplazamientos realizados	Veces	$NO = \text{número de desplazamientos realizados}$	$ND = 10$
Tiempo de duración	Mide el tiempo que se le destinara para que el usuario cumpla con el <i>ND</i>	Segundos, Minutos, Horas.	$t = \text{tiempo de duración destinado}$	$t = 1 \text{ minuto}$

Tabla 16.
Detalles ejercicio 3.

Ejercicio 3	
Nombre	Coincidir trazo
Objetivo específico	Identificar visualmente el objetivo inicial y final para ubicar la herramienta en el inicial y ajustar la inclinación conveniente para lograr ejecutar un trazo entre los dos objetivos.

Descripción	Sobre un objeto geométrico tridimensional, aparecerán 2 objetivos que el usuario debe identificar, uno como el inicial y el otro el objetivo final y ajustando la herramienta a una inclinación conveniente al objetivo inicial debe coincidir con el final y conseguir un trazo.			
Instrucciones	<p style="text-align: center;">Identificar el objeto Identificar los objetivos Identificar herramienta Acercar la herramienta al objetivo inicial. Ajustar la inclinación de la herramienta para que coincida el trazo con el objetivo final.</p>			
Indicadores de desempeño				
Factor	Indicador	Unidad	Dimensión	Valor de aceptación
Número de trazos	Mide el número de trazos realizados	veces	$NT = \text{número de trazos realizados}$	$NT = 10$
Tiempo de duración	Mide el tiempo que se le destinara para que el usuario cumpla con el <i>ND</i>	Segundos, Minutos, Horas.	$t = \text{tiempo de duración destinado}$	$t = 1 \text{ minuto}$

Tabla 17.
Detalles ejercicio 4

Ejercicio 4				
Nombre	Avanzar trazo			
Objetivo específico	Identificar visualmente el objetivo, para calcular su dimensión y ayudándose de las manos, sostener la herramienta e introducirla dentro del objetivo sin colisionar con las paredes.			
Descripción	Sobre un objetivo geométrico tridimensional, aparecerá un túnel que el usuario debe identificar para introducir la herramienta dentro de él y desplazarla hasta el final sin cometer ninguna colisión con las paredes.			
Instrucciones	<p style="text-align: center;">Identificar el objeto Identificar túnel Identificar herramienta Introducir la herramienta en el túnel. Desplazar la herramienta dentro del túnel sin colisionar con las paredes.</p>			
Indicadores de desempeño				
Factor	Indicador	Unidad	Dimensión	Valor de aceptación
Número de errores	Mide el número de objetivos no eliminados	porcentaje	$NE = \frac{\text{número de errores}}{\text{número de objetivos}} * 100\%$	$NE > 10\%$
Número de trayectos	Mide el número de trayectos realizados	veces	$NTY = \text{número de trazos realizados}$	$NTY = 10$
Tiempo de duración	Mide el tiempo que se le destinara	Segundos, Minutos, Horas.	$t = \text{tiempo de duración destinado}$	$t = 1 \text{ minuto}$

para que el
usuario
cumpla con el
ND

6. Fase III: Diseño de la herramienta

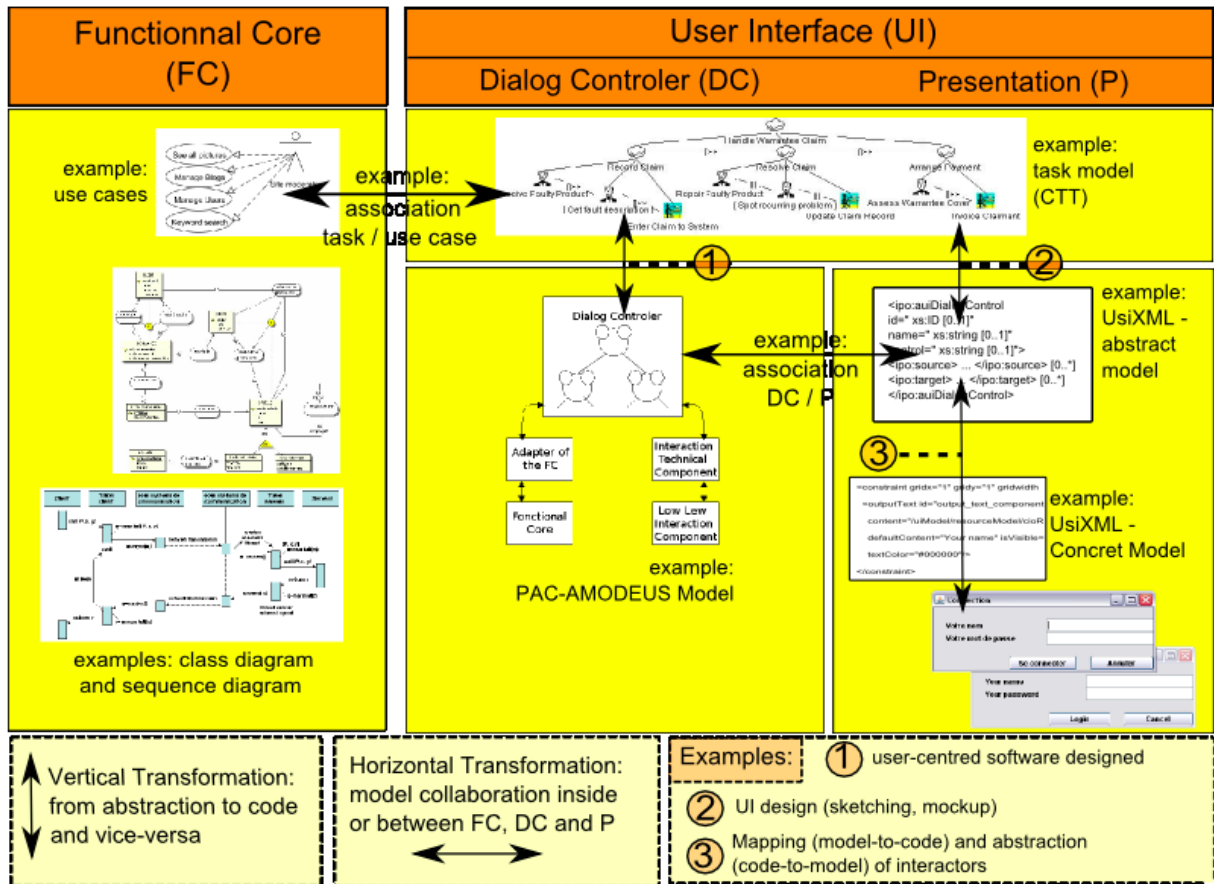


Figura 10 Etapas de diseño- tomada de Calvary, Dery-Pinna, Occello, Renevier-Gonin, & Riveill, 2014.

En este capítulo se da a conocer el diseño del módulo de entrenamiento que abarca en dos partes,

1. El núcleo funcional (parte oculta) y 2. La interfaz de usuario (parte visible).

El núcleo funcional corresponde a la parte funcional o parte oculta del módulo y se describió por medio de una arquitectura y la interfaz de usuario corresponde a la parte visible del módulo y

se diseñó siguiendo el enfoque basado en modelos desarrollado en el marco de referencia Cameleon, empezando con la identificación de modelos y tareas, luego interfaces de usuario abstractas (AUI), posteriormente interfaces de usuario concretas (CUI) y finalmente las interfaces de usuario finales (FUI), donde cada transformación se efectúa por medio de reglas de transformación.

6.1 Primera etapa: Núcleo funcional

Para la obtención la arquitectura del módulo de entrenamiento se estableció la metodología exhibida en la tabla 18.

Tabla 18.

Descripción de la metodología de la primera parte de la fase III

Actividades realizadas	Levantamiento de requerimientos	Diagramas de casos de uso	Arquitectura del módulo de entrenamiento
Resultados	-Documento de especificación de requerimientos	-Diagramas de casos de uso -Documento de especificación de casos de uso.	-Arquitectura del módulo de entrenamiento

6.1.1 Diagramas de casos de uso.

En la figura 11 se entrega la representación gráfica del módulo de entrenamiento, describiendo lo que hace desde el punto de vista del observador externo, concentrándose en lo que hacen los usuarios y no en dar respuesta en cómo lograr su comportamiento, la especificación de estos se encuentra en el Apéndice B.

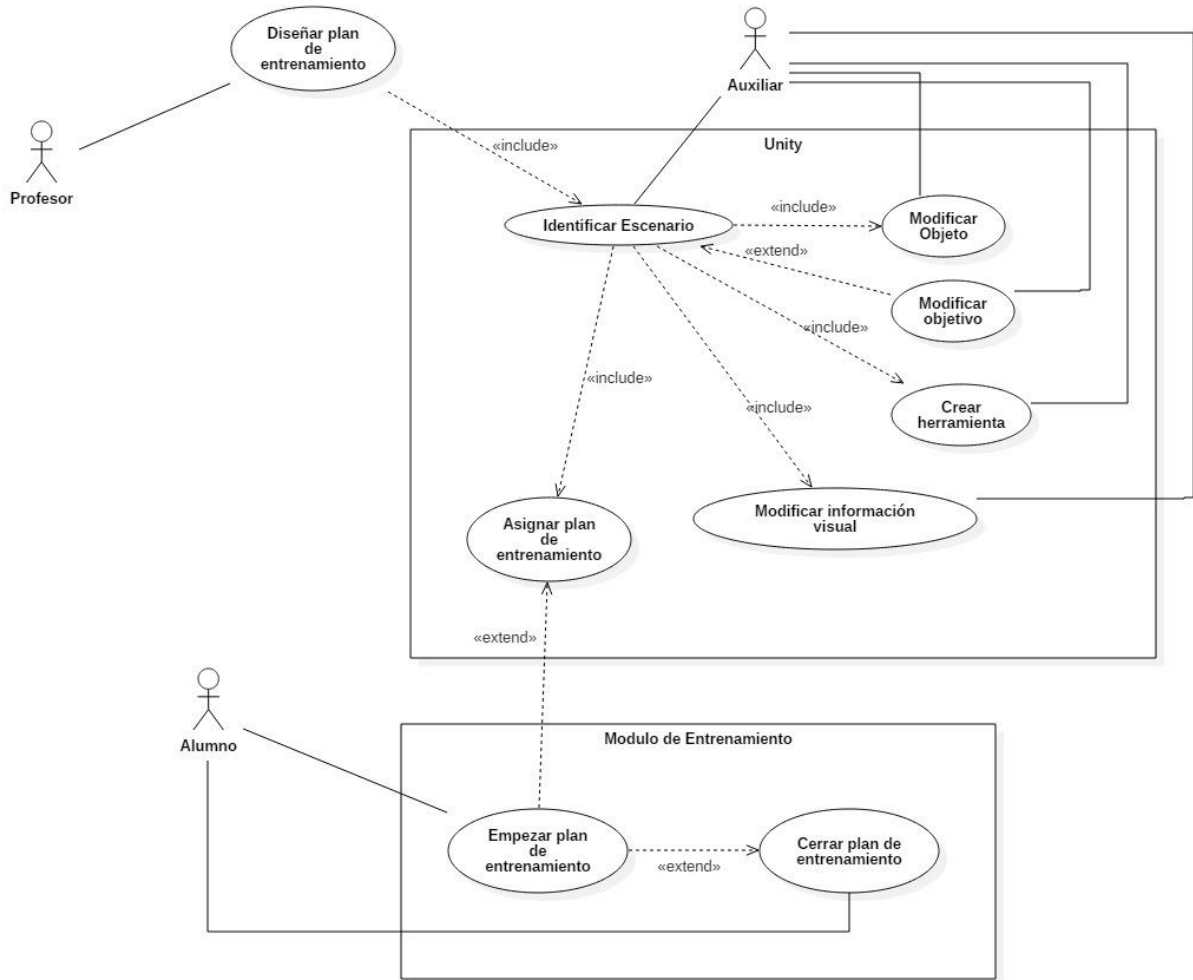


Figura 11 Diagrama de casos de uso

6.1.2 Arquitectura del software

Después de especificados los requerimientos (que se pueden encontrar en el documento de especificación de requisitos, apéndice C se crea una arquitectura figura 12 constituida por periféricos, exactamente 2 cámaras, que cumplen con el objetivo de reconocer los objetos en tiempo real y asociarlos con los reconstruidos.

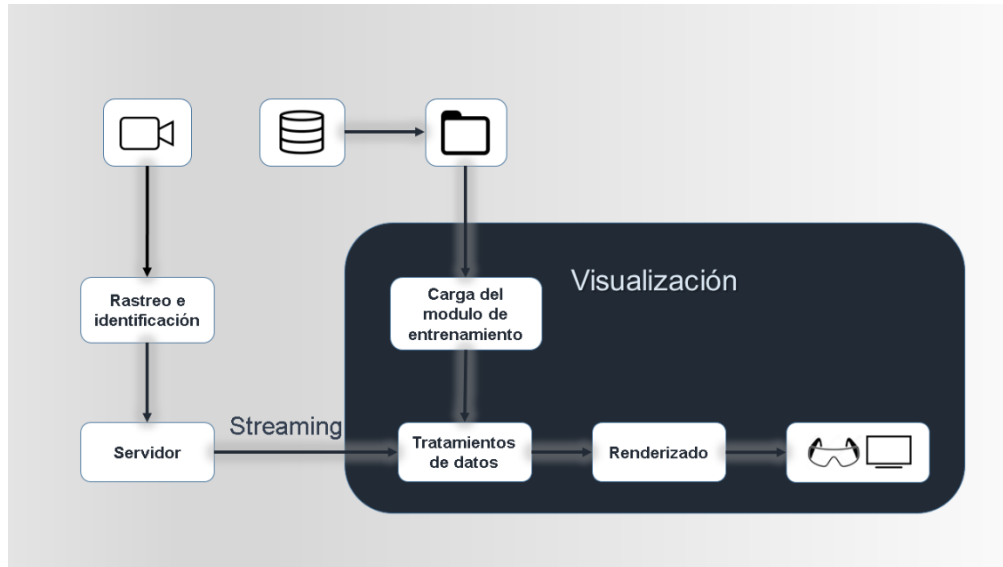


Figura 12 Arquitectura del módulo de entrenamiento

6.2 Interfaz de usuario

Para la obtención del diseño de la interfaz de usuario se siguieron las actividades descritas en la tabla 19.

