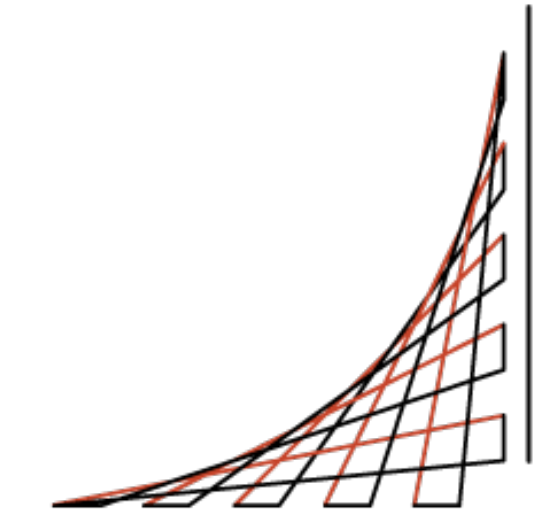

ESCUELA COLOMBIANA DE INGENIERÍA
JULIO GARAVITO

VIGILADA MINEDUCACIÓN

INGENIERÍA DE SISTEMAS

PRESENTADO POR:

Leonardo Castro Susa,
Laura Milena Ramos Bermudez,
Johan Sebastian Peña Peñaloza

PROYECTO DE GRADO:

Implementación de un sistema de aprendizaje de lenguajes de programación en línea para JavaScript y Haskell

---

Director del proyecto:
Daniel Benavides

Agradecimientos

Son los cimientos de nuestro desarrollo, los que con tanto sacrificio y lucha nos han dado las oportunidades, su tiempo y paciencia, con sus aportes invaluables que para toda la vida tendremos, sin ustedes nada de esto seria posible, a quienes primeramente damos el agradecimiento ya que gracias a ellos, somos grandes seres humanos.

Gracias mamá, gracias papá.

A nuestros formadores, a ellos que nos han guiado en este proceso que no ha sido sencillo, y a pesar de los tropiezos, siempre han estado allí para compartir sus conocimientos, y darnos esa guía para poder culminar esta época de estudios de una manera exitosa, a nuestra universidad, gracias por habernos permitido formarnos en ella, gracias a todas las personas que fueron participes de este proceso, ya sea de manera directa o indirecta, gracias a todos ustedes, fueron ustedes los responsables de realizar su pequeño aporte, que el día de hoy se verá reflejado en la culminación de nuestro paso por la universidad.

A nuestro director de proyecto Daniel Benavides, quien desde el inicio del proyecto creyó y confió en nosotros para poder desarrollar y dar un gran alcance a este, por compartirnos su sabiduría, y guiarnos por medio de su experiencia.

Índice general

Índice de figuras

Capítulo 1

Introducción

La educación es el arma más poderosa que puedes usar para cambiar el mundo. -Nelson Mandela.

El campo de la educación al rededor del mundo es un tema que ha llevado a varias discusiones que van desde el tipo de metodologías a utilizar para enseñar como ¿Qué se debe enseñar? ó ¿Cómo se debe enseñar?. Dependiendo de las aptitudes y actitudes personales e individuales de las personas estas metodologías pueden variar y, en torno a esta problemática se ha decidido seguir un estándar de enseñanza en general de tipo alumno - profesor que consiste en la interacción entre una persona que sabe sobre uno o varios temas en específico (profesor) y un receptor que desea aprender sobre esos tópicos (alumno). Con la aparición de las computadoras como elementos de trabajo diario y facilitadores de tareas repetitivas en las grandes industrias, se hizo también necesario implementar metodologías de enseñanza que tienen que estar cada vez más apegadas a la interacción humano computadora y, que de hecho ha generado nuevas metodologías de enseñanza donde los conocimientos son guardados y procesados por computadoras para compartir información, permitiendo quebrar la brecha de conocimiento al nivel de alcance de un computador o dispositivo que posea conexión a internet. Pero, ¿Qué hay actualmente, con respecto a las metodologías de enseñanza relacionadas con los lenguajes de desarrollo y manejo (lenguajes de programación) de estos dispositivos?.

En los últimos años, en las instituciones de educación superior se ha presentado una deserción cada vez mayor en los programas de tecnologías de la información (TI), en donde la rama de aprendizaje de lenguajes de programación se ha visto afectada al momento de que sus estudiantes adquieran y retengan los cocimientos, ya que estos no son estimulados para su óptimo

aprendizaje y ha llevado a que los estudiantes pierdan el interés y cambien de carrera profesional. En la Escuela Colombiana de Ingeniería Julio Garavito esta problemática ha sido analizada durante el proyecto de grado a cargo de este documento y, según nuestros estudios, con el pasar de los semestres es cada vez más evidente esta problemática al aumentar el numero de estudiantes de ingeniería de sistemas que cancelan o pierden las primeras asignaturas de o relacionadas con programación, esta problemática se ha intentado abordar desde distintas perspectivas y áreas.

El proyecto presentado a continuación, denominado ipp5 es, en primera instancia un desarrollo liderado por el área académica de la Escuela Colombiana de Ingeniería Julio Garavito para apoyar el aprendizaje temprano de lenguajes de programación en el primer curso de la línea de lógica de la carrera de Ingeniería de Sistemas. Según el plan de estudios de la carrera mencionada anteriormente, no se ven materias relacionadas con programación hasta segundo semestre en la asignatura Programación Imperativa Básica (PIMB), pero se ha decidido dar un acercamiento a la misma en la primera materia de la línea de lógica: Modelos Matemáticos para la Informática (MMIN). IPP5 es una plataforma de tipo aplicación web que busca dar un acercamiento de manera interactiva al usuario para estimular la atención y retención de conocimientos mediante módulos de aprendizaje y que sirve como herramienta de apoyo a los docentes para transmitir conocimiento. Para desarrollar estos módulos se inició con un esfuerzo para realizar investigaciones sobre el aprendizaje en el ser humano desde su contexto biológico y evolutivo, pasando por las etapas de desarrollo humano, las distintas formas de aprendizaje y diversas teorías sobre aprendizaje y metodologías de enseñanza; se continuó con una investigación sobre la enseñanza en ingeniería así como el cómo se enseña programación y diversas alternativas que ya existen. Toda esta investigación se realizó en conjunto con el estudio sobre la interacción humano computadora, y la educación virtual para hacer el desarrollo final de la plataforma ipp5.

El presente documento pretende plasmar todas las teorías y artículos estudiados en el desarrollo de investigación del proyecto y de la plataforma web, así como exponer también de manera técnica las especificaciones y elementos que conformaron el desarrollo de la herramienta de apoyo (ipp5) desde la estructura de su contenido hasta su arquitectura. El documento se estructura de la siguiente manera: El capítulo Aprendizaje tratará sobre las investigaciones realizadas que hablan de los principios de aprendizaje de los seres humanos desde el punto de vista biológico y evolutivo que permitirán establecer un contexto apegado al entendimiento de obtención y retención de información de los seres humanos como seres pensantes. Luego, en el capítu-

lo Human Interaction Computer se expondrán los estudios relacionados a la interacción entre los humanos y las computadoras, las cuales, a pesar de no ser seres orgánicos pueden mimetizar tales comportamientos para generar lazos de interacción parecidos a los que realizan los humanos. Después, en el capítulo Aprendizaje en la Programación cerramos el contexto del aprendizaje sobre el punto de vista de la programación presentando 4 metodologías que han sido probadas y desarrolladas a lo largo de los años y que han tenido éxito al rededor del mundo, cada una con sus fallos y posibles soluciones expuestos de manera explícita. Posterior a esto, en los capítulos .Estructura Del Aprendizaje Aplicada a los módulos de IPP5z .Arquitectura"se explicarán las estructuras de conocimiento sugeridas tras las investigaciones de los capítulos anteriores así como los requerimientos técnicos y tecnológicos referentes a la solución implementada. Finalmente, se presentarán las conclusiones referentes a nuestros aportes personales con respecto al desarrollo de la plataforma y su investigación así como el trabajo futuro sugerido tanto para la plataforma y como para el ámbito de la investigación sobre metodologías de aprendizaje.

Capítulo 2

Aprendizaje

Este capítulo aborda diversas investigaciones, teorías y otras discusiones, las cuales son de gran interés para el desarrollo y entendimiento de la problemática a tratar. Se da inicio al capítulo mostrando la investigación de los métodos biológicos de aprendizaje del ser humano abarcando el contexto evolutivo, pasando por las etapas de desarrollo humano y las distintas formas y teorías de aprendizaje. Se continuó con una investigación sobre la enseñanza en ingeniería así como las metodologías de enseñanza de programación y diversas alternativas existentes también digitales apoyándonos en el estudio la interacción humano computadora, y la educación virtual. A destacar y resaltar las teorías investigadas y resumidas de Edward Thorndike y Karl Popper, las cuales se plasman de manera y fueron base y pilar para el desarrollo de la plataforma.

2.1. ¿Cómo aprende la humanidad?

.A lo largo de los años, muchos científicos e investigadores se han dado a la tarea de analizar el proceso de aprendizaje en las personas. Son muchos los resultados y los debates que estos han generado debido a las diferentes opiniones que se tienen acerca de cómo se origina el aprendizaje, sus diferentes teorías y principios, cómo es que nuestro cerebro trabaja para obtener el conocimiento y generar diversas opciones para almacenarlo y clasificarlo."[12]

¿Qué nos diferencia a los seres humanos de los animales? El razonamiento. Los animales son capaces de construir sus hogares, buscar su propio alimento, defenderse de sus predadores, criar a sus hijos y transportarse de un lugar, pero entonces que hace que el aprendizaje humano sea diferente, definimos el aprendizaje, como un cambio relativamente permanente en la conducta como resultado de la experiencia, en efecto la distribución de nuestro cerebro y las

conexiones neuronales son la explicación biológica del aprendizaje.

El desarrollo del cerebro comienza a finales del primer mes del desarrollo prenatal; durante el segundo trimestre se forman la mayoría de las neuronas que una persona tendrá a lo largo de su vida. Poco después de nacer, la reproducción de las neuronas se incrementa de una forma espectacular, esto es para que los niños tengan más sinapsis que los adultos. A lo largo de la niñez y la adolescencia, ocurre la poda sináptica, que permite que las neuronas que no se usan sean desechadas sin causar daño alguno. El cerebro se sigue desarrollando en la juventud y la edad adulta, aunque en menor dimensión que en la niñez. Nuestros genes contribuyen en gran parte a nuestro desarrollo neurológico. De ellos dependen, ciertas enfermedades o trastornos hereditarios.

Existen algunos períodos críticos en el desarrollo del cerebro. No se sabe con certeza cual es la edad correcta para aprender o desarrollar alguna habilidad, como el lenguaje. Los científicos han descubierto que una persona puede seguir aprendiendo a lo largo de toda su vida, sin importar la edad.

Si bien es cierto que nuestros primeros años de vida son esenciales para el desarrollo del cerebro, de igual forma lo es permanecer alejados de cualquier cosa que pueda dañarlo, sustancias nocivas, radiación, etc. Esto no significa que únicamente en los primeros años seamos capaces de aprender algo, ya que podemos hacerlo a cualquier edad. Esto es algo que los educadores deben tomar en cuenta, no solo en la niñez se aprende, nunca terminamos de aprender.[25]

2.2. Principios del Aprendizaje: Edward Thorndike

Psicólogo y pedagogo estadounidense, considerado un antecesor de la psicología conductista estadounidense fue profesor de psicología durante más de treinta años en el Teachers College de Columbia, Estados Unidos. Lo que más atrajo su interés fue la teoría del aprendizaje, y se cuenta entre los importantes precursores del conductismo, identificado varios principios didácticos también conocidos como leyes de aprendizaje, que son generalmente aplicables a este proceso. Estos principios han sido descubiertos, probados, y utilizados en situaciones prácticas y proporcionan una idea acerca de las formas en que las personas aprenden de manera efectiva.[23]

2.2.1. Disposición

Debido a que aprender es un proceso activo, implica un grado alto de concentración y deseo, así mismo se aprende mejor cuando se está en óptimas condiciones tanto física, mental y emocional para ello, el caso opuesto ocurre si no ve razones para hacerlo, el maestro debe lograr que los estudiantes estén preparados para aprender, crear interés demostrando los beneficios del trabajo, y proporcionar un reto mental o físico continuo, son por lo general las responsabilidades de un profesor. Si los estudiantes tienen un propósito sólido, un objetivo claro, y una razón definida para aprender algo, presentan un mayor progreso que si carecen de motivación. Junto con esto es importante el estado del estudiante, en cuanto a la parte física debe tener un buen descanso, buena salud, en general las necesidades básicas de los estudiantes tienen que estar satisfechas para poder ser capaces de aprender, y en la parte emocional, un estudiante con horarios saturados, distraídos por responsabilidades exteriores, intereses, o preocupaciones, tendrán un desempeño inferior.

2.2.2. Ejercicio

Este principio nos indica que aquellas cosas que se realizan con una cierta repetición son más duraderas, se ha comprobado que los estudiantes aprenden más y retienen mejor la información cuando se realiza la acción de repetición más a menudo, esto debido a que la memoria humana es falible, la mente raramente puede retener, evaluar y aplicar conceptos o practicas, después de una sola sesión, se debe mostrar lo aplicado, y se debe repetir los elementos importantes en intervalos de tiempo razonables para dar la oportunidad de practicar con un objetivo.

2.2.3. Efecto

El principio del efecto, se fundamenta en la relación emocional del estudiante, el principio dicta que el aprendizaje se facilita cuando es acompañado de un sentimiento agradable, el estudiante buscara continuar con aquellos estímulos que le producen el sentimiento de satisfacción, esto nos da a entender los refuerzos positivos sin mejores para encaminar y motivar al estudiante, con esto se busca la prudencia al castigar dentro del aula. No necesariamente todas las lecciones tiene que ser exitosas, tampoco implica que el estudiante las domine por completo, pero cada sesión debería tener elementos que dejen al estudiante con buenos sentimientos, esto aumentara las probabilidades de éxito.

2.2.4. Primacía

El estado de ser primicia, crea una fuerte impresión, lo primero que se aprende crea una fuerte huella que es difícil de borrar, por ende lo que se enseña debe ser correcto desde su inicio, es más difícil desenseñar, esta primera experiencia debe ser positiva, funcional y correcta en todo sentido, así mismo presentarse le en orden lógico, paso a paso, verificando el aprendizaje. Cuando la lección es aprendida por partes o es confusa en su imposición inicial, tiene que volver a ser aprendida. Este proceso podría generar más confusiones, y consumir más tiempo.

2.2.5. Recencia

Cuanto más recientemente hubieren sido aprendido las cosas, son más memorables, y así mismo en viceversa cuanto más pase el tiempo entre la lección, sera más difícil recordarlo, así mismo para mantenerlo vigente se debe dar una revisión frecuente, crear y leer resúmenes que permitan fijar el material en la mente, repite, reafirma, o re-enfatiza puntos importantes al final de una lección para ayudar a el estudiante a recordarlos. El principio de recencia a menudo determina la secuencia de las clases dentro de un curso de instrucción.

2.2.6. Intensidad

Cuanto más intenso sea el material enseñado, entre más vívido, dramático, claro o apasiónate sea, se convierte en una experiencia más que en una lección, esto hace que sea más perdurable, el principio implica que el estudiante aprenderá más de algo real que un sustituto, un estudiante podría aprender más de un experimento en vivo que de su teoría, así mismo con películas que su guión, En contraste a la instrucción práctica, el aula impone limitaciones en el realismo de las lecciones. Un profesor debe entonces utilizar la imaginación para acercarse a la realidad tanto como sea posible. Las lecciones de aula pueden beneficiarse de una gran variedad de ayudas institucionales para generar realismo, motivar a aprender y retar a los estudiantes. Los profesores deberían enfatizar puntos importantes de la instrucción con gestos, representaciones y con la voz. Las exposiciones, parodias, y juegos de roles aumentan considerablemente la experiencia de aprendizaje de los estudiantes. Los ejemplos, las analogías, y las experiencias personales también ayudan a el aprendizaje a cobrar vida. Los profesores deberían hacer uso total de los sentidos.

2.2.7. Libertad

Cuando las lecciones son estudiadas libremente se asimilan mejor, por el contrario entre más sea un estudiante obligado, más difícil será aprender, asimilar e implementar lo estudiado, entre más libre sea un individuo mayor sera su avance intelectual. Siendo un proceso activo los estudiantes deben tener libertar: elección, acción, y libertad para asumir las consecuencias de actuar, siendo estas las tres grandes libertades, si ninguna es dada a los estudiantes tendrán poco interés en aprender.

2.2.8. Requisito

La ley o principio de requisito, indica que debemos tener algo para obtener o hacer algo, puede ser una capacidad, habilidad, instrumento o cualquier elemento que nos ayude a aprender u obtener algo. Cuando no se tienen las herramientas o las habilidades se imposibilita hacer la tarea, tal como puede ser dibujar, si no tengo lápiz y hoja, no podría dibujar, así mismo según lo que se requiera [24]

2.3. Karl Popper: Teoría del conocimiento objetivo

"El conocimiento objetivo es incuestionable, tanto en ciencia como en filosofía"

Popper define el conocimiento objetivo, como el conocimiento de la realidad, sobre el entorno que nos rodea, este también debe estar avalado por estudios científicos, investigaciones, experimentos, sus resultados y la repetición de los mismos en cualquier parte y por cualquier persona y debe llegar a la misma conclusión, esto sin lugar a dudas.

El conocimiento es una creencia, que es verdadera (si y solo si acontece en la realidad) y está justificada "(ya sea mediante la percepción, la lógica o a través de los principios de racionalidad: causa-efecto, uniformidad de la naturaleza...)".

Popper plantea una pregunta: ¿Es posible la objetividad? Esta pregunta queda sin respuesta puesto que debemos definir que es alcanzar la objetividad. En algunos casos, principalmente son los hechos, pero los hechos son en realidad, los hechos los percibimos a través de los sentidos, pero todo esto es subjetivo; está sujeta a los límites sensoriales que como humanos tenemos, la realidad que ahora mismo observamos no es tal y como la vemos, en primera instancia todos los colores que podemos ver no son tal y como los vemos en

la realidad, los lugares que vemos vacíos no están vacíos o incluso en un caso extremo, la mesa que veo delante de mí no está. En un principio lo de los colores y lo del espacio que teóricamente vemos vacío serían conocimiento objetivo según la definición dada de objetividad la mayoría de la gente verá lo mismo que nosotros (aunque realmente no tenemos pruebas para saber que el mundo que ve el resto de la gente es igual al nuestro, ¿cómo sabemos que los colores que ven son iguales que los que vemos nosotros?) Pero, ¿Qué la mayoría de la gente vea una cosa es sinónimo de objetividad? No estoy de acuerdo, si la especie humana sufriese esquizofrenia irremediablemente, o todos fuéramos bulímicos, ¿La realidad distorsionada sería la objetiva? Es una subjetividad compartida no por ello objetividad. Los colores que vemos están sujetos a nuestro sentido de la vista y a la capacidad de nuestro cerebro de formar la realidad con ellos, igual seríamos capaces mediante máquinas especiales de incrementar este conocimiento pero seguimos sin saber con seguridad que ese aparato está sintiendo todos los colores o que nosotros no lo estamos malinterpretando con nuestros límites humanos. Los lugares que vemos vacíos y que la gente afirma que es un conocimiento objetivo, no lo están, están llenos de microorganismos y partículas que somos incapaces de percibir. También con aparatos este conocimiento se puede aumentar e igual el verdadero conocimiento objetivo, el que acontece en la realidad se puede alcanzar, ya que es un "hecho" que delante de mí hay 4000 partículas de polvo, 400 especies de microorganismos... Sin embargo, explicando uno de los ejemplos puestos anteriormente en el que afirmaba que tal vez la mesa que tengo delante de mí no está ahí, hay otro argumento en contra de la práctica mayoría de los conocimientos que existen, y es que la realidad que observamos no es real, que seamos cerebros en una cubeta y nos recreen esta realidad, o que estemos en la Matrix que para el caso es lo mismo. En este escenario defender la existencia del conocimiento puede ser complicado, pero es posible defenderlo bajo el argumento de que lo que vemos es nuestra realidad justificada por nuestros sentidos. Pero esto le quita toda objetividad al conocimiento, nada de eso acontece en la realidad, el conocimiento objetivo debe ser aquel irrefutable, la propia realidad.

Por tal razón al no poder concluir correctamente y en realidad el conocimiento objetivo, de este se plantean dos problemas.

El planteamiento epistemológico de Popper empieza con dos problemas fundamentales: el problema de la inducción y el problema de la demarcación, cada una retoma planteamientos diferentes al de Hume y al de Kant [6]

2.3.1. El problema de la inducción

La inducción, es una forma de razonamiento que consiste en establecer una ley o conclusión general a partir de la observación de hechos o casos particulares. Es la inferencia que se hace de enunciados particulares a enunciados universales, observaciones a la formulación de hipótesis o teorías. No hay validez de los enunciados universales, cuestión que no es posible demostrar.

La inducción presenta la idea de la posibilidad de aprender inductivamente de la experiencia, es decir de la idea intuitiva de que a medida que se van acumulando los datos que confirman una teoría, aumenta la probabilidad de que esta sea verdadera, esto es rechazado por Popper al argumentar que no existe fundamento lógico ni psicológico para la probabilidad inductiva, imposibilidad de alcanzar inductivamente conocimiento cierto o incluso probable (en sentido matemático) a partir de los siempre de la experiencia aun mas cuando está es limitada, pues siempre podría haber una nueva observación que refutara la proposición.