Tabla 19.

Descripción de la metodología de la segunda parte de la fase III

Actividades realizadas	Modelos y mapeo de modelos	Modelos de interfaz de usuario abstractos	Modelos de interfaz de usuario concretos	Modelos de interfaz de usuario finales
------------------------	----------------------------	---	--	--

Resultados	<ul style="list-style-type: none"> -Modelos de tareas. -Modelos de dominio. 	<ul style="list-style-type: none"> -Reglas de transformación. -Modelos AIU. -Diseño de plan de pruebas. -Documento de ejecución y reporte de pruebas. 	<ul style="list-style-type: none"> -Modelo de intermodelos -Mapping -Selección de objetos de interacción concretos. -Reglas de transformación -Modelos CUI -Diseño de plan de pruebas. -Documento de ejecución y reporte de pruebas. 	-Modelos finales.
-------------------	---	---	---	-------------------

6.2.1 Modelos de tareas. Aplicando la notación CTT a la propuesta de ejercicios elaborada en la fase II, se obtienen 4 modelos de tareas que se añaden en el apéndice D; examinando estos modelos se determinó que cumplen con la estructura de la figura 13, la cual permite la separación de la escena o parte funcional que contiene objetos virtuales y reales moldeados por unas condiciones que proporcionan un nivel de dificultad; de la interfaz de usuario.

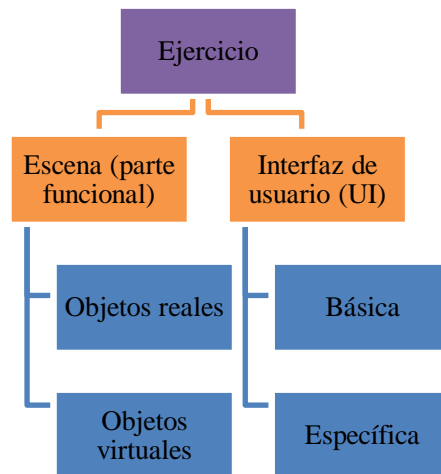


Figura 13 Composición de los ejercicios

- Modelo de tareas para la interfaz de usuario (UI)

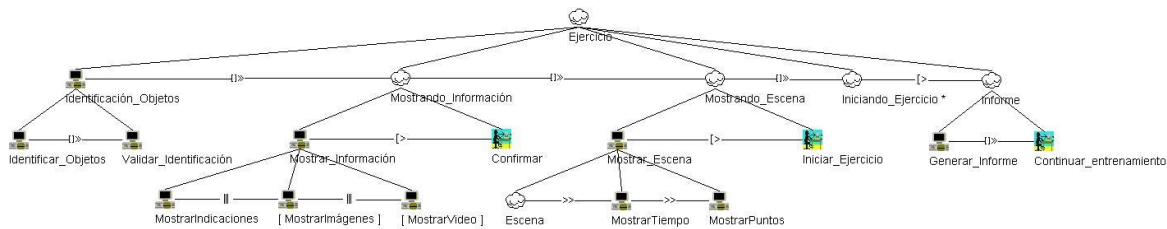


Figura 14 Modelo de tareas para la interfaz de usuario de los ejercicios.

En la figura 14 se modelaron las tareas a cumplir por el usuario, el sistema o ambos. Se inicia con la tarea abstracta “Ejercicio” que se compone de 5 tareas abstractas llamadas con los siguientes identificadores “*Identificación_Objetos*, *Mostrando_Información*, *Mostrando_Escena*, *Iniciando_Ejercicio*, *Informe*”, las cuales se descomponen jerárquicamente en nuevas tareas, que representan un paso a paso para lograr la realización de la tarea abstracta “*Ejercicio*”.

El segundo nivel empieza con la tarea de aplicación “*Identificación_Objetos*”, que se consigue efectuando la tarea “*Identificar_Objetos*” donde el sistema le proporciona información que el usuario debe seguir para que consecutivamente se validen los objetos a utilizar en la escena “*Validar_identificación*”.

La siguiente tarea abstracta es “*Mostrando_información*”, para cumplirla se debe poseer la identificación de la validación de objetos elaborada en la tarea “*Identificación_Objetos*” ([>>]); y se compone de la tarea de aplicación “*Mostrar_Información*” que cumple la función de mostrarle al usuario en cualquier orden (|||) texto, imágenes o videos las cuales explican las instrucciones que tendrá que seguir para cumplir los objetivos propuestos en el ejercicio mediante las tareas de aplicación identificadas como “*MostrarIndicaciones*, *MostrarImágenes*, *MostrarVideo*” siendo las dos últimas tareas opcionales.

Posteriormente sigue la tarea abstracta “*Mostrando_Escena*”, que consta de la tarea de aplicación “*Mostrar_Escena*”, la cual agrupa la tarea abstracta “*Escena*” y las tareas de

aplicación “*MostrarTiempo*” y “*MostrarContador*” que como sus nombres lo indican muestran el tiempo que se invierte en la realización del ejercicio y los puntos exitosos obtenidos por el usuario y pueden ser desactivadas ([>]) en cualquier momento por la tarea de interacción “*Iniciar_Ejercicio*”

Inmediatamente de identificar los objetos a emplear, obtener el conocimiento de los pasos a seguir en el ejercicio y mostrada la escena, continuamos con la tarea abstracta

“*Iniciando_ejercicio*” que es una tarea repetitiva (*) que agrupa las tareas comprometidas en la ejecución de un ejercicio específico de manera exitosa y varía en los 4 ejercicios.

Para finalizar encontramos la tarea abstracta “*Informe*” que se descompone en la tarea de aplicación “*Generar_Informe*” la cual presenta un resumen de todos los recursos manipulados en el ejercicio y finaliza con la tarea de interacción “*Continuar_Entrenamiento*” que genera la transición al siguiente ejercicio del módulo de entrenamiento.

Modelos de tareas para las escenas

En esta sección se describirán los modelos de tareas hechos para cada ejercicio, detallados en la tarea abstracta “*Iniciando_Ejercicio*”.

- **Modelo de tareas para la tarea abstracta “*iniciando_Ejercicio*” correspondiente al ejercicio 1.** La tarea abstracta “*Iniciando_Ejercicio*” está conformada por la tarea de aplicación “*Mostrar_Objetivos*” que se descompone de dos tareas que se realizan secuencialmente (>>) “*Mostrar_Objetivo1*” y “*Mostrar_Objetivo2*” que es una tarea opcional. Para este ejercicio (ejercicio 1) la segunda tarea no es necesaria, es decir, solo se efectúa la tarea “*Mostrar_Objetivo1*” que cumple la función de mostrar el objetivo para que el usuario lo perciba con el sentido de la vista y reaccione con la extremidad superior manipulando una herramienta, acercándola al objetivo para formar una colisión

que seguidamente el sistema debe verificar mediante la tarea “Verificación”, que posee la función de validar dos opciones ver figura 15; “selección_eliminación” o “Eliminar_por_Tiempo”. “selección_eliminación” comprueba si la colisión de la herramienta con el objetivo fue exitosa; si esto fue así, el objetivo se elimina y se le otorga un punto al usuario, de lo contrario el sistema verifica el tiempo límite establecido; si este se agotó el sistema elimina el objetivo en la tarea “Eliminar_por_Tiempo” y si no se agotó el usuario puede manipular nuevamente la herramienta para intentar eliminarlo.

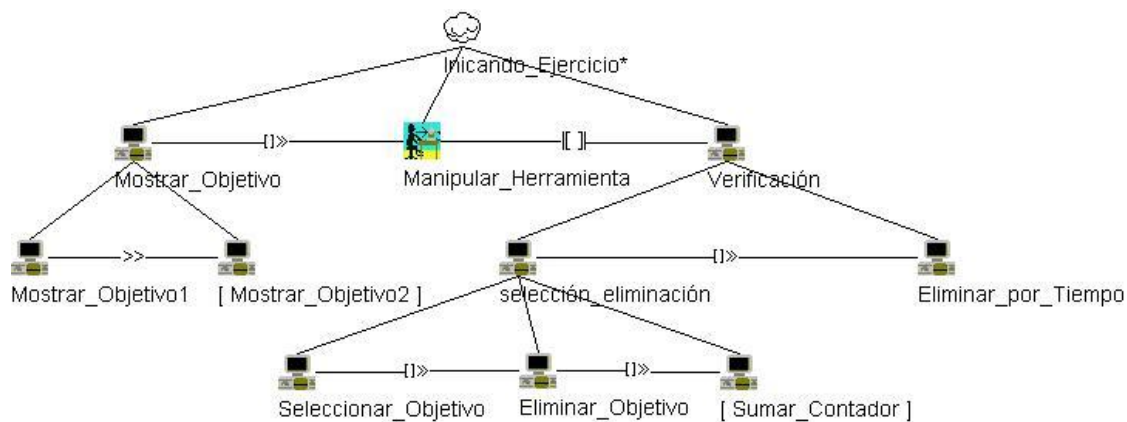


Figura 15 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 1

- Modelo de tareas para la tarea abstracta “iniciando_Ejercicio” correspondiente al ejercicio 2.** Para el segundo ejercicio (ejercicio 2) igualmente la tarea “Iniciando_Ejercicio” es repetitiva y se descompone de 3 tareas, dos de aplicación y una de interacción. La tarea de “Mostrar_Objetivos” revela el objetivo y el usuario debe acercar la herramienta al objetivo para colisionarlos durante todo el movimiento, mientras el sistema cumple la “Verificación” sincronizando ({}[]) la información que genera la tarea “Manipulando_Herramienta” (figura 16) y detectar si efectivamente se ejerció la colisión durante todo el movimiento; si esto fue exitoso se le proporciona un punto al usuario, de lo contrario debe volver a intentarlo.

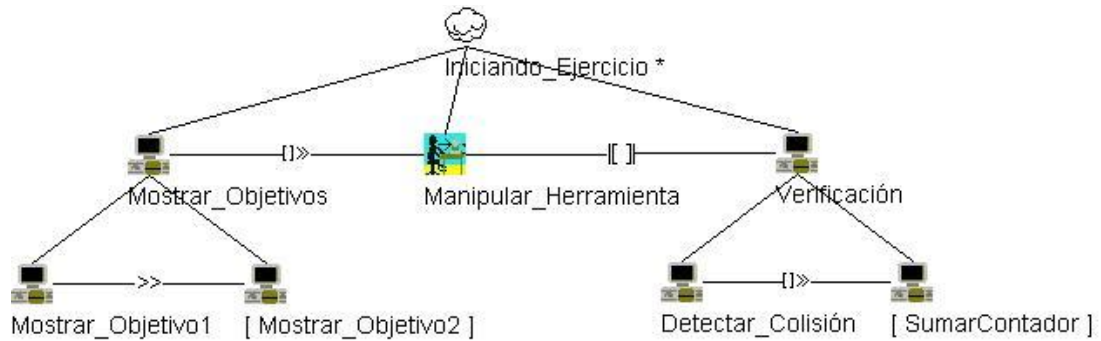


Figura 16 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 2

- Modelo de tareas para la tarea abstracta “iniciando_Ejercicio” correspondiente al ejercicio 3.** La tarea “Iniciando_Ejercicio” se compone de la tarea “Mostrar_Objetivos” que exhibe los dos objetivos, el inicial y el final; y la función del usuario es identificarlos para acercar la herramienta al objetivo inicial, modificando la posición y el ángulo, consiguiendo un trazo con el objetivo final que se verifica en la tarea “Verificar_trazado”, garantizando una sincroniza de información ([[]]) para posteriormente pintar el trazo mediante la tarea “Mostrar_Trazado” como se modelo en la figura 17, si este trazo coincidió se le concede un punto al usuario, de lo contrario el usuario debe volver a manipular la herramienta hasta lograrlo.

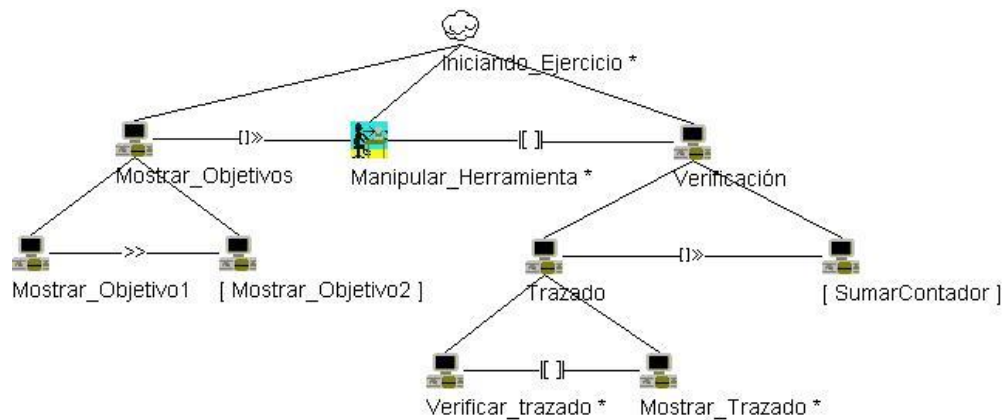


Figura 17 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 3

- Modelo de tareas para la tarea abstracta “iniciando_Ejercicio” correspondiente al ejercicio 4.** La tarea “Iniciando_Ejercicio” para el cuarto ejercicio (ejercicio 4) figura 18, está compuesta igual que los ejercicios anteriores por tres tareas, que de una u otra manera se sincronizan y generan información que necesita la próxima para poder realizarse. Después de mostrar el objetivo, el usuario debe ajustar la herramienta para iniciar el desplazamiento y mostrarlo; subsiguientemente se verifica que no se colisionaran las paredes del objetivo y se le confiere un punto al usuario, si no, debe volver a intentarlo.