Muchos científicos siguen creyendo en la inducción porque consideran que la ciencia natural se caracteriza por el método inductivo, es decir por un método que parte de largas series de observaciones y experimentos y se basa en ellos, creen que solo el método inductivo puede suministrar un criterio de demarcación entre ciencia y especulación metafísica o pseudociencia, y que este método permite establecer o verificar como seguras, o casi seguras o muy probables, las teorías.

Más si lo analizamos cuidadosamente los casos aparentes de inducción no son sino el método de ensayo y error, de conjeturas y refutaciones, es decir el método hipotético - deductivo. En primer lugar, una hipótesis no es generada por los datos sino inventada por los científicos para dar cuenta de las observaciones que forman parte del problema que la hipótesis trata de resolver, una vez obtenida evidencia pertinente y supongamos que a favor de la hipótesis, no se induce su probabilidad, sino que se evalúa o juzga la hipótesis a la luz de la evidencia y de argumentos críticos que ayudan a decidir sobre su aceptabilidad. Eso simula la inducción, pues se salta de las observaciones (particulares) a la hipótesis (general), como si dijéramos en dirección inductiva, pero no hay razonamiento o inferencia sino estimación y decisión en dicho paso. Por lo tanto, el método científico es un conjunto sistemático de reglas metodológicas, de decisiones dirigidas a asegurar el contraste de una hipótesis, la crítica o falsación, es decir, proponga hipótesis someterlas a pruebas y lógica. [3]

2.3.2. El problema de la demarcación

Popper es contradictor de la inducción como método de aprendizaje. Él plantea que la conjetura es antes que la observación. La inducción plantea que primero se observa y después induce el conocimiento. En el principio de la demarcación Popper nos habla de lo que es y no es ciencia, en general Popper argumenta que el humano tiene un proceso de creatividad (conjetura) y criticismo, Si queremos progresar en el conocimiento tenemos que ir en búsqueda de leyes de la naturaleza, y formular y contrastar enunciados universales, Ha de ser posible refutar por la experiencia un sistema científico empírico.

Según Karl Popper el problema de la demarcación hace referencia a la diferencia o los límites que existen y que se enmarcan en el concepto ciencia, es el problema de saber entender realmente qué es la experiencia, es poder comprender la diferencia entre ciencia y religión.

Popper hace referencia a tener una concepción empirista y positivista de la ciencia, de la razón y de la filosofía. Por ende, para él es incorrecto pensar que se puede afirmar proposiciones o teorías generales partiendo de teorías pequeñas, como se dice coloquialmente, no se puede generalizar ya que siempre surgirá una excepción.

A raíz de esto, Popper señaló la demarcación como un problema en la filosofía de la ciencia. Lo que lo llevó a proponer el "falsacionismo", el cual hace referencia a la forma de determinar si una teoría es científica o no, es decir, si una teoría es falsable, entonces es científica pero si no es falsable, entonces esta no sería ciencia.

Estos problemas son planteados para saber como deben proceder los científicos para descubrir leyes o teorías irrefutables, nosotros deberíamos poner a prueba nuestras ideas e invenciones, las criticamos y desechamos lo que está mal hecho y lo volvemos a intentar. El trasfondo de esto es que se avanza y se aprende a base de ENSAYO y ERROR. Incluso desde el conductismo se dice que hay dos formas de aprender:

1 APROXIMACIONES SUCESIVAS: Nadie aprende de un solo golpe. Es un proceso de aprendizaje continuo que nunca se acaba. No se puede ir directamente, hay que ir con aproximaciones sucesivas, poco a poco.

2 ENSAYO - ERROR: El equívoco tiene un gran potencial de aprendizaje. El crear conocimiento, el producir conocimiento tiene la premisa de Ensayo-Error.[19]

2.4. Aprendizaje Cognitivo

Este tema del cognitivismo abarca todo sobre la teoría del aprendizaje cognitivo basada en diferentes autores tales como Piget, Vigotsky, Ausubel, Bruner donde centran sus estudios principalmente en procesos mentales y procesos cognitivos básicos explicando la asociación de entrada de estímulos, procesamiento del individuo y la ejecución de las respuestas, estas respuestas se pueden traducir a decisiones y realizar acciones eficaces por parte del individuo teniendo en cuenta los procesos internos del individuo al procesar estos estímulos. El aprendizaje cognitivo señala la importancia de ciertas actividades mentales como las expectativas, el pensamiento, la atención o el recuerdo en el proceso de aprender, también ha propuesto modelos sobre el aprendizaje basados en las teorías sobre procesamiento de la información, otorgando especial importancia al papel de la memoria y el razonamiento dentro de los procesos conductuales centrándose principalmente en el estudio del aprendizaje humano.[17]

Como se decía anteriormente tiene como precursores barios autores los cuales plantearon y desarrollaron las siguientes teorías:

2.4.1. Jean Piaget: Teoría del desarrollo cognitivo

La cognición es definida como los procesos internos de la mente que conducen al conocimiento, y también es una red de estructuras mentales creadas por el individuo para dar un sentido a las experiencias que se hace mediante la memoria, simbolización, solución de problemas y hasta en los sueños. Ha lo anterior se le definió como esquemas que son patrones organizados de pensamientos o comportamientos, que tienden a cambiar con la edad en su gran mayoría lo hace en la infancia, si se generan o modifican estos esquemas se generan nuevos conocimientos.Para esto se necesitan unas funciones mentales como la adaptación que posibilita el cambio de esquemas, a través de la construcción de nuevos esquemas mediante la experiencia directa, mediante la realización de dos procesos muy importantes la *asimilación* que es la interpretación del mundo externo a nuestros esquemas y la *acomodación* que consiste en cambiar un esquema existente o crear uno nuevo para lidiar con un objeto o situación en particular. Un ejemplo claro para entender esto es la situación en la que un niño en un zoológico observa por primera vez a un camello, él asimila y asocia sus conocimientos previos con el camello y dice que es un caballo porque encuentra características similares como que tiene cuatro(4) patas, es peludo y alto, pero si el niño observa con más cuidado o es corregido se puede dar cuenta de que no es un caballo porque tiene características distintas como que la cola es más corta, tiene una joroba y no

relincha; con esto ya se aplica el proceso de acomodación y crea un nuevo esquema para el camello separándolo del esquema ya existente del caballo.

Esta teoría se centra en que existen diferentes etapas del desarrollo cognitivo y son afectados por los cambios cualitativos que tienen lugar en la formación mental de la persona, desde el nacimiento hasta la madurez. Estas etapas pueden verse afectadas por agentes externos tales como el ambiente cultural y genético lo que hace que cada niño pueda superar estas etapas en distintos tiempos. Se considera que es invariable y universal ya que todos los individuos de cualquier parte del mundo deben pasar por estas etapas y por obligación se hace de una manera cronológica relacionada con el desarrollo biológico del individuo. Estas etapas cognitivas son clasificadas en cuatro(4) fases distintas las cuales son:

La Etapa sensorio-motora (0-2 años) una de las principales características de esta etapa es el egocentrismo ya que el individuo es incapaz de distinguir su perspectiva y la de los demás, el individuo es estimulado, guiado y controlado por los reflejos innatos como ser humano, como lo son las acciones motrices y percepciones sensoriales, esta etapa se dividió en sub-etapas, la primera es *esquemas reflejos (0^o-$1^o mes$)* durante este periodo el individuo utiliza sus conductas reflejas para responder de forma indiscriminada a cualquier estimulo exterior, la segunda sub-etapa es la de *reacciones circulares primarias($1^o mes$-$4^o mes$)* ya puede realizar conductas volitarías para realizar acciones que le resultan placenteras, estas acciones son primarias porque van motivadas hacia sus propios cuerpos y necesidades básicas; la tercera sub-etapa es la de *reacciones circulares secundarias($4^o mes$-$8^o mes$)* en esta sub-etapa los individuos repiten acciones placenteras involucrando objetos ya que desarrollan habilidades físicas para alcanzar, manipular y tomar los objetos de su alrededor; la cuarta sub-etapa es coordinación de las *reacciones circulares secundarias($8^o mes$-$12^o mes$)* en esta etapa se producen dos cambios cognitivos, primero los individuos ya pueden realizar conductas intencionalmente gracias a la práctica y a la habilidad con una diversidad de esquemas que puede ordenar deliberadamente esto sumando a que ya es capaz de comprender la permanencia del objeto sin la necesidad de verlo pudiendo así buscar objetos escondidos; luego tenemos la quinta sub-etapa la de *reacciones circulares terciarias($12^o mes$-$18^o mes$)* en esta etapa pueden resolver problemas sencillos del sistema sensorio motor y por último está la sub-etapa de las *combinaciones mentales($18^o mes$-2años)* se desarrollan las habilidades de elaborar representaciones mentales lo que les permite resolver problemas a través de medios simbólicos en vez de solucionarlos por acierto y error, recuerdan y copian conductas de modelos que no están presentes físicamente.

La Etapa pre-operacional (2-7 años) los individuos ya aprenden a interac-

tuar con su ambiente mediante la comunicación con ayuda del lenguaje, esta etapa se caracteriza por egocentrismo pre-operacional donde el individuo se comunica y piensa sobre el mismo, siendo incapaz de ver o pensar desde otra perspectiva asumiendo que las otras personas ven, sientes y oyen lo mismo que el, pueden hacer representaciones simbólicas de objetos como dibujos o jugar con objetos que representan algo real lo que les permite tener una definición más sofisticada del objeto, al finalizar esta etapa el individuo ya diferencia que las plantas y los animales son seres vivos y son los únicos que sienten y tienen necesidades.

La Etapa de operaciones concretas (7-11 años) Aquí ya el individuo tiene la capacidad para convertir esos proceso de razonamiento a procesos lógicos lo que permite que pueda aplicarlo a problemas en concreto o reales en su ambiente, estos problemas son con objetos físicos y no abstractos, puede reconocer contradicciones y puede recorrer los procesos mentales lo que hace que pueda recordar, hacer ordenamientos mentales de conjuntos y clasificación, entendiendo también los conceptos de casualidad, espacio, tiempo y velocidad mejorando así su desarrollo espacial.

La Etapa de operaciones formales (11-15 años) el individuo ya es capaz de extraer conocimientos concretos, tiene pensamiento más allá de la realidad concreta mediante hipótesis, puede desarrollar sentimientos idealistas formando así continuamente su personalidad y sus conceptos morales como persona, en la adolescencia tiende a pensar en la audiencia imaginaria lo que conlleva a que se preocupe como lo ven los demás y busca llamar la atención porque cree que lo que él percibe de si mismo los demás también lo hacen de la misma manera, esto conlleva a que se sienta único y diferente, llegando a pensar que los demás no lo pueden entender. [18] A medida que trascurre el tiempo este pensamiento de ser único se va desvaneciendo porque deja de centrarse en si para formar relaciones que conllevan a compartir con otros individuos que tiene las mismas características sociales, lo que hace que disminuya el egocentrismo para poder pensar en los demás y como grupo mejorar.

2.4.2. Jerome Brunner: Aprendizaje por descubrimiento

Lo fundamental de la teoría es la construcción del conocimiento mediante la inmersión del individuo en situaciones de aprendizaje problemáticas, la finalidad de esto es que el individuo aprenda a realizar el procesamiento activo de la información a su manera, siendo capaz de seleccionar, procesar y organizar la información de forma particular. Se plantea que el conocimiento

no se puede adquirir mediante la memorización a través de la repetición de conceptos, por el contrario el conocimiento se adquiere mediante la interacción del individuo con consigo mismo y el entorno, motivado por curiosidad aunque en esta motivación puede darse Extrínseca o Intrínsecamente.Este aprendizaje por descubrimiento se puede dar por tres(3) formas de descubrimiento: *Descubrimiento inductivo:*El descubrimiento inductivo es el método de aprendizaje en el que el individuo busca de manera autónoma el conocimiento, en un proceso que parte de la observación y el análisis de las características de un concepto, habilidad o competencia a aprender y así puede aprender parte de un problema o desafío concreto para llegar a lo general o abstracto reorganizando y recolectando los datos para generar nuevas categorías. *Descubrimiento Deductivo:*Es la combinación o puesta en relación de ideas generales, con el fin de llegar a enunciados específicos, como en la construcción de un silogismo o la generación de deducción proporciones lógicas complejas del lenguaje. *Descubrimiento Transductivo:* En el descubrimiento pensamiento transductor el individuo relaciona varios elementos particulares y comprende que son similares en varias características particulares, siendo de gran utilidad para motivar y potenciar el pensamiento imaginativo o artístico. En esta teoría se presentan tres sistemas de procesamiento de la información que son: *Modelo Enactivo:* Se refiere al aprendizaje kinestésico, a través de la ejecución de una acción. *Modelo Icónico:* Se refiere al aprendizaje visual, representaciones mentales, imágenes. *Modelo Simbólico:* Se refiere a las representaciones abstractas o de símbolos. Con esta teoría se busca que el individuo aprenda a aprender a su manera haciendo que el individuo se percate de la estructura del contenido que se quiere aprender y de la relaciones con sus elementos, facilitando con ello la retención del conocimiento a largo plazo. [18]

2.4.3. Lev Vigotsky: Teoría del aprendizaje sociocultural

Lev Vigotsky centro su estudio de la teoría en las interacciones sociales del triángulo educativo que relaciona el quien aprende, quien enseñanza y contenidos e interacciones sociales que intervienen en el proceso de enseñanza aprendizaje. Esta teoría radica principalmente en que el aprendizaje de cada individuo depende del medio en el cual se desarrolla estudiando la interrelación individuo - sociedad, para comprender su comportamiento de acuerdo a su entorno y que el desarrollo de sus habilidades mentales aparecen en la interacción con otras personas y artefactos culturales. El conocimiento no se pasa de uno a otro, se construye por medio de operaciones cognoscitivas que

se inducen en la interacción social.

Zona de desarrollo próximo es la distancia entre el nivel de desarrollo efectivo del alumno (aquello que es capaz de hacer por sí solo) y el nivel de desarrollo potencial (aquello que sería capaz de hacer con la ayuda de un adulto o un compañero más capaz).

El Andamiaje se refiere a que el maestro es un modelo de apoyo y soporte para el proceso de aprendizaje del individuo con menor conocimiento en todas las actividades que se necesiten para lograr adquirir ciertas habilidades cognitivas necesarias para desarrollar una determinado proceso para realizar una tarea, esto se logra exponiendo al estudiante a los ejercicios que le permitan desarrollar estos conocimientos nuevos, y la idea es que en cualquier problema el maestro pueda ser una guía para que el aprendiz pueda desarrollar su nivel real a un nivel potencial de conocimiento en donde ya no necesite la guía del tutor. Este concepto de andamiaje se logra a través de dos(2) niveles progresivos y se desarrolla en cualquier ámbito sociocultural educativo como lo es el familiar; el primer nivel es la ejecución del procedimiento por parte del experto y el segundo la ejecución guiada del procedimiento por parte del aprendiz.

2.5. Aprendizaje basado en problemas

El aprendizaje basado en problemas (PBL) es un enfoque de instrucción que se ha utilizado con éxito completamente por más de 30 años y continúa ganando aceptación en múltiples disciplinas. Es un enfoque instructivo (y curricular) centrado en el alumno que permite a los alumnos conducir investigación, integre la teoría y la práctica, y aplique el conocimiento y las habilidades para desarrollar un solución a un problema definido. Este resumen presenta una breve historia, seguida de una discusión de las similitudes y diferencias entre ABP y otros enfoques basados en experiencia enseñando, e identificando algunos de los desafíos que se avecinan para PBL. Palabras clave: aprendizaje basado en problemas, definiciones, características Los estudiantes deben tener la responsabilidad de su propio aprendizaje. El PBL es un enfoque centrado en el alumno: los alumnos se involucran con el problema, El aprendizaje debe integrarse desde una amplia gama de disciplinas o asignaturas La colaboración es esencial. En el mundo después de la escuela, la mayoría de los estudiantes se encontrarán en trabajos donde necesidad de compartir información y trabajar productiva mente con otros. [20]

2.6. Educación en Ingeniería

La pedagogía dominante para la educación de ingeniería sigue siendo "mostrar y hablar", a pesar de la gran cantidad de investigaciones sobre educación que demuestran su ineficacia. En los últimos años, la profesión de ingeniería y los organismos responsables de acreditar los programas de ingeniería han pedido un cambio. Este documento discute la aplicación del aprendizaje basado en problemas y basado en proyectos a la educación de ingeniería, examina la diferencia entre ellos. Revisa algunos ejemplos de dónde se han utilizado hasta la fecha y analiza la efectividad y la relevancia de cada método para la educación de ingeniería. La profesión de ingeniería moderna trata constantemente con la incertidumbre, con datos incompletos y demandas competitivas (a menudo contradictorias) por parte de los clientes, gobiernos, grupos ambientalistas y el público en general. Requiere habilidades en relaciones humanas, así como competencia técnica. Mientras intentas incorporar más habilidades "humanas.[en] su base de conocimiento y práctica profesional, los ingenieros de hoy también deben hacer frente a continuos cambio tecnológico y organizacional en el lugar de trabajo. Además ellos debe hacer frente a las realidades comerciales de la práctica industrial en el moderno mundo, así como las consecuencias legales de cada decisión profesional que hacer. A pesar de estos desafíos, el modelo predominante de educación en ingeniería sigue siendo similar a la practicada en la década de 1950 - "tiza y hablar", con grandes las clases y la entrega basada en clases de una sola disciplina son la norma, particularmente en los primeros años de estudio. Desarrollos en el aprendizaje centrado en el estudiante como El aprendizaje basado en problemas y basado en proyectos hasta ahora ha tenido relativamente poco impacto en la educación de ingeniería convencional. Este documento comienza por examinar los problemas críticos para la educación de ingeniería y su impacto en requisitos de acreditación. Luego mira la naturaleza de ambos problemas Los métodos actuales de enseñanza se basan en una serie de pasos comienza con un problema o situación identificada que dirige el área de los estudiantes o contexto de estudio. Se confía en la investigación iniciada por el alumno para que el alumno progrese a través del proyecto, así como para su propio aprendizaje. Requiere altos niveles de iniciativa estudiantil, los estudiantes necesitan desarrollar motivación y habilidades de organización.[14]

2.7. Conclusiones y Discusión

En las teorías mostradas podemos llegar ciertos análisis biológicos, que son esenciales en el aprendizaje no solo humano, sino de todos los animales. Estas teorías se alinean a ciertos factores importantes como lo son los estímulos, la competencia, la intensidad entre otros, para lograr dar alcance y aporte de esto a la plataforma, se decide darle bastante importancia y estudio al problema de la demarcación que plantea Karl Popper.
De la demarcación, damos cabida al ensayo y error, al hecho de que el ser humano no obtiene conocimiento directamente sino que realiza experimentos y a partir de estos genera conocimiento y logra dar un entendimiento de este mismo.
Dentro de nuestros apuntes debemos rescatar todo el poder de entendimiento de Karl Popper y sus pensamientos reflejados en el área de aprendizaje y de obtención de conocimiento en el que plantea la contradicción de la inducción como método de aprendizaje considerando que debe haber conjetura antes que observación alegando que la inducción de conocimiento se obtiene al observar y que este elemento es primordial antes de inferir un conocimiento ya que se induce conocimiento después de observar, cuestión que personalmente contradecimos con el ejemplo de las matemáticas, ya que estas no las podemos ver pero hemos sido capaces de entenderlas e interpretarlas como seres humanos sin necesidad de observarlas; a este hecho podemos agregar que Karl Popper hace referencia al término observación como aquella interacción que conlleva a la visualización de objetos tangibles e interacciones sensoriales.

Capítulo 3

Human Interaction Computer (HIC)

Los conceptos que forman el área de la interacción persona - computadora, abarcan las áreas que estudian el intercambio de información mediante software entre las personas y las computadoras. Esta disciplina se encarga del diseño, evaluación e implementación de los aparatos tecnológicos interactivos, estudiando el mayor número de casos que les pueda llegar a afectar. El objetivo es que el intercambio sea más eficiente: minimizar errores, incrementar la satisfacción, disminuir la frustración y, en definitiva, hacer más productivas las tareas que rodean a las personas y los computadores.[8]

La investigación en este campo es muy complicada, la recompensa una vez conseguido el objetivo de búsqueda es muy gratificante. Es muy importante diseñar sistemas que sean efectivos, eficientes, sencillos y amenos a la hora de utilizarlos Los gráficos por computadora nacieron de la utilización del CRT y de las primeras utilizaciones del lápiz óptico. Eso llevó al desarrollo de técnicas pioneras para la interacción persona - computador, a partir de aquel momento se ha continuado trabajando en este campo, creando y mejorando los algoritmos y el hardware que permiten mostrar y manipular objetos con mucho más realismo. Dentro de los campos de estudio se encuentran los métodos para diseñar nuevas interfaces de ordenadores, y así optimizar el diseño de una propiedad que se desee, como por ejemplo la capacidad de aprendizaje o la eficiencia de uso. Métodos para implementar las interfaces, por ejemplo, por medio de bibliotecas informáticas. Métodos para evaluar y comparar interfaces con respecto a sus propiedades, como por ejemplo su usabilidad. Métodos para estudiar el uso de los ordenadores y sus implicaciones socioculturales. Modelos y teorías sobre el uso humano de los ordenadores, así como marcos de referencia conceptuales para el diseño de interfaces, como modelos de usuario cognitivas, la teoría de la actividad o

consideraciones metodológicas sobre el uso de ordenadores en humanos. Así mismo presenta principios de diseño, los cuales se deben adaptar según tipo de usuario.[16]

La Association for Computing Machinery (ACM) define la interacción humano - computadora como ünа disciplina relacionada con el diseño, evaluación e implementación de sistemas informáticos interactivos para uso humano y con el estudio de los principales fenómenos que los rodean "Thomas T la define La interacción humano - computadora es una disciplina relacionada con el diseño, la evaluación y la implementación de sistemas informáticos interactivos para uso humano y con el estudio de los principales fenómenos que los rodean . -Thomas T. Hewett y otros, 1992

El origen del campo se centra en una pregunta ¿ las tecnologías son aptas para el aprendizaje, utilizables, útiles, confiables, comprensibles y éticas?, la evaluación de las tecnologías nos involucran, y es grato cuando satisfacen las necesidades de las personas; Si bien es cierto HCI siempre se ha asociado con la estabilidad, su fin es mucho mas amplio, y continua creciendo con el tiempo, y muy importante viene siendo la parte psicológica, sociológica y cultural, para determinar que rasgos se mantienen y como la tecnología puede aportar ventajas, como podemos nivelar el campo de acción, y que estilos y tecnologías sobreviven al paso del tiempo.