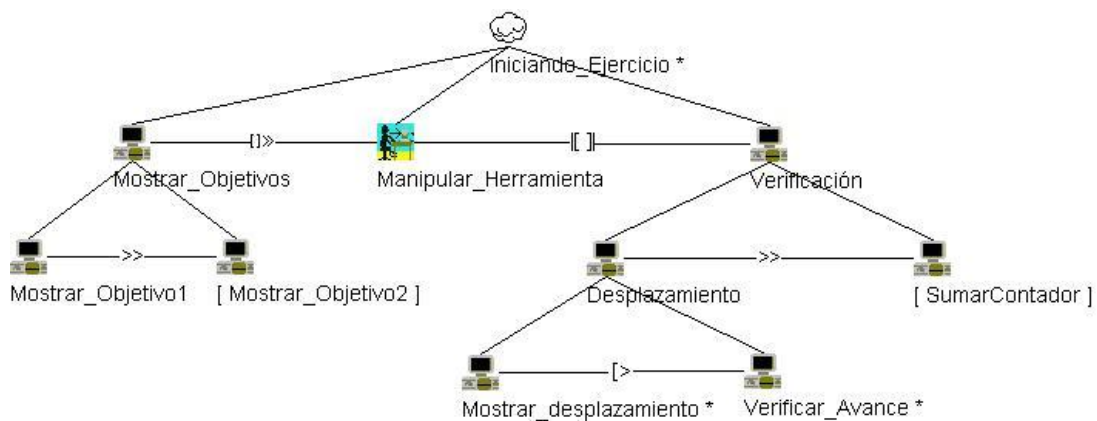


Figura 18 Tarea abstracta: Iniciando_Ejercicio para el ejercicio 4

6.2.2 Modelo de dominio de los ejercicios. Representan una visión de los objetos sobre los cuales actúan las tareas capturadas en los modelos de tareas, es modelado usando diagramas de clases, para definir atributos y relaciones que involucra el dominio del problema.

En la figura 19 se expresa el modelo de dominio creado para describir los ejercicios utilizando las nociones de herencia, agrupación, composición y dependencia.

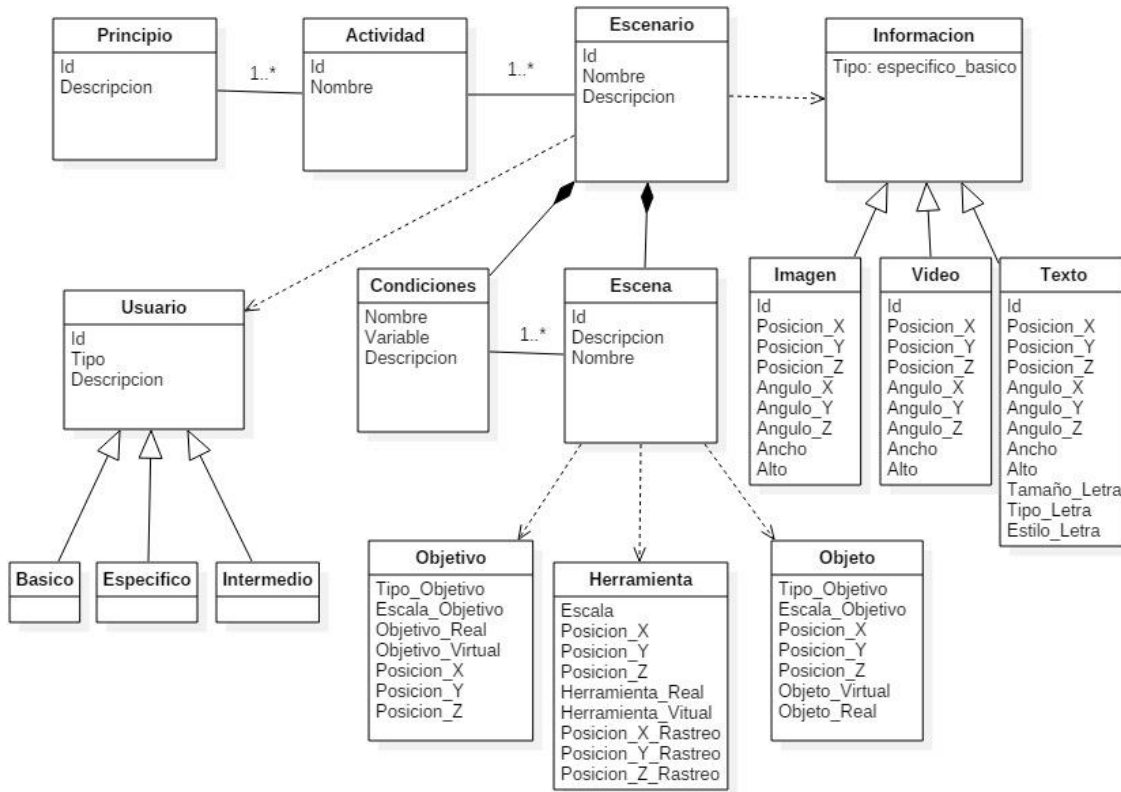


Figura 19 Modelo del dominio de los ejercicios

Modelos de presentación. Después de creados los modelos de tareas y dominio, se procede a efectuar transformaciones para crear los modelos de interfaz de usuario abstracta y concreta, empleando herramientas diseñadas para la generación de interfaces de usuario para distintas plataformas.

Inicialmente se planteó el uso de la herramienta de código libre Maria (“HIIS Laboratory | Research,” n.d.), que es el resultado de una investigación del laboratorio de interacción hombre-computador para describir los modelos de interfaces abstractos y concretos. Sin embargo se observaron limitaciones para la inclusión de elementos de interfaz requeridos, adicionalmente está limitada a plataformas de escritorio, móviles (“HIIS Laboratory | Home,” n.d.) incluyendo reglas de transformación determinadas que no se pueden modificar ni evaluar. Por lo tanto, se toma la

decisión de crear y validar reglas de transformación que permitan generar diseño de interfaces de usuario abstractas. Estas reglas y proceso se describen a continuación.

6.2.3 Interfaz de usuario abstracta -AUI. En esta sección se define un conjunto de detalles y reglas para la transformación de un modelo de tareas a un modelo de interfaz de usuario abstracta, que satisfacen con las nociones estudiadas en el capítulo 3.

6.2.3.1 Definición de categorías.

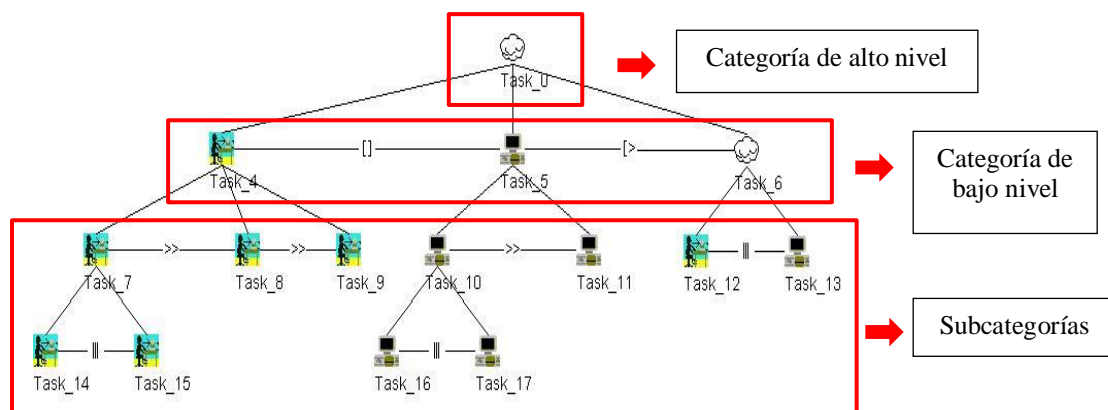


Figura 20 Categorías

En la figura 20, se presenta un ejemplo de modelo de tareas en notación CTT, y se identifican las 3 categorías que se definieron para expresar de las reglas de transformación.

Las categorías de niveles de abstracción son:

- Categoría de alto nivel:** Es la tarea principal o tarea raíz del modelo de tareas, se define como la tarea que se va analizar.
- Categoría de bajo nivel:** Son las tareas que se descomponen de la tarea raíz, generalmente están descritas en términos neutrales y satisfacen una zona de interés.

- c) **Subcategorías:** Son las tareas que van desde el 3 nivel hasta el último nivel que conforme el modelo de tareas, se caracterizan por representar un paso a paso de actividades que se deben cumplir para una tarea de categoría de bajo nivel.

6.3.2.2 Reglas de transformación de modelo de tarea a AUI.

Orden de ejecución de la transformación. La transformación del modelo de tareas a interfaz de usuario abstracta, se realizará de forma secuencial, garantizando el acatamiento de los operadores temporal de la notación CTT.

- Verticalmente: Se iniciará con la tarea de categoría de alto nivel hasta las subcategorías.
- Horizontalmente: Se transformarán de izquierda a derecha.

Regla de identificación. Esta regla se conforma de dos partes, la primera genera el nombre del identificador y la segunda establece la notación gráfica a implementar.

- Nombre del identificador: Cuando se efectúe la transformación, el objeto de interacción abstracto (AIO) conservará el identificador determinado en el modelo de tareas.
- Notación gráfica: Cada tarea se transformará en un rectángulo que en la parte inferior derecha sujetará el icono establecido a cada objeto abstracto o faceta y el nombre del identificador, ver tabla 20 y 21.

Tabla 20.
Establecimiento de iconos para los AIO.







Objeto abstracto	Icono
Componente de interacción – AIC	
Contenedor -AC	

Tabla 21.
Establecimiento de iconos para las facetas de los AIC

Faceta	Icono
Entrada	
Salida	
Navegación	
Control	

Regla de división. Esta transformación admitirá la segmentación de la presentación total (categoría de alto nivel), en un conjunto de interfaces individuales que conformarán las tareas de categoría de bajo nivel, que a su vez actuarán en espacios de tiempo limitado a nivel concreto y real. Esta regla de transformación proporcionará redundancia interna dentro de las interfaces individuales, pues la tarea de categoría de alto nivel se transfigurará cuantas veces existan la tarea de categoría de bajo nivel.

Regla de transformación. Especifica el AIO a utilizar dependiendo el tipo de tarea identificado en el modelo de tareas a transformar. Esa transformación se efectuará de la siguiente manera:

- a) Categoría de alto nivel:
 - La tarea de alto nivel se convertirá en un AC.
- b) Categoría de bajo nivel:
 - Si una tarea de bajo nivel se descompone en subtareas, se transformará en un AIC.
 - De lo contrario se convertirá en una faceta AIC y para ello se debe verificar la transformación estimulada en las subcategorías.

c) Las subcategorías:

- Si la tarea de subcategorías se descompone en subtareas, está se transformará en un AIC.
- Del contrario se transformarán en una faceta AIC. Para ellos se debe tener presente el tipo de tarea en notación CTT de la siguiente forma:
 - Las tareas de interacción, se convertirá en una faceta AIC de control.
 - Las tareas de sistema, se convertirán en una faceta AIC de salida.
 - Las tareas de usuario, serán eliminadas en la transformación ya que éstas no generan valor agregado a la interfaz abstracta.
 - Las tareas abstractas, no constarán con ninguna faceta de componentes de interacción abstracto, se transformará en un AC.

6.2.3.3 Aplicación de las reglas de transformación. Para aplicar las reglas de transformación se limita el enfoque sistémico ilustrado en la figura 21, estableciendo las entradas que se deben tener para poder aplicar las reglas de transformación y generar las salidas.



Figura 21 Enfoque sistemático de transformación de modelos de tareas a AUI

6.2.3.4 Modelos de interfaces de usuario abstractas para el módulo de entrenamiento.

Antes de digitalizar los modelos de interfaz de usuario abstractos, se diseñó y ejecutó un plan de pruebas para confirmar que, al aplicar las reglas de transformación al modelo de tareas, se generarán las mismas salidas, de lo contrario se tomarán acciones correctivas.

6.2.3.4.1 Plan de pruebas. Se diseñó un plan de pruebas estáticas, que permitió la generación de salidas a nivel de implementación sin necesidad de ejecutar un software (apéndice E).

En este plan se estableció un proceso de pruebas, en el cual los evaluadores tomaron una aproximación de “caja negra”, otorgándoles unas entradas para que ellos produjeran salidas. las salidas fueron verificadas, encontrándose hallazgos reportados en la tabla 22.

6.2.3.4.2 Ejecución y reporte de pruebas. Se estimó una muestra no probabilística homogénea a nivel de conocimiento, es decir, se les solicitó a 5 ingenieros de sistemas e informática con experiencia de 2 años, que tomarán el papel de evaluador y ejecutarán la prueba. Los resultados mostraron la obtención satisfactoria de salidas, a pesar de que se identificó la necesidad de realizar una introducción a las nociones de MBUID (apéndice F). Las salidas fueron verificadas y se reportan en la tabla 22, donde se identifican los números de AIO y facetas esperadas (fila resaltada con fondo verde) y los entregados por los evaluadores (final con fondo blanco), evidenciando que dos de ellos descartaron la transformación de a lo más una tarea (resaltados en naranja).

Tabla 22.

Reporte de pruebas interfaz de usuario abstracta.

E VAL	OBJETOS DE INTERACCIÓN ABSTRACTOS - AIO			
	Subinterfaz 1	Subinterfaz 2	Subinterfaz 3	Subinterfaz 4

	CONTENEDOR	COMPONENTE	SALIDA	ENTRADA	NAVEGACIÓN	CONTROL	CONTENEDOR	COMPONENTE	SALIDA	ENTRADA	NAVEGACIÓN	CONTROL	CONTENEDOR	COMPONENTE	SALIDA	ENTRADA	NAVEGACIÓN	CONTROL	CONTENEDOR	COMPONENTE	SALIDA	ENTRADA	NAVEGACIÓN	CONTROL
	1	1	2	0	0	0	1	2	3	0	0	1	2	2	2	0	0	1	1	1	1	0	0	1
1	1	1	2	0	0	0	1	2	3	0	0	1	2	2	2	0	0	1	1	1	1	0	0	1
2	1	1	2	0	0	0	1	1	3	0	0	1	2	2	2	0	0	1	1	1	1	0	0	1
3	1	1	2	0	0	0	1	2	3	0	0	1	2	2	2	0	0	1	1	1	1	0	0	1
4	1	1	2	0	0	0	1	2	3	0	0	1	1	2	2	0	0	1	1	1	1	0	0	1
5	1	1	2	0	0	0	1	2	3	0	0	1	2	2	2	0	0	1	1	1	1	0	0	1

Tabulados los datos se procedió a graficarlos figura 22, apoyando una tasa de reconocimiento del 95,8% para los AIO.

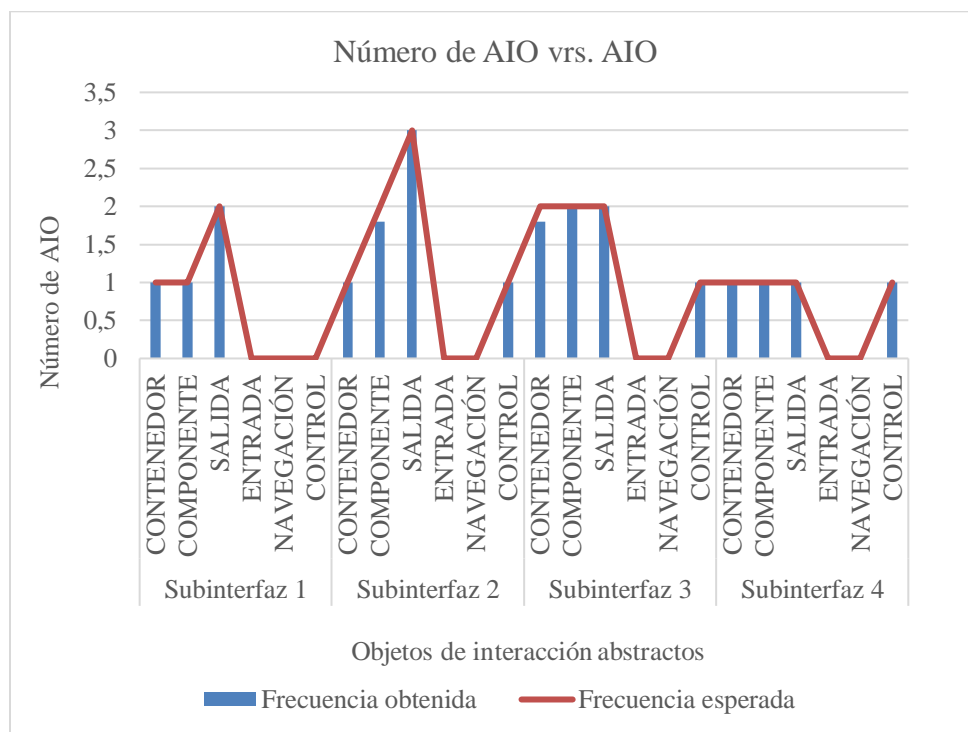


Figura 22 Resultados prueba estática 1

Otros errores percibidos, fue el orden de ejecución de la transformación y omisión de la colocación del icono definido para cada componente AIO y las facetas, además en el apéndice F se documentaron observaciones dadas por los evaluadores y se establecieron las acciones correctivas a tomar.

6.2.3.4.3 Modelos de interfaz de usuario abstractos. Los modelos de interfaces obtenidos como salidas se digitalizaron manipulando la herramienta Balsamiq Mockup (“Balsamiq. Rápido, efectivo y divertido software de wireframing. | Balsamiq,” n.d.) que facilita la creación de esquemas.



Figura 23 Subinterfaz de usuario abstracta 4

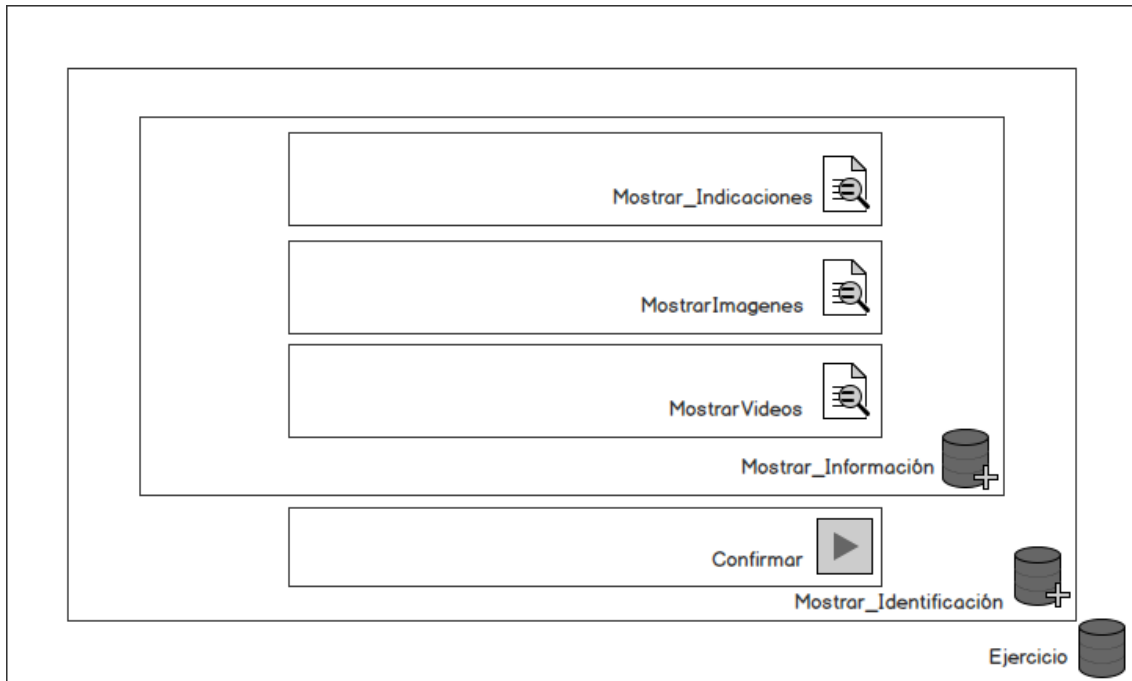


Figura 24 Subinterfaz de usuario abstracta 4

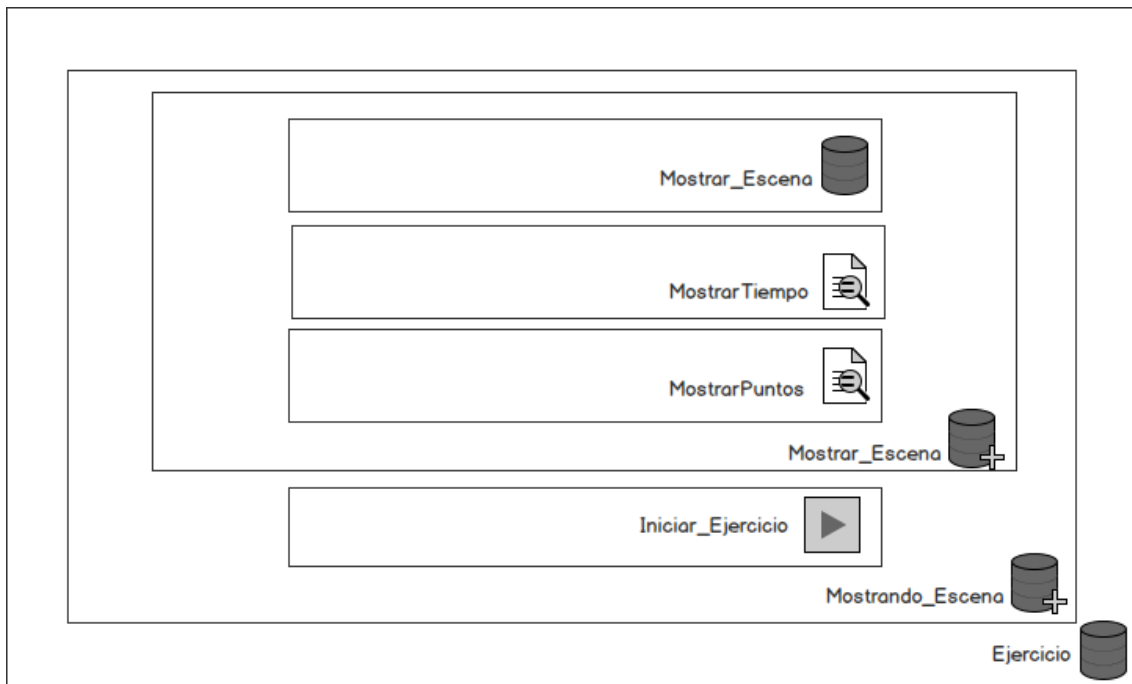


Figura 25 Subinterfaz de usuario abstracta 3

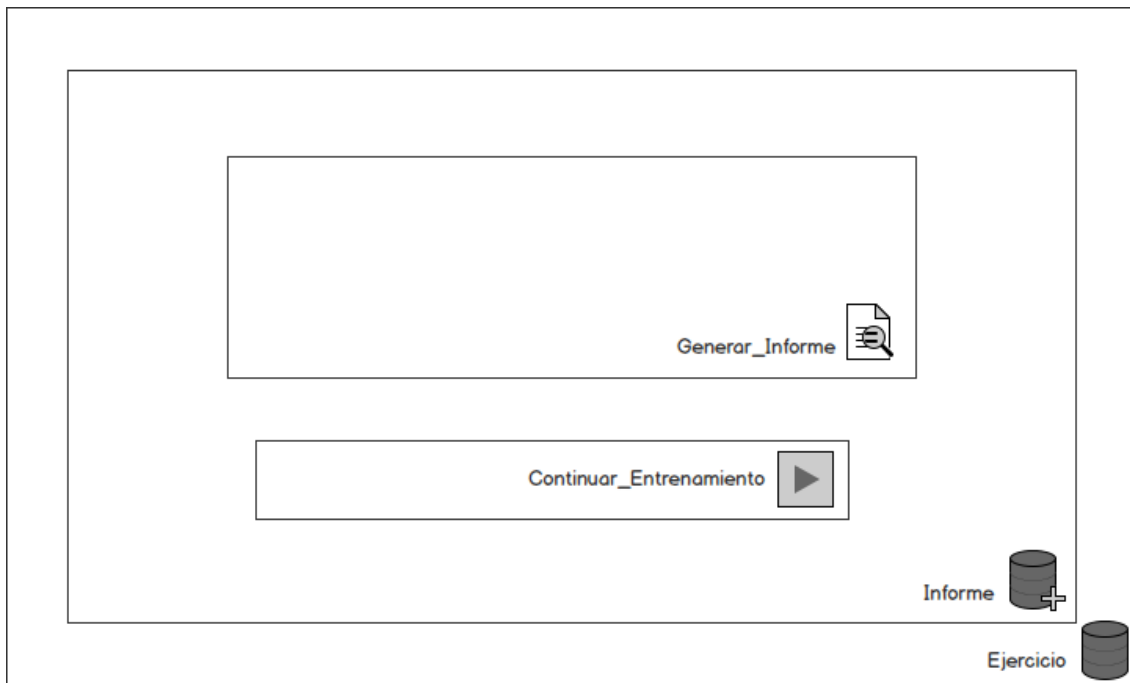


Figura 26 Subinterfaz de usuario abstracta 4

6.2.4 Interfaz de usuario concreta -CUI. A continuación, se muestra los modelos y reglas que se tuvieron que crear para lograr los modelos de interfaces de usuario concretas que se caracterizan por describir los elementos visuales de la interfaz incluyendo fuentes, tamaños, posiciones e interacciones.

6.2.4.1 Modelo de relaciones intermodelos (*Mapping*). Este modelo define las relaciones que aparecen entre el metamodelo de notación CTT y el metamodelo del diseño de los ejercicios (apéndice D).

En la figura 27 se ilustra que la única relación se da entre la noción de tarea de aplicación del metamodelo de notación CTT y la noción de información del metamodelo de los ejercicios.

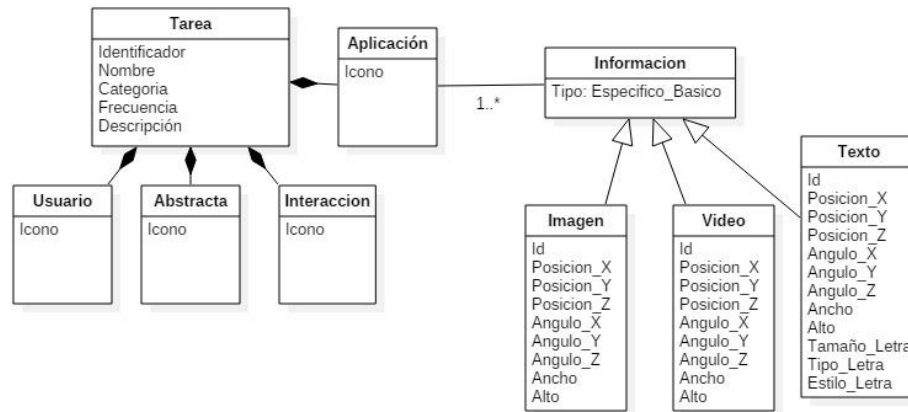


Figura 27 Mapping

6.2.4.2 Mapeo entre el modelo de tareas y las instancias del metamodelo de ejercicios. Se instanciaron las clases que heredan de la superclase información de forma general, y se hizo el mapeo que se muestra en la figura 28, donde se relacionan todas las tareas de sistema con las instancias correspondientes.