3.1. El humano

Ël humano tiene limitaciones naturales para el procesamiento de información que impactan en el diseño de sistemas interactivos (por ejemplo, un cirujano no puede controlar una cámara mientras realiza una operación) ∴ Estas limitaciones o características se consideran y a menudo se estudian para el diseño de productos en HCI. Además, hay factores inherentes a la condición humana, como el cansancio, el aburrimiento y el enojo, que también se deben tomar en cuenta. Esto es particularmente relevante cuando las computadoras se usan en condiciones extremas o críticas (por ejemplo, por un piloto de avión). De igual manera la información que proviene de las computadoras y del ambiente se captura por diversos canales como el auditivo, el visual, y el movimiento, y se guarda de manera temporal o definitiva en nuestra memoria de corto o largo plazo. Todos los datos que recibimos a través de los canales mencionados se procesan de manera consciente o inconsciente y a través de mecanismos muy complejos, nos permiten resolver problemas, razonar, adquirir habilidades y conocimiento, y hasta cometer errores. De igual manera las emociones impactan en nuestras actividades mentales y/o físicas, e incluso, si son muy fuertes, pueden llevar nuestras capacidades al límite.

Aun cuando los humanos compartimos diferentes habilidades y capacidades las diferencias individuales se deben considerar para diseñar productos que puedan ser usados. [7]

3.2. La Computadora

Los dispositivos clásicos de entrada, como el teclado y el ratón, y de salida, como el monitor o proyector, el audio y la impresora, fueron objeto de intenso estudio en los ochenta, pero desde entonces ha habido innovaciones notables, como el reconocimiento del habla, las pantallas táctiles, las plumas digitales y, recientemente, el auge de los sensores o bio - señales en el caso de los dispositivos de entrada, y las pantallas públicas y el papel digital en los de salida. De igual forma, también ha habido un desarrollo importante, aunque en menor grado, en otro tipo de dispositivos de entrada/salida, como los controles físicos, la realidad virtual y aumentada y olfativa.

3.3. La interacción

La interacción se entiende como un diálogo entre la computadora y el humano.Debido a la gran diversidad de personas y de contextos en los que se usan las computadoras, continuamente se proponen métodos y técnicas para entender mejor cómo es o cómo debería ser este proceso. De igual manera, una vez que se ha diseñado un sistema interactivo, se tiene que verificar que efectivamente el producto cumple con su propósito. Por ejemplo, si el sistema es un software para aprendizaje de matemáticas se tiene que evaluar si este permite aprender igual o mejor que con el apoyo de un profesor tradicional o con el de otro software diseñado para el mismo propósito.

La evaluación no solo abarca la efectividad del sistema interactivo sino también otros aspectos como eficiencia, eficacia, satisfacción al momento de usarlo e incluso su adopción fina El diseño de computadoras y software es un proceso inherente al IHC en el que intervienen diversos factores como:

3.3.1. ¿Para quien se diseña?

El diseño se debe orientar según las personas a quienes se diseña, son niños, adultos, jóvenes, sus limitaciones, visuales, motoras, auditivas, cognitivas, o sus capacidades y habilidades, esto debe guiar, desde su diseño para lograr una mejor interacción, claramente debemos estudiar como se comporta cada persona y como sera su reacción ante diversos estímulos, así como a un niño no podemos presentarle tablas y gráficas, a un hombre de negocios no

se le dará una serie de bonitas y coloridas pantallas con diseño minimalista, establecer cuántos usuarios se necesitan para realizar la tarea y determinar quiénes deben ser los usuarios apropiados, alguien que nunca ha usado la interfaz, y que no usará la interfaz en el futuro, probablemente no sea un usuario válido, esta razón de para quien, y su nivel cognitivo es el pilar base para el diseño y una buena HCI.

3.3.2. Que se diseñara

La actividad en la cual se realizara y se desempeñara con la computadora es clave de la razón de ser, ya sea un aula de clases, o una sala de cirugía, cada elemento debe orientarse a su labor final, no siempre la interacción se dará para ingresar datos como un oficinista, en ocasiones tienen múltiples usos y variaciones como sus respuestas.

3.3.3. En donde se usara - contexto

El lugar en el cual se realizará la actividad (por ejemplo, un quirófano, sentado en la sala o en una oficina o un vehículo de carreras). Generalmente, la interacción se da en un lugar donde los aspectos sociales y el contexto organizacional tienen un efecto importante tanto en la persona como en el sistema, esto genera que no solo debemos pensar en que y para quien, sino también en dónde, para lograr adaptar a la necesidad y aumentar y posibilidad del éxito y de rehúso.

3.3.4. Diseño Iterativo

Tras aplicar los anteriores puntos el diseño se debe re evaluar constantemente, de forma que permita encontrar los aspectos positivos, negativos, a potenciar y a mejorar, se debe diseñar la interfaz de usuario, prueba, analizar resultados, repetir, esto con el fin de poder tener claro y acercar cada vez mas a una interfaz óptima y adecuada para el usuario.

3.4. Como estudiar y entender HCI

Despues de entender la necesidad e importancia de HCI, En 1988, el Comité Ejecutivo de SIGCHI patrocinó una investigación que condujo, en colaboración con varios pensadores clave en paneles relacionados, a un documento publicado en 1992 titulado Çurrículos de ACM para la interacción humano - computadora".

Se centró específicamente en la noción de que los programas educativos deberían ser cada vez más "HCI orientados", no "HCI centrado", y que HCI podría caracterizarse como más de una "sensibilidad.[en] la necesidad de un conjunto específico de métodos y herramientas, que, aunque en desarrollo, estaban menos establecidos que en la mayoría de las otras partes de la informática. Como resultado, los autores argumentaron, La educación de HCI necesitaba continuar incluyendo métodos y herramientas apropiados para el dominio que fueran familiares a los contextos tradicionales de los que emergía el HCI: principalmente ciencias de la computación, psicología e ingeniería. También argumentaron que nosotros, como educadores y estudiantes, necesitábamos adoptar nuevas perspectivas y nuevas áreas de enfoque; las áreas convocadas fueron asignaturas típicamente enseñadas en escuelas de arte y diseño, y en la información y ciencias de la biblioteca. Dicho esto, los autores favorecieron que HCI se vincule fuertemente con los programas de informática y sistemas de información, refiriéndose a estas como "disciplinas básicasz afirmó: "Por lo tanto, hemos diseñado marcos para una orientación de HCI dentro de un programa de informática y dentro de un sistema de información de gestión programa.[11]

Así mismo, ya existen programas de HCI : De la universidad Politécnica de Cataluña https://www.upc.edu/content/grau/guiadocent/pdf/esp/804029

3.5. Principios del Diseño de Visualización

Christopher Wickens definió 13 principios de diseño de visualización en su libro .[A]n Introduction to Human Factors Engineering.", Estos principios de percepción humana y procesamiento de la información se pueden utilizar para crear un diseño de visualización efectivo. Una reducción en los errores, una reducción en el tiempo de capacitación requerido, un aumento en la eficiencia y un aumento en la satisfacción del usuario son algunos de los muchos beneficios potenciales que se pueden lograr a través de la utilización de estos principios, ciertos principios pueden no ser aplicables a diferentes pantallas o situaciones. Algunos principios pueden parecer conflictivos, y no existe una solución simple para decir que un principio es más importante que otro. Los principios se pueden adaptar a un diseño o situación específicos.[27]

3.5.1. Principios perceptivos

Esta sección se abordan principios que buscan dar un primer conocimiento de algo por medio de las impresiones que comunican los sentidos a partir de allí interpreta la información proveniente de estímulos, pensamientos y sen-

timientos, a partir de su experiencia previa, de manera lógica o significativa, todo esto buscando generar la mayor posibilidad de retención de información y generación de conocimiento posible.[27]

- Haga que las pantallas sean legibles (o audibles). La legibilidad de una pantalla es crítica y necesaria para diseñar una pantalla utilizable. Si los caracteres u objetos que se muestran no pueden entenderse, entonces el usuario no puede hacer uso de ellos de forma efectiva.

- Evite las ambigüedades. No solicite al usuario que determine el nivel de una variable sobre la base de una única variable sensorial (por ejemplo, color, tamaño, volumen). Estas variables sensoriales pueden contener muchos niveles posibles.

- Procesamiento de arriba hacia abajo. Es probable que las señales se perciban e interpreten de acuerdo con lo que se espera según la experiencia del usuario. Si se presenta una señal contraria a las expectativas del usuario, es posible que deba presentarse más evidencia física de esa señal para garantizar que se entienda correctamente.

- Ganancia de redundancia. Si una señal se presenta más de una vez, es más probable que se entienda correctamente. Esto puede hacerse presentando la señal en formas físicas alternativas (por ejemplo, color y forma, voz e impresión, etc.), ya que la redundancia no implica repetición. Un semáforo es un buen ejemplo de redundancia, ya que el color y la posición son redundantes.

- La similitud causa confusión: usa elementos distinguibles. Las señales que parecen ser similares probablemente se confundan. La relación de características similares a las diferentes características hace que las señales sean similares. Por ejemplo, A423B9 es más similar a A423B8 que 92 a 93. Se deben eliminar funciones innecesariamente similares y se deben resaltar características diferentes.

3.5.2. Principios del modelo mental

Un modelo mental es un mecanismo del pensamiento mediante el cual un ser humano, u otro animal, intenta explicar cómo funciona el mundo real. Es un tipo de símbolo interno o representación de la realidad externa, hipotética, que juega un papel importante en la cognición, estos se aplican al diseño para lograr una mejor interacción[27]

- Principio del realismo icónico. Una pantalla debe verse como la variable que representa (por ejemplo, temperatura alta en un termómetro que se muestra como un nivel vertical más alto). Si hay múltiples elementos, se pueden configurar de una manera que se vería en el entorno representado.

- Principio de la parte móvil. Los elementos en movimiento deberían moverse en un patrón y dirección compatibles con el modelo mental del usuario de cómo se mueve realmente en el sistema. Por ejemplo, el elemento móvil en un altímetro debería moverse hacia arriba al aumentar la altitud.