6.2.4.3 Selección de CIO. En la tabla 5 se registraron los CIO encontrados en la literatura para implementaciones bidimensionales, de tal manera que, se propone llevar a efecto una extensión de algunos de ellos para emplearlos en interfaces de usuario para aplicaciones que involucren realidad aumentada.

Se seleccionaron 4 widgets, label, panel, button y box, sometidos a una parametrización detallada de la siguiente manera:

- **Panel 3D:** Widget contenedor que puede contener otros widgets
- **Label 3D:** Este widget tiene por objetivo mostrarle texto estático al usuario que no puede ser modificado.
- **Complemento visual 3D:** Admite la inserción de imágenes y videos.

- **Botón 3D:** Widget que activa un evento cuando el usuario lo notifique con una acción

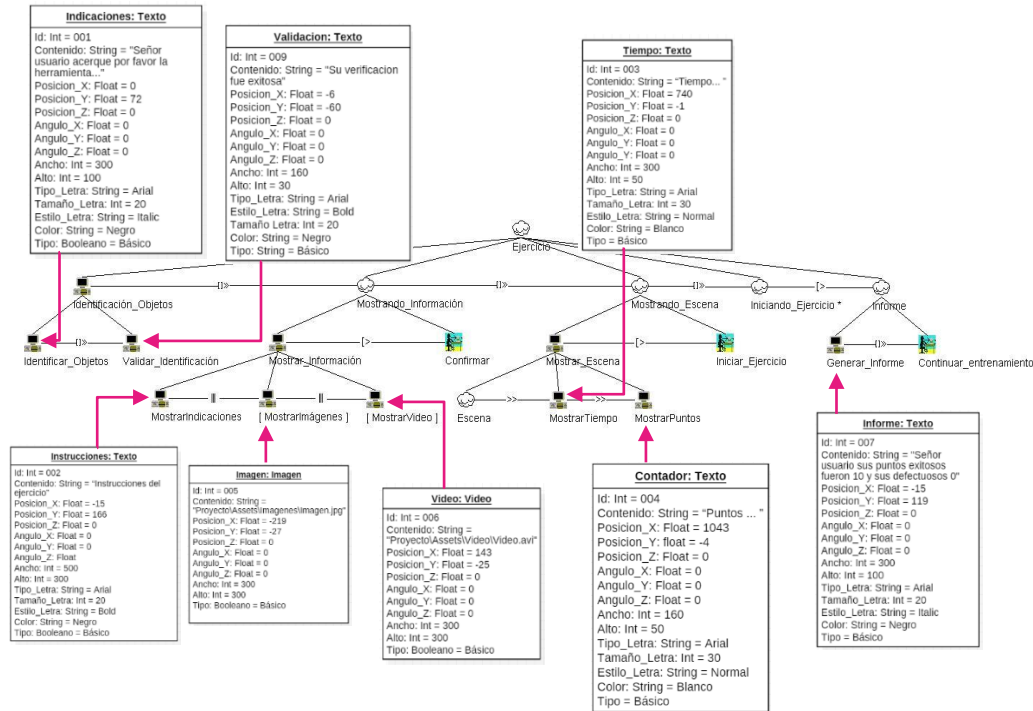


Figura 28 Mapeo entre los modelos de tareas y las instancias del modelo de dominio

6.4.4.4 Parametrización de los widgets 3D

Tabla 23.
Parámetros widgets label 3D.

Label 3D	
Parámetros	Descripción
Texto-predeterminado	Contenido del label.
Posición-x	Ubicación del label en el eje x.
Posición -y	Ubicación del label en el eje y.
Posición -z	Ubicación del label en el eje z.
Rotación-x	Rotación del label en el eje x.
Rotación-y	Rotación del label en el eje y.
Rotación-z	Rotación del label en el eje z.

Alto	Determina la altura del label
Ancho	Determina el ancho del label
Familia tipográfica	Estilo de fuente
Color de fuente	Determina el color de la fuente.
Tamaño de fuente	Determina el tamaño de la fuente.
Color de fondo	Establece el color de fondo del panel.
Name	Fija el nombre del panel

Tabla 24.

Parámetros widgets componente visual 3D.

Complemente visual 3D	
Parámetros	Descripción
Recurso	Contiene la URL de la imagen y/o video.
Posición-x	Ubicación del componente visual en el eje x.
Posición -y	Ubicación del componente visual en el eje y.
Posición -z	Ubicación del componente visual en el eje z.
Rotación-x	Rotación del componente visual en el eje x.
Rotación-y	Rotación del componente visual en el eje y.
Rotación-z	Rotación del componente visual en el eje z.
Escala	Escala del componente visual.
Name	Fija el nombre del componente visual.

Tabla 25.

Parámetros widgets botón 3D

Botón 3D	
Parámetros	Descripción
Etiqueta	Descripción el texto del botón
Posición-x	Ubicación del botón en el eje x.
Posición -y	Ubicación del botón en el eje y.

Posición -z	Ubicación del botón en el eje z.
Rotación-x	Rotación del botón en el eje x.
Rotación-y	Rotación del botón en el eje y.
Rotación-z	Rotación del botón en el eje z.
Alto	Determina la altura del botón
Ancho	Determina el ancho del botón
Name	Fija el nombre del botón

Tabla 26.
Parámetros widgets panel 3D

Panel 3D	
Parámetros	Descripción
Posición-x	Ubicación del label en el eje x.
Posición -y	Ubicación del label en el eje y.
Posición -z	Ubicación del label en el eje z.
Rotación-x	Rotación del label en el eje x.
Rotación-y	Rotación del label en el eje y.
Rotación-z	Rotación del label en el eje z.
Alto	Determina la altura del label
Ancho	Determina el ancho del label
Name	Fija el nombre del panel

6.2.4.5 Reglas de transformación de AUI a CUI.




Regla de orientación:


La orientación, es dependiente del modo de visualización final, puesto que cambia las propiedades en modo vertical u horizontal, por lo cual se define que la interfaz use una orientación horizontal.

Regla de notación gráfica:

Cada transformación de objeto de interacción abstracto a objeto de interacción concreto se hará manipulando un rectángulo, que sujetará el nombre de los widgets utilizado y el identificador del objeto abstracto que trae la interfaz abstracta.

Regla de transformación:

- Un AC personificado en la interfaz abstracta () se transformará en un panel 3D.
- Los AIC () no constarán de una transformación, es decir, quedarán eliminados en la interfaz de usuario concreta.
- La faceta AIC de control () se transformará en un botón 3D.

La faceta AIC de salida representada en la interfaz abstracta como () se transformarán verificando el mapeo realizado entre el modelo de tareas y las instancias del metamodelo de escenario de la siguiente manera:

- Si el mapeo se realiza con una instancia tipo texto, la faceta de salida AIC se va a transfigurar en un label 3D.
- Si el mapeo se relaciona con una instancia tipo video o imagen, la faceta de salida AIC se va a convertir en un componente visual 3D.

Regla de tamaño:

- **Panel 3D:** Su tamaño será proporcional a la suma de la altura de los CIO que contiene en su interior, tamaño máximo 18 cm x 10 cm
- **Label 3D:** Su tamaño será relativo a la cantidad de texto incluido y a su tamaño de fuente. Por consiguiente, se sugiere un tamaño de letra 10 y una cantidad máxima de 350 caracteres.

- **Botón 3D:** Su tamaño depende de la longitud y tamaño de la etiqueta. La longitud máxima será 25 caracteres en tamaño de letra 12.
- **Componente visual 3D:** Su tamaño debe proporcionar una visión clara, que permita al usuario final poder interpretarla. Su tamaño máximo será 5 cm x 5 cm.

Regla de posición:

El posicionamiento de los CIO define gran parte del aspecto visual final de la aplicación, por lo tanto, se determinan las siguientes posiciones:

Panel 3D

- El panel inicial no constará de una posición determinada puesto que hace las veces de contenedor o ventana principal.
- Si dentro del panel principal se registra otro panel este se posicionada en el centro del panel principal.

Botón 3D

- El botón 3D se posicionarán en el centro la parte inferior del panel.

Componentes visuales 3D

- Si existen dos componentes visuales 3D, estos se posicionarán uno en la derecha y el otro en la izquierda de parte inferior de la pantalla, de lo contrario se posicionará en el centro de la parte inferior.

Label 3D:

- Si existen dos label 3D seguidos, el primero se posicionará en el centro de la parte superior y el segundo en el centro de la parte inferior del panel.
- Si dentro de un panel, existen dos label 3D después de un panel, estos dos se posicionarán en la parte superior derecha de forma horizontal, respetando el orden.

6.2.4.6 Transformación de modelos de interfaz de usuario abstractas a modelos de interfaz de usuario concretas. El enfoque sistémico que describe el modo como interactúan los modelos necesarios para aplicar las reglas de transformación se ilustra en la figura 29.



Figura 29 Enfoque sistémico de transformación AUI a CUI

6.2.4.7 Modelos de interfaces de usuario concretas para el módulo de entrenamiento. El proceso para digitalizar las interfaces de usuario concretas se realizó de manera similar que, en los modelos de interfaz de usuario abstractos, donde se diseñó y ejecutó un plan de pruebas para confirmar que las reglas generaran las mismas interfaces en todos los casos.

6.2.4.7.1 Plan de pruebas. El plan de pruebas estáticas (apéndice C), descrito anteriormente contiene dos instrumentos para recolectar las salidas necesarias, en esta parte se usó el instrumento 2, el cual concede salidas de interfaces de usuario concretas.

6.2.4.7.2 Ejecución y reporte de pruebas. Para facilitar la adopción de nociones se les pidió a las mismas personas que generaron las salidas de interfaces de usuario abstractas para que continuaran con el proceso de transformación y llegarán a interfaces de usuario concretas; no se evidenciaron dificultades, el proceso de transformación fue más rápido y se registraron las salidas en tablas separando cada subinterfaz y contando el número de CIO en las posiciones identificadas como horizontales y verticales.

Tabla 27.

Reporte de errores subinterfaz de usuario concreta 1

EVALUADOR	Objetos de interacción concretos						
	Subinterfaz 1						
	PANEL 3D	LABEL 3D		COMPONENTE VISUAL 3D		BOTÓN 3D	
		H	V	H	V	H	V
	1	0	2	0	0	0	0
1	1	0	2	0	0	0	0
2	1	0	2	0	0	0	0
3	1	0	2	0	0	0	0
4	1	0	2	0	0	0	0
5	1	0	2	0	0	0	0

Nota: La abreviatura H corresponde a horizontal y la V corresponde a vertical

Tabla 28.

Reporte de errores subinterfaz de usuario concreta 2

EVALUADOR	Objetos de interacción concretos						
	Subinterfaz 2						
	PANEL 3D	LABEL 3D		COMPONENTE VISUAL 3D		BOTÓN 3D	
		H	V	H	V	H	V
	1	0	1	2	0	0	1
1	1	0	0	3	0	0	1
2	1	0	1	2	0	0	1
3	1	0	1	2	0	0	1
4	1	0	1	2	0	0	1
5	1	0	1	2	0	0	1

Tabla 29.

Reporte de errores subinterfaz de usuario concreta 3

EVALUADOR	Objetos de interacción concretos						
	Subinterfaz 3						
	PANEL 3D	LABEL 3D		COMPONENTE VISUAL 3D		BOTÓN 3D	
		H	V	H	V	H	V
	2	2	0	0	0	0	1
1	2	2	0	0	0	0	1
2	2	0	2	0	0	0	1
3	2	2	0	0	0	0	1
4	2	2	0	0	0	0	1
5	2	2	0	0	0	0	1

Tabla 30.

Reporte de errores subinterfaz de usuario concreta 4

EVALUADOR	Objetos de interacción concretos						
	Subinterfaz 4						
	PANEL 3D	LABEL 3D		COMPONENTE VISUAL 3D		BOTÓN 3D	
		H	V	H	V	H	V
	1	1	0	0	0	0	1
1	1	1	0	0	0	0	1
2	1	1	0	0	0	0	1
3	1	1	0	0	0	0	1
4	1	1	0	0	0	0	1
5	1	1	0	0	0	0	1

Los errores encontrados corresponden al posicionamiento de los widgets y se obtuvo una tasa de reconocimiento del 96,4% para los CIO, como se evidencia en la figura 30.

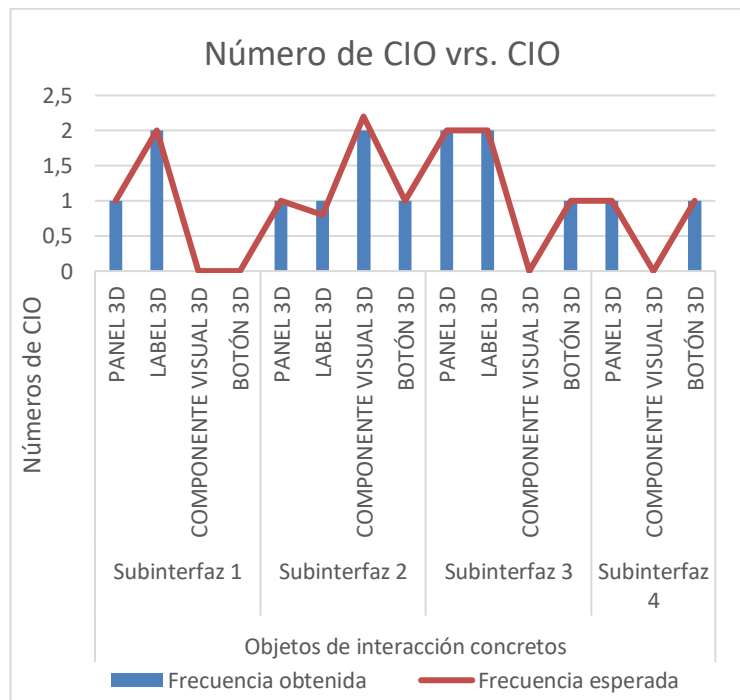


Figura 30 Resultado de prueba estática 2

6.2.4.7.3 Modelos de interfaz de usuario concreta. Los modelos de interfaces concretas obtenidos como salidas se expresan en lenguaje XML, es importante destacar que los parámetros son

modificables, correspondiendo a un bosquejo de lo que es la UI y las visualizaciones gráficas se anexan en el apéndice G

```

<cuimodel>
<panel3D name=" Subinterfaz2- Ejercicio" posición-x="0" posición-y="0"
posición-z="0" Rotación-x=" 0" Rotación-y= "0" Rotación-z= "0" ancho="1280"
alto="720" >
<Label3D name=" Identificar_Objetos" posición-x="0" posición-y="0" posición-
z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0" ancho="800"
alto="10" familiatipográfica="Arial" colorfuente="255,255,255"
tamañoFuente="14" colorfondo="0,0,255" />
<Boton3D name= "Confirmar" posición-x="0" posición-y="0" posición-z="0"
Rotación-x= "0" Rotación-y= "0" Rotación-z= "0" ancho="9" alto="3" />
</panel 3D>
</cuimodel>

```

```

<cuimodel>
<panel3D name=" Subinterfaz2- Ejercicio" posición-x="0" posición-y="0"
posición-z="0" Rotación-x=" 0" Rotación-y= "0" Rotación-z= "0"
ancho="800" alto="10" familiatipográfica=" Arial"
colorfuente="255,255,255" tamañoFuente="14" colorfondo="0,0,255" />
<Label3D name=" Mostrar_Indicaciones" posición-x="0" posición-y="1"
posición-z="0" Rotación-x=" 0" Rotación-y= "0" Rotación-z= "0"
ancho="800" alto="10" familiatipográfica=" Arial"
colorfuente="255,255,255" tamañoFuente="14" colorfondo="0,0,255" />
< Componentevisual3D name=" MostrarImágenes" escala="" posición-
x="-0.5" posición-y="0" posición-z="0" Rotación-x=" 0" Rotación-y= "0"
Rotación-z= "0" Recurso=" Imagen.png" />
< Componentevisual3D name=" MostrarVideos" escala="" posición-
x="0.5" posición-y="0" posición-z="0" Rotación-x=" 0" Rotación-y= "0"
Rotación-z= "0" Recurso= "Video.mp3" />
<Boton3D name=" Confirmar" posición-x="0" posición-y="-1" posición-
z="0" Rotación-x=" 0" Rotación-y= "0" Rotación-z= "0" ancho="9"
alto="3" />
</panel 3D>

```

```

<cuimodel>
<panel3D name=" Subinterfaz3- Ejercicio" posición-x="0" posición-y="0"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="1280" alto="720" >
<Label3D name=" Mostrar_Tiempo" posición-x="1" posición-y="1"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="200" alto="100" familiatipográfica=" Arial"
colorfuente="255,255,255" tamañofuente="14" colorfondo="0,0,255" />
<Label3D name=" Mostrar_Puntos" posición-x="3" posición-y="1"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="200" alto="100" familiatipográfica=" Arial"
colorfuente="255,255,255" tamañofuente="14" colorfondo="0,0,255" />
<panel3D name=" Mostrar_Escena" posición-x="0" posición-y="0"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="800" alto="500" >
<Boton3D name="" posición-x="0" posición-y="-1" posición-z="0"
Rotación-x= "0" Rotación-y= "0" Rotación-z= "0" ancho="9" alto="3" />
</panel 3D>
</cuimodel>

```

```

<cuimodel>
<panel3D name=" Subinterfaz4-Ejercicio" posición-x="0" posición-y="0"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="1280" alto="720" >
<Label3D name=" Generar_Informe" posición-x="0" posición-y="0"
posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="200" alto="100" familiatipográfica=" Arial"
colorfuente="255,255,255" tamañofuente="14" colorfondo="0,0,255" />
<Boton3D name=" Continuar_Entrenando" posición-x="0" posición-y="-
0.5" posición-z="0" Rotación-x= "0" Rotación-y= "0" Rotación-z= "0"
ancho="9" alto="3" />
</panel 3D>
</cuimodel>

```

7. Fase IV: Desarrollo e implementación

En este capítulo se evidenciará el desarrollo un prototipo nivel básico para el módulo de entrenamiento, utilizando unity3D que es un motor gráfico principalmente enfocado en el desarrollo de videojuegos y aplicaciones de realidad aumentada, Natnet que es un SDK enfocado a la transmisión de datos generados por software y hardware a NaturalPoint a través de la red.

7.1 Creación de objetos rígidos

La creación de los objetos rígidos se hizo en el sistema optitrack que inicialmente de debe calibrar utilizando las barras de rastro figura 31 y asignar un plano de trabajo por medio del plano de calibración figura 32, y consecutivamente se coloca dentro del área de trabajo el objeto a rastrear con las respectivas esferas reflectivas y se crea el objeto rígido asignándole un ID.



Figura 31 Barras de calibración



Figura 32 Plano de calibración

Nota: Imágenes tomadas de (“OptiTrack - Calibration Tools,” n.d.)

7.2 Streaming.

El Streaming del sistema optitrack hacia el aplicativo donde se está ejecutando el módulo prototipado se realiza a través de la librería NatNetSDK, la cual facilita la transmisión de datos. Sin embargo, es necesario hacer un tratamiento de datos entre un sistema y el otro debido a el sistema de coordenadas y la forma de calcular la matriz de transformación de cada objeto rígido.

En total se rastrean y se envían por medio del streaming 2 objetos rígidos (Pointer, Gafas.).

7.3 Funcionamiento de Natnet.

NatNet contiene dentro de su SDK ejemplos implementados en diferentes lenguajes que involucran básicamente la creación y funcionamiento de un servidor y un cliente, en este proyecto, el módulo prototipado será el cliente, ya que es el que va a recibir los datos de los objetos rígidos proporcionados por el sistema optitrack (servidor) a través de Ethernet o Wifi.

Al momento de configurar el cliente es necesario ajustar la IP del servidor y el puerto a usado para el Streaming de datos, el cual por defecto se usa el 16000. Estos ajustes se realizan sobre un script base de ejemplo que contiene la librería. En la figura 33 se muestran las configuraciones necesarias.

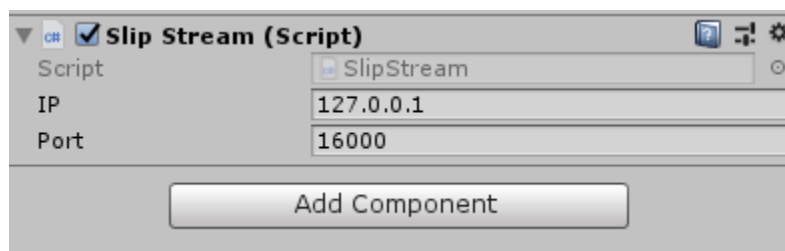


Figura 33 Ejemplo de Streaming

Este script es el encargado de recibir el archivo XML enviado a través de protocolo UDP. Sin embargo, también es necesario usar otro script, el cual se encarga de desempaquetar y asignar los

valores desempaquetados a un objeto nuevo o predefinido, usando el ID, posición XYZ y las rotaciones en cuaternios.

7.4 Tratamiento de datos.

Debido a diferentes causas, es necesario realizar un tratamiento a la información de los objetos rígidos proporcionada por el sistema optitrack. Las principales causas de esto es la diferencia en el sistema de coordenadas entre el sistema optitrack y Unity3D.

Como se muestra en la figura 34, el módulo de administración usa un sistema de coordenadas de mano derecha y Unity3D uno de mano izquierda, por esta razón es necesario realizar un tratamiento a la matriz de transformación de los objetos. Para solucionar esto, se realiza una inversión de posición y rotación en el eje Y ($y = -y$, $qy = -qy$). Este cambio es necesario realizarlo a todos los objetos rígidos recibidos por el Streaming, ya que, si no se realiza, las rotaciones y traslaciones en tiempo real se moverán en ejes incorrectos.

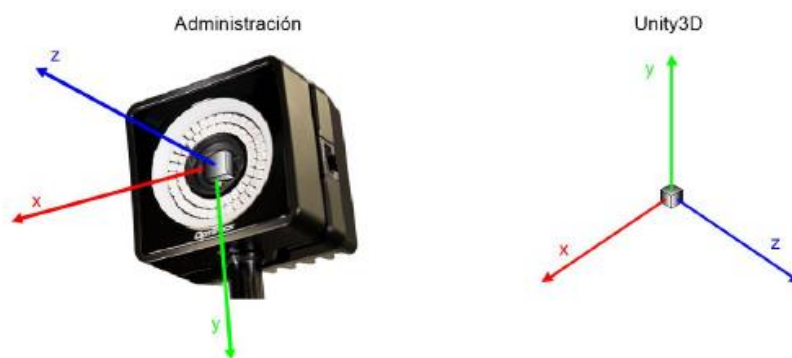


Figura 34 Sistema de coordenadas. Tomada de (PINZÓN DÍAZ, 2017)

7.5 Interfaz de Usuario.

El diseño que se ejecutó en el capítulo 6 de la interfaz de usuario fue el utilizado al momento de codificar el prototipo.

Componentes:

La interfaz se implementó nativamente en Unity3D y consiste en una ventana (*Canvas*) con:

1. Botones que dan acceso a información del módulo de entrenamiento.
2. Textos que permiten mostrarle información al usuario.
3. Imágenes que facilitan la interacción con el usuario.
4. Materiales que son utilizados para definir como se muestran los objetos (texturas, colores)

7.6 Escenas jugables

Para las escenas jugables se utilizaron los *GameObjects* que trae por defecto Unity3D, también se usaron elementos desarrollados en el marco del proyecto de investigación “Realidad aumentada para guiar la perforación de túnel tibial y femoral en reconstrucción de ligamento cruzado anterior”, desarrollado en el grupo de investigación Interfaz de la Escuela de Diseño Industrial.



Figura 35 Objeto

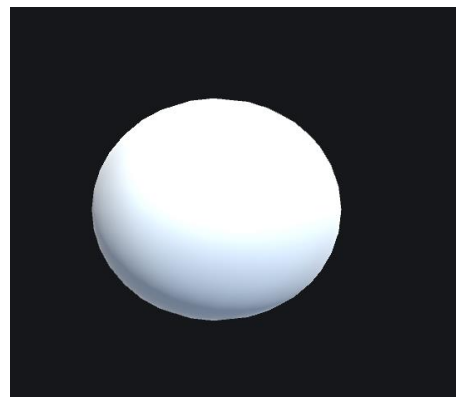


Figura 36 Objetivo



Figura 37 Herramienta

Además, estos objetos son controlados por medio de scripts creados en lenguaje C#, permitiendo activar/desactivar eventos y modificar propiedades.

7.7 Modelos finales de la interfaz de usuario.

A continuación se muestran los modelos finales de la interfaz de usuario diseñada y consecutivamente prototipada.

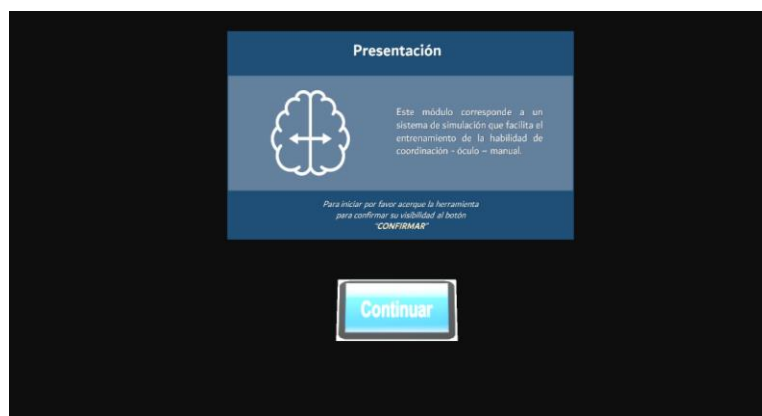


Figura 38 Subinterfaz 1

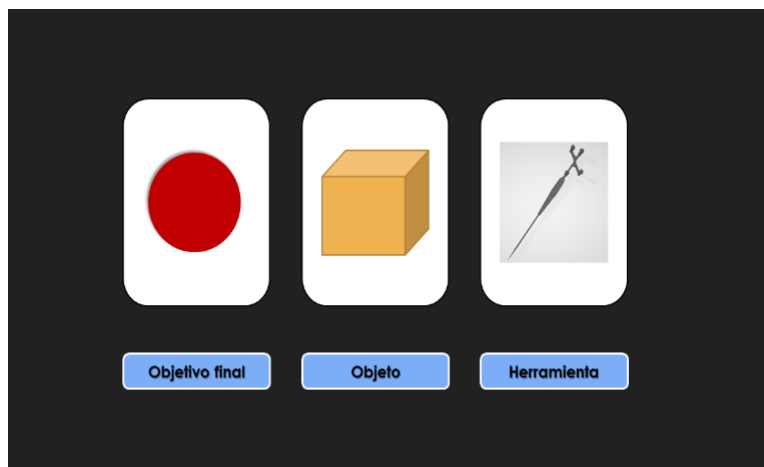


Figura 39 Subinterfaz 2



Figura 40 Subinterfaz 3

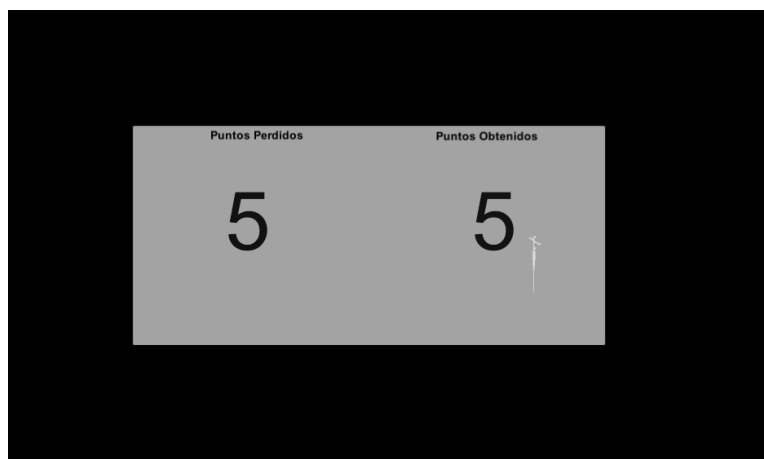


Figura 41 Subinterfaz 4

8. Fase V: Pruebas

Las pruebas técnicas de rendimiento se le realizaron al módulo haciendo uso del profiler incorporado en Unity3D, el cual ofrece información sobre CPU y memoria. A continuación, se mostrarán los respectivos resultados que se obtuvieron al ejecutar el plan de pruebas técnicas que se encuentra en el apéndice H.