3.5.3. Principios basados en la atención

Atención es el proceso de tipo de conducta cognitivo de concentración selectiva en un aspecto discreto de la información, ya sea considerada subjetiva u objetiva, mientras que se ignoran otros aspectos perceptibles. La atención también ha sido denominada como la asignación de recursos de procesamiento limitados, todo esto para lograr un mejor y optimo conocimiento.[27]

- Minimizar el costo de acceso a la información o el costo de la interacción. Cuando la atención del usuario se desvía de un lugar a otro para acceder a la información necesaria, existe un costo asociado en tiempo o esfuerzo. Un diseño de pantalla debe minimizar este costo al permitir que las fuentes a las que se accede con frecuencia se ubiquen en la posición más cercana posible. Sin embargo, no se debe sacrificar una legibilidad adecuada para reducir este costo.

- Principio de compatibilidad de proximidad. La atención dividida entre dos fuentes de información puede ser necesaria para completar una tarea. Estas fuentes deben estar mentalmente integradas y se definen por su estrecha proximidad mental. Los costos de acceso a la información deben ser bajos, lo que se puede lograr de muchas maneras (por ejemplo, proximidad, vinculación por colores comunes, patrones, formas, etc.). Sin embargo, la proximidad de la pantalla cercana puede ser perjudicial al causar demasiado desorden.

- Principio de recursos múltiples. Un usuario puede procesar información más fácilmente a través de diferentes recursos. Por ejemplo, la información visual y auditiva puede presentarse simultáneamente en lugar de presentar toda la información visual o auditiva.

3.5.4. Principios del avance

Esta sección busca dar un mayor alcance, en busca de darle mayor visión al futuro inmediato de las diversas interacciones de la persona. [27]

- Reemplaza la memoria con información visual: conocimiento en el mundo. Un usuario no debería necesitar retener información importante únicamente en la memoria de trabajo o recuperarla de la memoria a largo plazo. Un menú, lista de verificación u otra pantalla puede ayudar al usuario al facilitar el uso de su memoria. Sin embargo, el uso de la memoria a veces puede beneficiar al usuario al eliminar la necesidad de hacer referencia a algún tipo de conocimiento en el mundo (por ejemplo, un operador de computadora experto preferiría usar comandos directos de la memoria que consultar un manual). El uso del conocimiento en la cabeza de un usuario y el conocimiento en el mundo deben ser equilibrados para un diseño efectivo.

- Principio de ayuda predictiva. Las acciones pro activas suelen ser más efectivas que las acciones reactivas. Una pantalla debe intentar eliminar las tareas cognitivas que exigen recursos y reemplazarlas con tareas perceptivas más simples para reducir el uso de los recursos mentales del usuario. Esto permitirá al usuario enfocarse en las condiciones actuales y considerar posibles condiciones futuras. Un ejemplo de ayuda predictiva es una señal de carretera que muestra la distancia a un determinado destino.

- Principio de consistencia. Los viejos hábitos de otras pantallas se transferirán fácilmente para admitir el procesamiento de nuevas pantallas si están diseñadas de manera consistente. La memoria a largo plazo de un usuario desencadenará acciones que se espera que sean apropiadas. Un diseño debe aceptar este hecho y utilizar consistencia entre diferentes pantallas[27].

3.6. Conclusiones y Discusión.

La Interacción Humano-Computador es una disciplina relativamente reciente dentro de la Informática, que ha ido tomando cuerpo fundamentalmente, a la necesidad de facilitar el uso de los computadores a una población más amplia -y de este modo alcanzar un mayor mercado y, por otro lado, a la necesidad de técnicas de interacción sofisticadas para determinados tipos de aplicaciones.

La revisión de los métodos y técnicas de HCI presentada nos lleva a la necesidad de aplicar conceptos, métodos y técnicas procedentes de otros campos. Uno de ellos (quizá) el que más influencia esté teniendo en el avance hacia interfaces más sofisticadas es la Inteligencia Artificial. Si tuviéramos que resumir en una frase la tendencia en lo relativo a las interfaces inteligentes, diríamos que el objetivo general es conseguir interfaces aparentemente inteligentes y fácilmente confundibles con humanidad.

La creación y desarrollo de nuevas tecnologías no solo debe enfocarse en hacer creer al interactuante que en la maquina hay una persona, esta interacción debe aportar a la persona y debe ser generadora de estímulos y refuerzos, al conocimiento.

Adicionalmente y como comentario personal, debemos agregar que también es importante el uso de la interfaz gráfica de cada una de estas herramientas software para el mejor uso e interacción de la HCI, teniendo en cuenta tanto el tipo de computadora (que podemos definir dentro de un conjunto de dispositivos que van desde un celular, pasando por una tablet, luego un portátil y por último un computador de escritorio) hasta el tipo de usuario final que va a utilizar la herramienta (personas del común, comerciantes, profesionales, estudiantes, etc...), ya que para cada una de estas combinaciones de humano - computadora la interacción debe ser diferente y ajustada al punto de mayor interactividad.

Capítulo 4

Aprendizaje en la programación

En este capítulo haremos evidente los métodos de aprendizaje (desprendido del concepto biológico y apegado al concepto académico) estudiados en el marco de contexto de los lenguajes de programación y el aprendizaje de los mismos, estudiando diferentes metodologías en su mayoría de tipo experimental, expuestas por diferentes eminencias en el área de la programación así como ejemplares académicos al rededor del mundo. El objetivo de este capítulo es conseguir definir y comprender los diferentes métodos de aprendizaje utilizados por los autores expuestos para poder recopilar y sintetizar junto con los demás estudios de este documento, herramientas y metodologías que puedan unirse para ser aplicadas en la plataforma ipp5, tanto en el trabajo actual como en el trabajo futuro.

4.1. Aprendizaje extremo

Este método se ha implementado en varios casos de estudio que han arrojado diferentes resultados, está el caso de la universidad Helsinki[2] donde se reestructuraron las clases de programación básicas reduciendo las horas implementadas al exponer las temáticas pertinentes y la forma de evaluar a los estudiantes; el segundo caso investigado fue aplicado a un marco de tutorías de la Universidad Indiana Bloomington[22]. Este método se apoya en un modelo cognoscitivo del aprendizaje que ha tenido recientemente muchas aplicaciones en la enseñanza de la programación, este modelo se basa en la forma tradicional como se ha venido aprendiendo durante la historia de la humanidad que consiste en *aprender mientras se trabaja bajo la guía de un maestro de alto nivel*[2] con la diferencia de que este método se centra en el aprender habilidades cognitivas mediante la constante evaluación del proceso en vez de evaluar el producto final.

El método de aprendizaje extremo consiste en aplicar el ciclo de vida de la metodología de desarrollo *Xtreme Programming* enfocado en Scaffolding que se centra en realizar ejercicios con puntos de partida claros y en parte rutinarios con el fin de que el estudiante piense en solucionar problemas con los conocimientos que cuenta siendo capaz de adaptarlos al problema propuesto y esto se realiza en un estimado semanal que va aumentando de dificultad con la condición de que los ejercicios de mayor dificultad requieran a los de menor dificultad para que el estudiante se adapte y aprenda a dividir los problemas de mayor dificultad en sub tareas más sencillas de realizar. Teniendo en cuenta que estos ejercicios deben animar al estudiante a buscar información sobre elementos que no se explicaron en la exposición de los temas y que presenten metas cortas en intervalos de los mismos para generar motivación porque al completar la meta intermediaria los estudiantes se sentirán más cómodos y seguros de realizar el ejercicio por completo, evitando así que desistan. Por otro lado están las tutorías que hacen parte del proceso de andamiaje del aprendizaje extremo que consiste en ayudar continuamente a los estudiantes en los laboratorios, guiándolos y haciendo aportes constructivos sobre su trabajo mediante un tutor en este caso el profesor que evalúe su proceso cada 15 días y que publique aproximadamente entre 15 a 30 ejercicios semanales, explique lo requerido por los ejercicios propuestos durante la semana y que publique el historial virtual sobre las clases expuestas como consulta para los mismos estudiantes, centrándose principalmente en empezar la labor práctica en los laboratorios y utilizar el aprendizaje extremo lo antes posible. Esta metodología extrema aumento así el porcentaje de 58.5 % a 70.1 % de estudiantes que aprobaron la materia, se ha concluido que esta metodología tiende a mejorar los resultados ya que se promueven tareas repetitivas construyendo rutinas que permite reutilizar conocimientos adquiridos, lo que hace mas fácil aprender programación avanzada.

4.2. Aprendizaje basado en concursos

En esta sección vamos a hablar sobre un método de enseñanza que se ha promovido desde los inicios de los lenguajes de programación y que ha integrado nuevos avances y mejoras en cuanto a metodología como solución para transmitir conocimientos sobre lenguajes de programación, el aprendizaje de programación basado en concursos. En esta sección revisaremos las ventajas que este posee definiendo la forma didáctica en la cual los aprendices compiten para poder lograr un objetivo y de esta manera afianzar sus conocimientos así como practicar y aprender nuevos paradigmas. También se mostrará una breve introducción cronológica de esta metodología así como

sus avances y fallos a través del tiempo para obtener un punto de vista sobre lo que se ha convertido.

El método de aprendizaje mediante concursos se lleva implementando hace un buen tiempo, empezó con el ACM International Collegiate Programming Contest patrocinado por IBM en 1977, en donde anualmente los estudiantes universitarios tenían la posibilidad de competir por equipos con participantes internacionales. A partir de aquí han surgido numerosos concursos nacionales e internacionales al rededor del mundo (Cada uno con sus reglas) que, con el paso del tiempo se han refinado con reglas independientes que los hacen únicos y que han convertido a los concursos en metodologías de práctica y enseñanza de programación y lenguajes de programación en general.

Para determinar el porqué de los concursos de programación debemos entender la psicología detrás de estos, la meta - información generada al aplicar estos concursos en grupos de estudiantes o participantes, de aquí es destacable el hecho de que estos promueven una competitividad sana entre los participantes la cual genera un mayor interés aplicable para los participantes que el hecho de tener que aprender teoría sin ningún motivo, este caso se puede ver reflejado en los concursos en los cuales se debe utilizar un paradigma de programación determinado para resolver los problemas, en el cual los estudiantes se harán a la tarea de aprender la teoría necesaria para aplicarla en un caso real y resolver la tarea (Objetivo del problema propuesto) para así sumar puntos. Al momento de identificar esta gran ventaja sobre los métodos tradicionales de enseñanza se empezó a aplicar en universidades y colegios donde tomó gran fuerza, se cambiaron los talleres por pequeños concursos grupales y se cambiaron los parciales por retos de programación, el inconveniente es que con esta solución llegó un problema. Al estudiar el comportamiento de los estudiantes a través del tiempo con respecto al nuevo método de enseñanza se encontró que el desempeño de los estudiantes buenos se iba resaltando a través del tiempo, todo bien en este caso, pero en cambio el desempeño de los estudiantes de bajo rendimiento tendía a empeorar y se incentivaba el desuso del concurso por miedo al fracaso generando estudiantes insatisfechos y que cada vez tenían un peor rendimiento. A partir de este hecho se decidió investigar sobre los concursos de talla internacional y ver si este caso se replicaba también en ellos, pero la respuesta fue no, en cambio los participantes que habían perdido en años anteriores mejoraban y llegaban a puestos mucho más altos a medida que pasaba el tiempo, la pregunta es ¿porqué? y la respuesta está en las entrañas de las reglas de los concursos.