8.1 Uso de CPU.

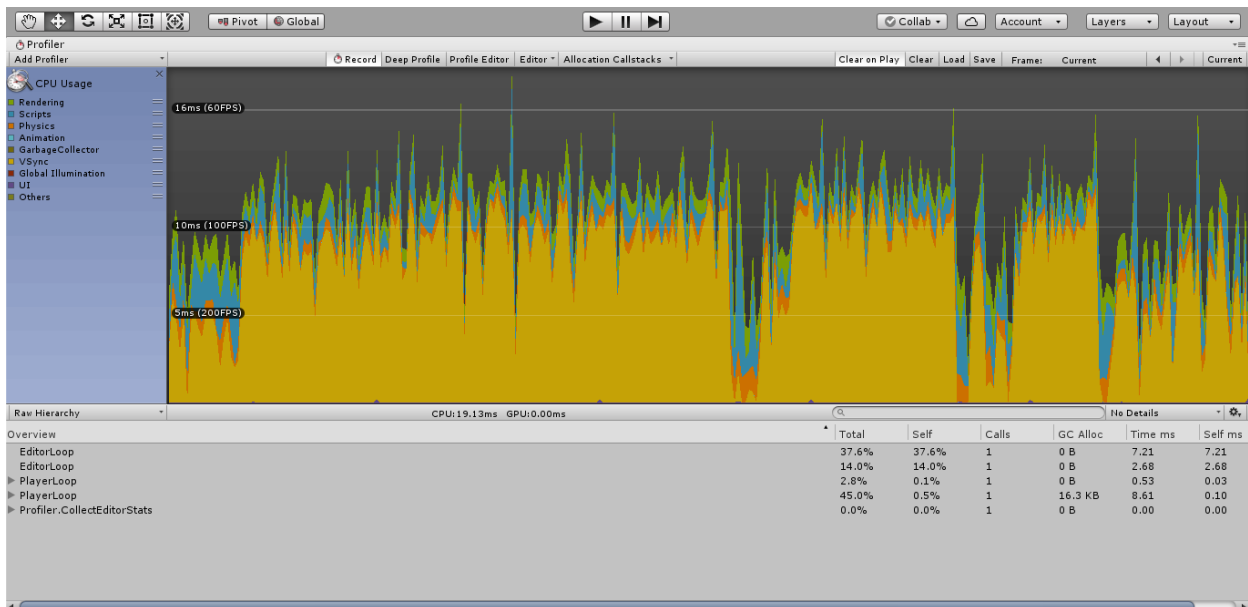


Figura 42 Uso de CPU

En la figura 42 se observa que el uso de la CPU se distribuye principalmente en el renderizado (verde), las físicas (naranja), los scripts (azul), la iluminación (rojo) y otros (amarillo), animación (azul claro) e interfaz de usuario (morado), que corresponde principalmente a la captura de frames para el profiler y al procesamiento interno de la escena. Adicionalmente se observa que la frecuencia de actualización de la imagen oscila principalmente entre los 100 y 200 FPS garantizando una buena fluidez, además se mantiene el uso de la CPU en alrededor del 48%.

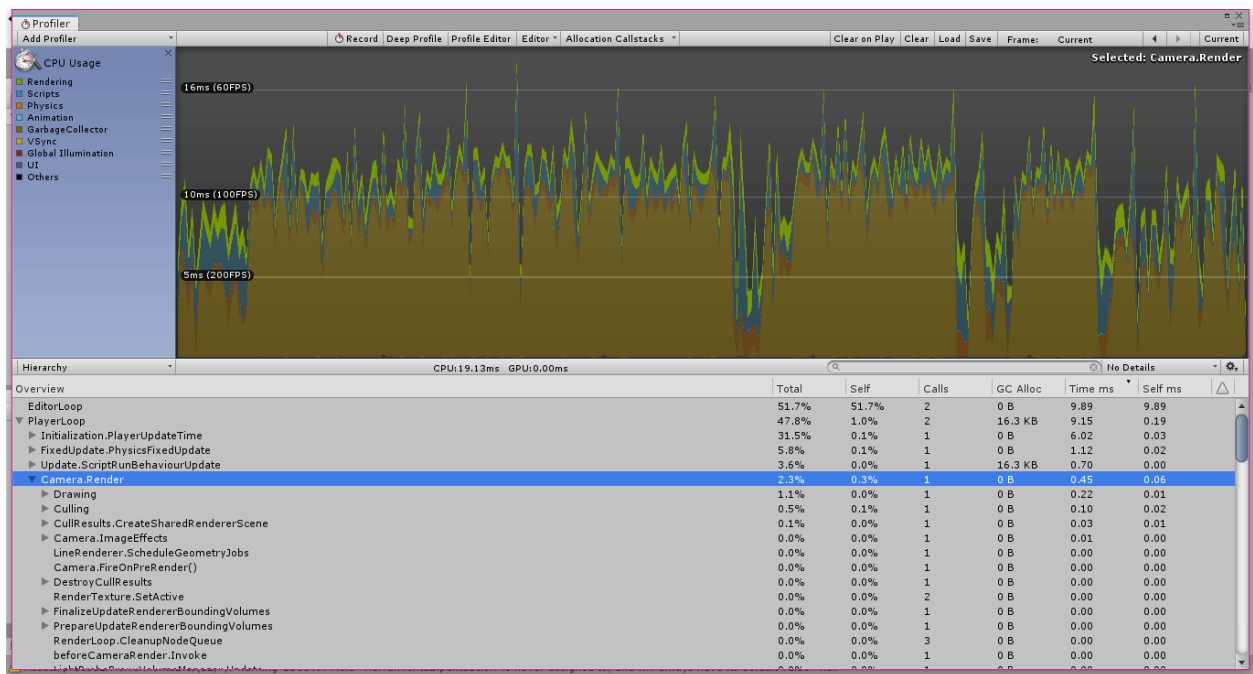


Figura 43 Uso de CPU por porcentaje de uso.

Como se observa en la figura 44, el 2.3% del tiempo se gasta en la función Camera.Render, encargada de llamar funciones exclusivas de dibujo y selección.

8.2 Memoria.

Durante la ejecución del módulo se usa aproximadamente 0.74 MB como se observa en la figura 44. Unity reserva grupos de memoria para unas asignaciones con el fin de evitar pedirle al OS memoria con demasiada frecuencia. Esto se muestra en 4 grupos:

Unity: Cantidad de memoria rastreada por las asignaciones en el código nativo de Unity.

Mono: El tamaño de la pila completa y el tamaño de la pila utilizada, además en esta memoria se hace la recolección de basura.

GfxDriver: La memoria utilizada por texturas, shaders, meshdata, rendertargets.d.

FMOD: El uso de memoria estimado para los controladores de Audio.

Profiler: Memoria usada para la profilerdate.

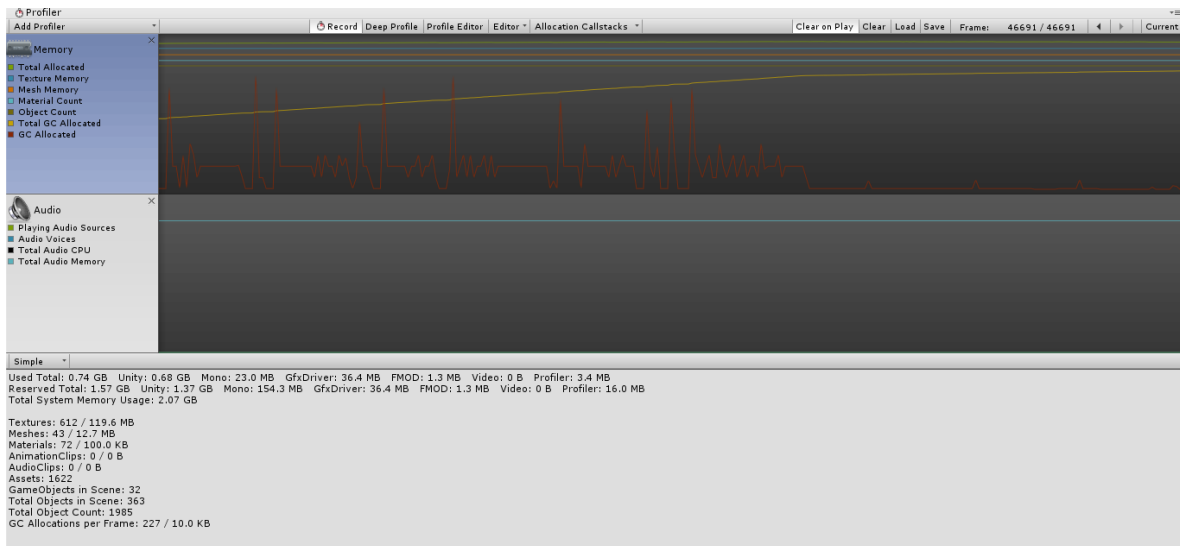


Figura 44 Uso de memoria.

9. Conclusiones

El resultado del análisis de literatura permitió crear un listado de habilidades técnicas que debe desarrollar el cirujano ortopedista antes de realizar a una fijación transpedicular.

Posteriormente se sometió el listado a evaluación con cirujanos ortopedistas y así a establecer la habilidad técnica coordinación óculo-manual para la colocación de tornillos transpedicular en un ambiente quirúrgico.

Se aplicó un enfoque basado en modelos para el diseño de la UI, se encontró la necesidad de incluir nuevos elementos de interfaz, para lo cual se propusieron y verificaron nuevas reglas de transformación.

En este trabajo se proponen un conjunto de reglas de transformación que fueron evaluadas y orientan el proceso de generación de UI, haciéndolas entendibles y sencillas para estudiantes o ingenieros que no tengan conocimiento en nociones de diseño de interfaz gráfica de usuario.

Se realizó el diseño de desarrollo basado en modelos para el módulo de entrenamiento, donde se estandarizó la UI, para que sea flexibles a cambios, buscando que la inclusión de nuevos ejercicios no genere inconsistencias en el módulo y promoviendo la rapidez en el proceso de desarrollo.

Se hicieron pruebas técnicas de rendimiento con el objetivo de determinar el rendimiento de la CPU y memoria cuando se está ejecutando el prototipo en un equipo de bajo recursos.

El trabajo interdisciplinario realizado a lo largo del proyecto de investigación fue de gran ayuda para abordar el problema expuesto y ofreció un escenario de aprendizaje que aporta el desarrollo de experiencia profesional.

10. Recomendaciones.

Se sugiere la creación de una herramienta que permita la generación automatizada de interfaces de usuario abstractas y concretas, que permita la lectura de modelos en CTT, capaz de generar interfaces gráficas y generación de código XML.

Se sugiere continuar con el desarrollo del prototipo anexando nuevos ejercicios y niveles de dificultad hasta lograr un entrenamiento del 100% de la habilidad.

Se recomienda la realización de pruebas con usuarios para validar la interacción del módulo con usuarios reales.

Se sugiere la inclusión de un modelo 3D de columna vertebral para incluir un ejercicio que simula la identificación de punto inicial y posteriormente el avance a través del pedículo, permitiendo evaluar de forma general cual ha sido el avance de los estudiantes entrenados, además de diseñar nuevos ejercicios.

Referencias Bibliográficas.