En el escrito de Sébastien COMBÉFIS y Jérémy WAUTELET, se definen para los concursos dos tipos de criterios de clasificación, el primero es el conjunto de criterios relacionado con la información general del concurso, y

el segundo es el conjunto de criterios relacionado con las tareas (problemas) a realizar por parte de los estudiantes.[8]

El primer conjunto (información general del concurso) define los siguientes seis conceptos:

1. Se refiere a si el tipo de concurso es individual o en grupos.
2. Se refiere a las condiciones que los participantes deben satisfacer relacionado en cuanto a edad, género o cualquier restricción relacionada con sus estudios.
3. Se refiere a los lenguajes de programación permitido.
4. Se refiere a la duración del concurso o el intervalo de tiempo permitido que los participantes tienen para desarrollar sus tareas.
5. Se refiere a la frecuencia de los concursos, por ejemplo algunos concursos o retos comienzan tan pronto se decide empezarlo.
6. Se refiere al cómo serán publicados y calculados los puntajes de los participantes.

Una vez entendidos los conceptos base para la clasificación de los concursos de programación debemos entender donde se establece la falla principal, puesto que falta algo muy importante, la retro alimentación. Es tan importante que define el éxito de un concurso de programación puesto que para poder resolver el problema mencionado anteriormente se requiere de una metodología que promueva el uso de los concursos y no que lo desprestigie y es la respuesta para que los concursantes se motiven cada vez más y mejoren en cada concurso, de hecho la retro alimentación es la base para poder enseñar lenguajes de programación sin fallar en el intento y se debe implementar para todos y cada uno de los participantes del concurso generando comentarios no solamente de el porqué de la solución ideal sino que también les muestre recomendaciones de como haber mejorado en la solución entregada del problema.

4.3. Aprendizaje basado en juegos en línea

En la educación secundaria así como en la educación superior se debe buscar que el estudiante se mantenga interesado y entusiasmado en el aprendizaje, en el caso de la educación de programación esta genera beneficios al

estudiante, "*la educación de programación permite el desarrollo de capacidades que contribuyen a mejorar el pensamiento lógico de estudiantes, y el segundo es que tal proximidad puede ayudar establecer la visión y las actitudes de los estudiantes con respecto al uso de las tecnologías que generalmente manejan*"[21], mas sin embargo en esta época los métodos clásicos para mantener la atención de los estudiantes han quedado obsoletos. Se realizó un caso de estudio en donde se analizaron estudiantes de educación secundaria aplicando diferentes metodologías de enseñanza presencial y a distancia, para este caso, se usaron primero para dar a conocer el temario de vídeo conferencias, las cuales alientan al estudiante a pensar de manera independiente, a desarrollar un énfasis práctico y alienta al estudiante a generar la capacidad de poder repetir una o varias veces las lecciones.

Para el caso de estudio se realizaron una serie de entrenamientos y desafíos en donde luego de cada sesión de clase los estudiantes realizaban ejercicios de práctica, posterior a esto competían entre ellos mediante retos ya previamente creados o creaban nuevos, esto lo podían compartir mediante sus redes sociales, como Facebook, Twitter ó Google+. Esto los incentivaba a iniciar lecciones cada vez mas rápido, como objetivo se tenia el fomentar la creatividad y las habilidades relacionadas con el diseño y desarrollo, fomentar la programación en la escuela secundaria, trabajar el despertar vocacional, atraer nuevos talentos al área de computación, entrenar y desarrollar las habilidades de los estudiantes en desarrollo de juegos digitales a través del uso de herramientas de juego (motores) como Construct2, promover un aprendizaje en línea ambiente para permitir a los estudiantes ingresar a el desarrollo de juegos digitales en el área académica e incentivar a participar estudiantes y profesores en actividades que estimulan el conocimiento sobre el pensamiento computacional. [21]

Los resultados del estudio indicaron que los estudiantes que no tenían ningún tipo de conocimiento previo en programación fueron más reacios a participar en las actividades, también se evidenciaron claras tendencias de deserción en comparación a los estudiantes que ya tenían conocimientos previos; así mismo la mayoría de los estudiantes con conocimientos previos continuaron y finalizaron con éxito las actividades, y continuaron retándose entre ellos para superar sus propias puntuaciones y las de los demás; también se encontró que los vídeos de referencia y tutoriales de menos de 8 minutos eran visualizados en su totalidad mientras que los de duración entre 8 a 15 minutos no se completaban de visualizar en su totalidad. Esto nos da una aproximación de qué tantos vídeos, duración en tiempo y uso de juegos en línea se deben usar en la enseñanza de la programación.

Dentro de los factores que afectaron el desarrollo del caso de estudio se encontró uno de los principales obstáculos que radica en la interacción humano

- máquina, puesto que los estudiantes requerían de un estímulo constante que simulara el acompañamiento constante de una persona puesto que la interacción entre humanos es parte esencial de la cotidianidad, por tal razón, al no poder suplir esta interacción se genera cierto rechazo y falta de continuidad dentro del proceso de aprendizaje. [7]

4.4. Aprendizaje mediante robots personales

El uso de robots para la realización, automatización y ayuda en las tareas humanas ya es un echo y esto se puede aprovechar para generar una motivación en ciertas áreas del aprendizaje que tengan que ver con el desarrollo y uso de las tecnologías. En estos casos se han utilizado robots físicos y virtuales tales como iRobot Create, Lego Mindstorms, King, IPRE entre otros ya que cualquiera de estos le brinda a los programadores novatos la motivación para aprender lenguajes de programación.

En uno de los casos de estudio en el que se implementaron entornos de aprendizaje usando dispositivos programables como los robots personales desarrollados por la IPRE para la introducción a la programación con Python y el paquete Myro, se identificó el impacto que generaba usar estos robots en el aula de clase en la motivación de diferentes grupos de estudiantes con diferentes características teniendo en cuenta que los objetivos del curso introductorio a la programación eran:*"proporcionar una experiencia de inmersión en la programación introductoria que incluyen elementos de diseño, introducción el proceso de desarrollo utilizado en la creación de proyectos de medios interactivos y promover una comprensión de los roles entrelazados de diseño y desarrollo."*[13]. La metodología de trabajo que se realizaba para cumplir con los tres objetivos del curso se presentaban mediante un conferencia de 1 hora y un laboratorio de 2 horas durante 15 semanas en donde un instructor utilizaba los laboratorios para responder inquietudes sobre los robots y el trabajo realizado, pruebas, exámenes y proyectos de los cuales el primero se realizaba a la mitad del curso implementando funciones básicas de programación (bucles, funciones aleatorias, variables y expresiones) e incorporando el diseño en la creación de una historia interactiva para la parte creativa y de diseño; el segundo proyecto ya más complejo, se tenia que incorporar aparte de lo que ya se había incorporado en el primer proyecto mediante el uso de una interfaz gráfica, sensores adicionales, objetos, sonidos, música y utilizando a más de un robot para la simulación de un entorno interactivo.

Los resultados estadísticos del estudio se realizaron mediante encuestas para medir la diferencia de motivación y la concepción del aprendizaje mediante robots entre el pre y el post de haber realizador el curso con éxito,

analizando los resultados el *.estudio de investigación indica que el uso de robots IPRE en un curso de introducción puede motivar a los estudiantes de otras carreras para aprender programación, específicamente por que llaman su atención."*[13] sin embargo, este resultado solo se afianza con los estudiantes que al finalizar el curso les termina interesando el desarrollo como tal, al resto de estudiantes simplemente les fue indiferente. Por otro lado pero no menos importante es que la motivación de algunos estudiantes se veía afectada porque las tecnologías que se usaban en el aula tenían problemas de compatibilidad en cuanto al hardware y el sistema operativo de los dispositivos, lo cual permitió generar una retroalimentación constructiva para el mejoramiento de los robots y su uso posterior.

Para concluir se puede intuir que los robots no son una motivación muy atractiva para los estudiantes ya que su uso en el aprendizaje es reciente y genera complicaciones como la portabilidad, compatibilidad y costos; por esto se recomienda que se genere motivación en el aprendizaje de los diferentes lenguajes de programación mediante otras herramientas más cotidianas y dispositivos de fácil acceso como lo son los celulares, computadores, televisores, y demás elementos electrónicos portátiles ya que son herramientas más pertinentes para los jóvenes de esta generación.[15]

4.5. Conclusiones y Discusión

La enseñanza, no solo de la programación tiene muchas variaciones, en este capitulo se busca analizar diferentes formas algunas experimentales de enseñanza, buscando las ventajas y desventajas de cada una de ellas, todo esto conlleva a usar partes de estas teorías, para la aplicación e implementación de la plataforma ipp5, se plantea el uso y la aplicación en la plataforma de los concursos en la enseñanza, todo esto para lograr impactar en la parte motivacional de los estudiantes, así mismo se busca dar una guía y estimulo mediante el uso de robots personales que estimulen la interacción humano computadora, de igual manera se plantea en el capitulo de trabajo futuro los potenciales usos sobre la investigación de las practicas de la enseñanza en la programación.

Capítulo 5

Estructura Del Aprendizaje Aplicada a los módulos de IPP5

En este capítulo se pretende mostrar de manera explícita y sintetizada la forma del contenido programático que deberán tener los módulos iniciales de la plataforma ipp5 así como una breve explicación de su razón de ser alineados a los estudios mostrados en este documento. Adicionalmente este capítulo servirá como bitácora a futuros desarrollos para su estudio y mejoramiento como plataforma y como herramienta de enseñanza.

5.1. Aplicación de metodologías de enseñanza e Interacción Humano Computadora.

Para la aplicación de las metodologías para la enseñanza estudiadas se decidió seleccionar las siguientes: -Estímulos y su respuesta. -Refuerzos positivos. -Karl Popper:Demarcación

Así como su unión con la interacción humano computadora, se plantea el uso de bots para dar respuesta a diversas interacciones.

5.2. Módulos de enseñanza

5.2.1. JavaScript

El módulo de JavaScript tiene como razón de ser mostrar y explicar el contenido de un lenguaje de programación de manera interactiva. A lo largo del documento, hemos visto la importancia de la interacción humano - computadora para poder transmitir de manera eficaz cualquier tipo de información

generando estímulos e interfaces que le permiten al usuario familiarizarse con algo que ya conocen, precisamente esa es la razón de ser de este módulo, ya que gracias a la versatilidad de JavaScript podemos generar elementos gráficos e interactuantes que le permiten al usuario adquirir no solamente un conocimiento teórico sino práctico, sensorial y memorial gracias al uso de librerías gráficas que nos permiten usar el canvas del navegador para mostrar objetos gráficos complejos.