- Balme, L., Demeure, A., Barralon, N., Coutaz, J., & Calvary, G. (2004). CAMELEON-RT: A Software Architecture Reference Model for Distributed, Migratable, and Plastic User Interfaces, 291–302. https://doi.org/10.1007/978-3-540-30473-9_28
- Balsamiq. Rápido, efectivo y divertido software de wireframing. | Balsamiq. (n.d.). Retrieved October 25, 2018, from <https://balsamiq.com/>
- Bergamasca, M., Bardy, B., & Gopher, D. (2013). Skill training in multimodal virtual environments.
- Bergeson, R. K., Schwend, R. M., DeLucia, T., Silva, S. R., Smith, J. E., & Avilucea, F. R. (2008). How Accurately Do Novice Surgeons Place Thoracic Pedicle Screws With the Free Hand Technique? *Spine*, 33(15), E501–E507. <https://doi.org/10.1097/BRS.0b013e31817b61af>
- Bollati, V. A., Leone, H., Gonnet, S., & Cuenca Pletsch, L. (n.d.). *DESARROLLO DE SOFTWARE DIRIGIDO POR MODELOS EN ENTORNOS ÁGILES*. Retrieved from <http://ria.utn.edu.ar/bitstream/handle/123456789/1502/Trabajo1886-BollatiGonnetCuencaLeoneCREADY.pdf?sequence=1&isAllowed=y>
- Calvary, G., Coutaz, J., Thevenin, D., Bouillon, L., Florins, M., Limbourg, Q., ... Santoro, C. (2002). The CAMELEON Reference Framework. *R&D Project IST-2000-30104*, 60. [https://doi.org/10.1016/S2095-3119\(16\)61498-3](https://doi.org/10.1016/S2095-3119(16)61498-3)
- Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., & Vanderdonckt, J. (2003). A

- unifying reference framework for multi-target user interfaces*. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=92DA7F00C02EC36ECC3F730CD03626A3?doi=10.1.1.100.4512&rep=rep1&type=pdf>
- Calvary, G., Dery-Pinna, A.-M., Occello, A., Renevier-Gonin, P., & Riveill, M. (2014). ARPN Journal of Systems and Software At the Cross-Roads between Human-Computer Interaction and Model Driven Engineering, 4(3). Retrieved from <http://www.scientific-journals.org>
- Carlos Serna-Ojeda, J., Borunda-Nava, D., Domínguez-Cherit, G., Médicas Nutrición Salvador Zubirán, C., & Domínguez Cherit, G. (2012). La simulación en medicina. La situación en México. *Cir*, 8080(3), 301–305.
- CENTIC, U. I. de S.-D. de S. de I. (n.d.). Universidad Industrial de Santander. Retrieved November 6, 2018, from <https://www.uis.edu.co/webUIS/es/academia/facultades/salud/labSimulacion.html>
- Centre Maribor, M., & Merc, S. M. (2014). Error rate of multi-level rapid prototyping trajectories for pedicle screw placement in lumbar and sacral spine. *Chinese Journal of Traumatology*, 17(2615), 261–266. <https://doi.org/10.3760/cma.j.issn.1008-1275.2014.05.003>
- Chaparro Lemus, L. O., & Gómez Estupiñan, J. F. (2012). Una visión del desarrollo de software utilizando modelos, 14. Retrieved from <https://revistas.uis.edu.co/index.php/revistagti/article/view/2818/3060>
- Chavez Diego, E. A., & Valdivia Fierro, R. (2015). *Ejercicios motrices en el desarrollo de la coordinación óculo manual de los niños y niñas de 4 y 5 años en la institución educativa inicial N° 568 Pucarumi*. Universidad Nacional de Huancavelica. Retrieved from <http://repositorio.unh.edu.pe/bitstream/handle/UNH/573/TP - UNH INIC>.

030.pdf?sequence=1&isAllowed=y

Contreras Jordán, O. R. (1998). *Didáctica de la educación física : un enfoque constructivista*.

INDE Publicaciones. Retrieved from

<https://books.google.es/books?id=pFpdIfg130IC&printsec=frontcover&hl=es#v=onepage&q&f=false>

Cuppens, E., Raymaekers, C., & Coninx, K. (n.d.). *A Model-Based Design Process for*

Interactive Virtual Environments. Retrieved from

https://bibliotecavirtual.uis.edu.co:2142/content/pdf/10.1007%2F11752707_19.pdf

France, R., & Rumpe, B. (2007). Model-Driven Development of Complex Software: A Research

Roadmap. *Workshop on the Future of Software Engineering (FOSE 2007), at the 29th*

International Conference on Software Engineering (ICSE 2007), Minneapolis, Minnesota,

USA, (May 2007), 37–54. <https://doi.org/10.1109/FOSE.2007.14>

Frankel, D. S. (2004). *MDA Journal*. Retrieved from www.eclipse.org/emf

García Gonzáles, V., Burgos Flores, J., Del Olmo, T., Barrios Pitarque, C., Hevia Sierra, E.,

Antón Rodríguez, L. M., & De Blas Beorlegui, G. (2013). MAL POSICIÓN DE

TORNILLOS PEDICULARES LUMBARES QUE PRODUCEN RADICULALGIA

MECÁNICA POSTOPERATORIA CON ESTIMULACIÓN NEUROFISIOLÓGICA

INTRAOPERATORIA NORMAL: EL VALOR DE LA ESTIMULACIÓN

NEUROFISIOLÓGICA DEL TRAYECTO DEL TORNILLO MAU, *12*(4), 322–325.

Giraldo Orozco, W. J. (2010). *Marco de Desarrollo de Sistemas Groupware Interactivos*

Basado en la Integración de Procesos y Notaciones. Retrieved from

<https://www.educacion.gob.es/teseo/imprimirFicheroTesis.do>

Henrique, D., & Karahan, M. (2017). *Motor Skills Training in Orthopedic Sports Medicine*.

<https://doi.org/10.1007/978-3-662-53229-4>

HIIS Laboratory | Home. (n.d.). Retrieved November 25, 2018, from

<http://giove.cnuce.cnr.it/lab/home>

HIIS Laboratory | Research. (n.d.). Retrieved May 16, 2019, from

<https://hiis.isti.cnr.it/lab/research/MARIA/home>

Jiménez Ortega, J., & Alonso Obispo, J. (2007). *Manual de psicomotricidad : teoría,*

exploración, programación y práctica. La Tierra Hoy. Retrieved from

[https://books.google.es/books?hl=es&lr=&id=18c4WWH6TCwC&oi=fnd&pg=PA7&dq=Manual+de+Psicomotricidad+\(Teoría,+Exploración,+Programación+y+Práctica\)&ots=-aaFUusyckj&sig=i9yI-xTgaUXA-IrdJc4LeQ8agBs#v=onepage&q=coordinación+o&f=false](https://books.google.es/books?hl=es&lr=&id=18c4WWH6TCwC&oi=fnd&pg=PA7&dq=Manual+de+Psicomotricidad+(Teoría,+Exploración,+Programación+y+Práctica)&ots=-aaFUusyckj&sig=i9yI-xTgaUXA-IrdJc4LeQ8agBs#v=onepage&q=coordinación+o&f=false)

Manual+de+Psicomotricidad+(Teoría,+Exploración,+Programación+y+Práctica)&ots=-aaFUusyckj&sig=i9yI-xTgaUXA-IrdJc4LeQ8agBs#v=onepage&q=coordinación+o&f=false

aaFUusyckj&sig=i9yI-xTgaUXA-IrdJc4LeQ8agBs#v=onepage&q=coordinación+o&f=false

Jiménez Ortega, J., & Jiménez De La Calle, I. (2010). *Psicomotricidad. Teoría y programación*

para educación infantil, primaria y especial. Madrid.

Karahan, M., & Tuijthof, G. (n.d.). *Arthroscopic Simulation in Skills Training: European*

Initiatives.

Lopez Araoz, A. (2014). Rol e importancia de la simulación en la educación médica y

broncoscópica. *Revista Americana de Medicina Respiratoria*, 4, 362–364.

<https://doi.org/10.7202/1006343ar>

López Jaquero, V. M. (2005). *Interfaces de Usuario Adaptativas Basadas en Modelos y Agentes*

Software. Universidad de Castilla- La Mancha. Retrieved from

<http://www.dsi.uclm.es/personal/VictorManuelLopez/mipagina/archivos/thesis.pdf>

Martínez-Poncede León, A., de la Garza-Castro, O., Morales Ávalos, R., Elizondo Omaña, R.,

Vílchez Cavzos, F., Elizondo Riojas G, ... Guzmán López, S. (2012). *Fijación vertebral por*

vía transpedicular. Importancia de los estudios anatómicos y de imagen. Acta Ortopédica

Mexicana (Vol. 26). Retrieved from

<http://www.medigraphic.com/actaortopedicawww.medigraphic.org.mx>

Montero, F., & López-Jaquero, V. (2007). *IDEALXML: AN INTERACTION DESIGN TOOL A*

Task-based Approach to User Interface Design. Retrieved from

https://bibliotecavirtual.uis.edu.co:2142/content/pdf/10.1007%2F978-1-4020-5820-2_20.pdf

Montero, F., López-Jaquero, V., Vanderdonckt, J., González, P., & Lozano, M. (n.d.). *Solving the Mapping Problem in User Interface Design by Seamless Integration in IDEALXML*.

Retrieved from

http://www.dsi.uclm.es/personal/FranciscoMSimarro/idealxml_archivos/Montero-DSV-IS2005.pdf

Muñoz Arteaga, J., Hernández, Y. C., Bustos Amador, V., Aranda Chavarría, A., Calderón, M.

E., Collazos, C. E., ... David, C. H. (2014). *Temas de Diseño en Interacción Humano-Computadora*.

Muñoz Márquez, F. J. (2007). ACAUI: Abstracción de interfaces de usuario a.

Muñoz Rivera, D. (n.d.). La coordinación y el equilibrio en el área de Educación Física.

Actividades para su desarrollo. Retrieved November 7, 2018, from

<http://www.efdeportes.com/efd130/la-coordinacion-y-el-equilibrio-en-el-area-de-educacion-fisica.htm>

Muñoz Rodríguez, P. E., Albán, A. V., & Muñoz Rodríguez, P. E. (n.d.). Sistema Robótico Guía para la Inserción de Tornillos en Cirugía de Fijación de Columna, *36*(2), 143–154.

<https://doi.org/10.17488/RMIB.36.2.3pdf>

NatNet SDK 3.0 - NaturalPoint Product Documentation Ver 2.0. (n.d.). Retrieved May 11, 2019,

from https://v20.wiki.optitrack.com/index.php?title=NatNet_SDK_3.0

Nóbrega, L., Nunes, N. J., & Coelho, H. (2006). *Mapping ConcurTaskTrees into UML 2.0.*

LNCS (Vol. 3941). Retrieved from

https://bibliotecavirtual.uis.edu.co:2142/content/pdf/10.1007%2F11752707_20.pdf

OptiTrack - Calibration Tools. (n.d.). Retrieved May 11, 2019, from

<https://optitrack.com/products/tools/>

OptiTrack - Motion Capture Systems. (n.d.). Retrieved May 11, 2019, from

<https://optitrack.com/>

Página de inicio del proyecto CAMELEON. (n.d.). Retrieved October 24, 2018, from

<http://giove.isti.cnr.it/projects/cameleon.html>

Pastor, O., Aquino, N., España, S., & Panach, J. I. (2008). Model-Driven Development Piecing

Together the MDA Jigsaw Puzzle 394 *Informatik_Spektrum_31_5_2008*.

<https://doi.org/10.1007/s00287-008-0275-8>

Pestana Tirado, R. Al., Moreno Ballesteros, L. R., & González Di Filippo, A. (n.d.). DOMINIO

Y DESTREZAS DEL CIRUJANO, REVISTA DE CIRUGÍA, SALUD, MEDICINA.

Retrieved from [https://encolombia.com/medicina/revistas-medicas/cirugia/vc-](https://encolombia.com/medicina/revistas-medicas/cirugia/vc-194/ciru19404-dominio/)

[194/ciru19404-dominio/](https://encolombia.com/medicina/revistas-medicas/cirugia/vc-194/ciru19404-dominio/)

PINZÓN DÍAZ, D. S. (2017). *HERRAMIENTA SOFTWARE DE VISUALIZACIÓN PARA ASISTIR LA PERFORACIÓN DEL TÚNEL TIBIAL Y FEMORAL EN RECONSTRUCCIÓN DE LIGAMENTO CRUZADO ANTERIOR (RLCA) USANDO REALIDAD AUMENTADA.*

Real Academia Española. (n.d.). Retrieved November 7, 2018, from <http://www.rae.es/>

Realidad Aumentada : una nueva lente para ver el mundo. (n.d.).

Rigueros Bello, C. (2017). La realidad aumentada: lo que debemos conocer. *Tecnología,*

- Información y Academia*, 5(2), 257–261. Retrieved from <http://revistas.udistrital.edu.co/ojs/index.php/tia/issue/archive>
- Sader, E., Yee, P., & Hodaie, M. (2017). Barriers to Neurosurgical Training in Sub-Saharan Africa: The Need for a Phased Approach to Global Surgery Efforts to Improve Neurosurgical Care. *World Neurosurgery*, 98, 397–402. <https://doi.org/10.1016/j.wneu.2016.07.098>
- Safir, O., Dubrowski, A., Carnahan, H., Mirsky, L., Lin, C., & Backstein, D. (2013). What skills should simulation training in arthroscopy teach residents? *International Journal of Computer Assisted Radiology and Surgery*, 8(6), 945–953. <https://doi.org/10.1007/s11548-013-0833-7>
- Sierra Bueno, D. A., Yáñez Canal, G., & Monroy Allado, R. (2016a). Estudiantes graduados, 109–124.
- Sierra Bueno, D. A., Yáñez Canal, G., & Monroy Allado, R. (2016b). Estudiantes matriculados.
- Stirling, E. R. B., Lewis, T. L., & Ferran, N. A. (2014). Surgical skills simulation in trauma and orthopaedic training. *Journal of Orthopaedic Surgery and Research*, 9, 126. <https://doi.org/10.1186/s13018-014-0126-z>
- Stobbs, N., & Kumar, B. N. (2014). The role of simulation in surgical training. *Otorhinolaryngologist*, 7(3), 169–172. <https://doi.org/10.1007/s12178-014-9209-z>
- Unity - Download Archive. (n.d.). Retrieved May 11, 2019, from <https://unity3d.com/es/get-unity/download/archive>
- UsiXML - Lenguaje de marcado extendido de la interfaz del usuario. (n.d.). Retrieved October 29, 2018, from <http://www.usixml.org/en/what-is-usixml.html?IDC=236>
- Vanderdonckt, J. M., & Bodart, F. (1993). Encapsulating Knowledge For Intelligent Automatic

Interaction Objects Selection, 424–429.

Varas, J., Mejía, R., Riquelme, A., Maluenda, F., Buckel, E., Salinas, J., ... Boza, C. (2012).

Significant transfer of surgical skills obtained with an advanced laparoscopic training program to a laparoscopic jejunojejunostomy in a live porcine model: Feasibility of learning advanced laparoscopy in a general surgery residency. *Surgical Endoscopy and Other Interventional Techniques*, 26(12), 3486–3494. <https://doi.org/10.1007/s00464-012-2391-4>

Wang, V. Y., Chin, C. T., Lu, D. C., Smith, J. S., & Chou, D. (2010). Free-hand thoracic pedicle screws placed by neurosurgery residents: a CT analysis. *European Spine Journal*, 19(5), 821–827. <https://doi.org/10.1007/s00586-010-1293-1>