Primeros Pasos

El módulo primeros pasos (Ver figura: 5.1) contiene la información inicial con la cual los usuarios se empaparán del lenguaje de programación JavaScript, en este se pretende que mediante la ayuda de un tutor con experiencia el usuario pueda afianzar los conocimientos básico e iniciales sobre los lenguajes de programación (independiente del lenguaje) dando ejemplos de datos generales que ayudan a entender qué es un lenguaje de programación, cómo funciona y para qué sirve. Luego, naturalmente se muestran conceptos específicos del lenguaje JavaScript que van a ser el punto de entrada para que las personas interactúen y realicen un reto al final. La idea en este y cada uno de los módulos es que haya una retroalimentación para todos los usuarios que les permita despejar dudas y así crear unas bases sólidas que les permitan entender los tópicos posteriores.

Figura 5.1: Página inicial Primeros Pasos.

Memoria

En el módulo de Memoria (Ver figura: 5.2) se presenta una pequeña parte teórica que pretende afianzar desde el punto de vista general de lenguajes de programación, el uso de las variables desde su definición en un contexto global, como local dentro de un bloque o texto de código. En este se presentan los datos de una manera clara y se espera que la ayuda del tutor no sea necesaria dado su bajo nivel de dificultad, la idea es que el usuario pueda afianzar estos conocimientos por su cuenta pero, de ser necesaria la ayuda del tutor esta se debe presentar. Adicional a esto, el módulo de Memoria sigue los lineamientos de interacción humano - computadora utilizando un botón interactuante que servirá como guía para ayudar a afianzar el tema tratado en la teoría, así mismo se presenta un reto sencillo que acumula información del módulo anterior (Primeros Pasos) para realizar una continuidad en el proceso de aprendizaje.

En este ejemplo se muestra el código base para el reto del módulo de Memoria, en este se pretende que los usuarios, dado el código, puedan hacer que la pelota se mueva a una velocidad diferente.

```
var pos = 300;

function setup() {
  createCanvas(600, 400);
}

function draw() {
  background(220, 180, 200);
  pos = pos + 1;
  ellipse(pos, pos, 100, 100);
  ellipse(pos-200, pos-200 , 50, 50);
}
```

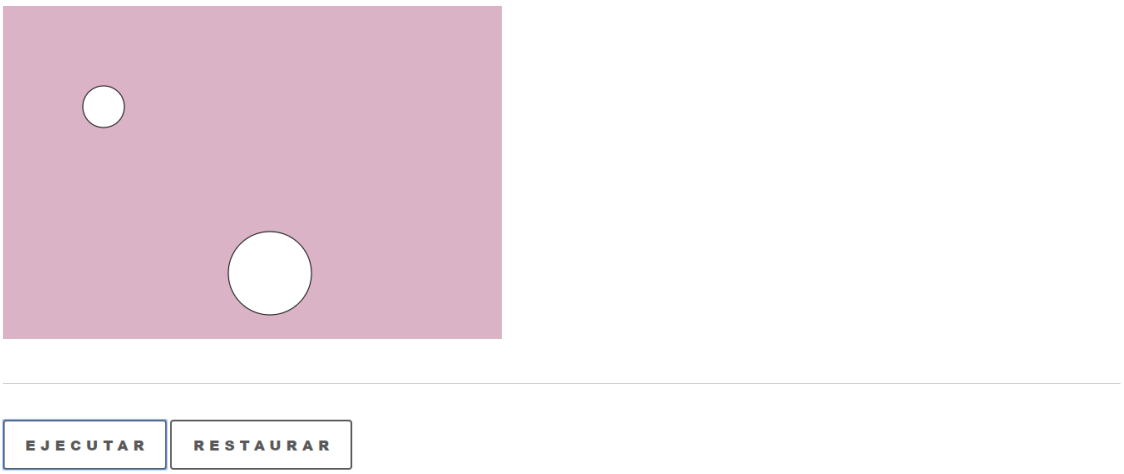

Figura 5.2: Reto del módulo Memoria.

Eventos y condicionales

El módulo de Eventos y condicionales (Ver figura: 5.3) es uno de los módulo más complejos, en este módulo se sintetizan dos temáticas que esta fuertemente relacionadas y que al igual que en los módulos anteriores consiste de una parte teórica, un reto acumulativo y una parte de interacción sobre un botón interactuante que permite visualizar de manera interactiva estas temáticas, se espera que la ayuda del tutor sea presente en este módulo ya que es uno de los más complicados para la mayoría. Se pretende que el tutor, luego de que los usuarios estudien por su cuenta, explique en qué consisten los condicionales y, con base a esto, explicar como generar eventos que se disparen dada una situación específica. Para esto se ha desarrollado

un reto que va de acuerdo a lo que se quiere explicar donde se combinan los condicionales y eventos para manejar una bola de billar sobre el tablero, en este caso se recomienda mostrar a los usuarios, con base a lo visto anteriormente la manera lógica en la que funcionan las computadoras. Un caso que va a ocurrir con gran frecuencia es cuando se implemente una velocidad de la bola de billar y para algunos casos cumpla, pero para otros no. En este caso, tener en cuenta como funcionan las computadoras y sus estados en memoria (Tener en cuenta la diferencia entre menor que <, mayor que >, menor igual <= y mayor igual >=).

En este ejemplo se muestra el código base para el reto del módulo de Eventos y condicionales, en este se pretende que los usuarios, dado el código, puedan hacer que la pelota rebote y se mueva a una velocidad diferente.

```
var posX = 10;
var posY = 10;
var velocidadX = 4;
var velocidadY = 3;

function setup() {
  createCanvas(600, 400);
}

function draw() {
  background(0, 180, 0);
  ellipse(posX, posY, 20, 20);
  posX = posX + velocidadX;
  posY = posY + velocidadY;
}
```

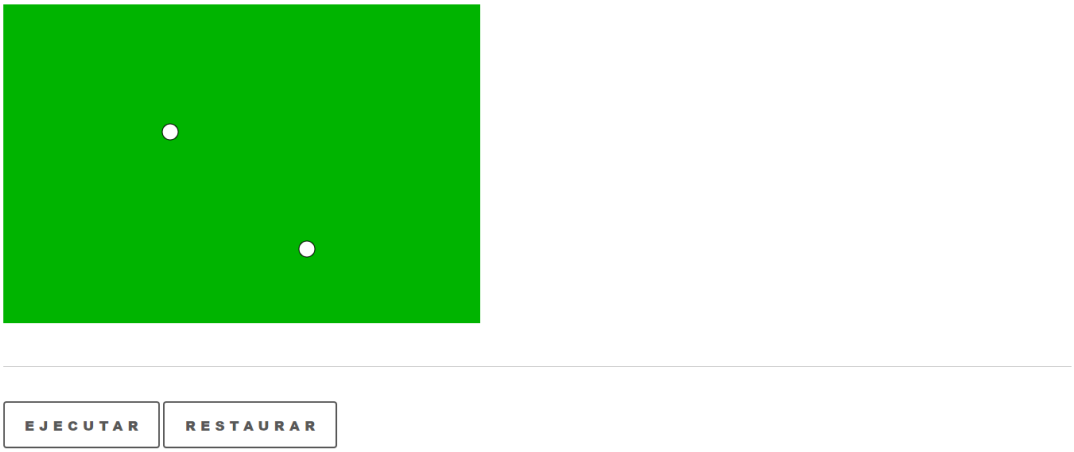

Figura 5.3: Reto del módulo Eventos y condicionales.

Ciclos

El módulo de Ciclos (Ver figura: 5.4) contiene una estructura similar al de los demás componentes, la única diferencia, es que en este módulo se muestra información de los dos tipos de ciclos más populares, el ciclo FOR, y el ciclo WHILE. Ambos ciclos son expuestos de manera separada en sub secciones, la intención de este módulo es presentar a los usuarios por un lado, uno de los ciclos más comunes y utilizados por excelencia para hacer recorridos generalmente finitos (ciclo FOR) y por el otro un ciclo que utiliza condicionales booleanos de manera infinita hasta que se deje de cumplir su condición. A pesar de que la teoría está explicada con ejemplos y siguiendo los lineamientos de la interacción humano computadora con distintos ejemplos interactuantes se recomienda que los usuarios sean acompañados de un tutor que pueda resolver sus dudas, ya que en algunos casos se hará evidente que ambos pueden cumplir con las mismas tareas dadas las circustancias pertientes.

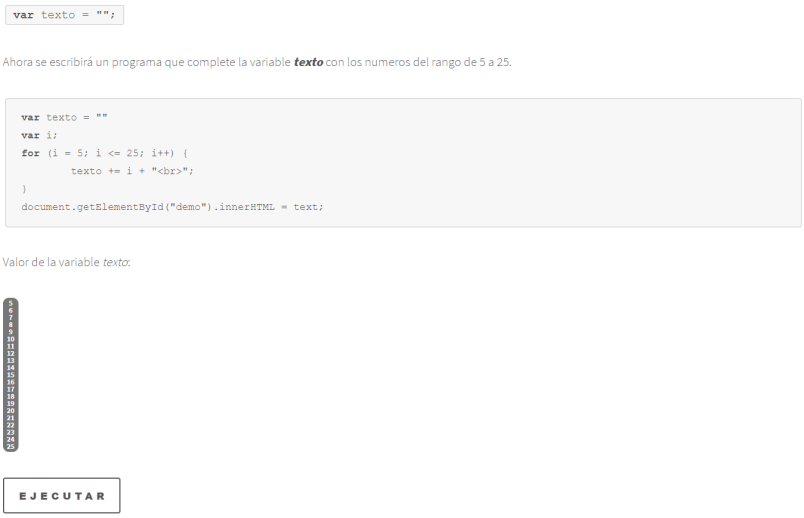

Figura 5.4: Teoría y ejemplo del módulo Ciclos.

Retos

El módulo de Retos(Ver figura: 5.5) es uno de los más importantes, no contiene algún tipo de teoría ni de interacción dirigida directamente a los usuarios, pero es fruto de las investigaciones realizadas ya que propone un espacio para que los docentes a cargo de la plataforma puedan realizar retos que les permitan a los estudiantes recapitular experiencias y afianzar las que ya han conseguido. El uso de este módulo debe ser expandido en el futuro para que los usuarios puedan también generar retos que les permitan competir contra sus compañeros en una sana competencia, siempre con la premisa de que cada reto debe tener una solución que sea explicada por medio de una entrada tipo blog para el proceso de retro alimentación. De igual manera, este módulo deberá ser usado para generar retos adicionales y de refuerzo a cada uno de los módulos existentes con el fin de dejar trabajo en casa autodidacta que le permita a los usuarios trabajar de manera autónoma y así llegar a reforzar sus conocimientos en clase.

En este ejemplo se muestra el código base para el reto 1 del módulo de Retos, en este se pretende que los usuarios, dado el código, puedan hacer que la pelota rebote contra la segunda pelota interactuante.

```
var tamanoBolas = 60;

var posX1 = 10;
var posY1 = 10;
```

```
var velocidadX1 = Math.floor((Math.random() * 10) + 1);
var velocidadY1 = Math.floor((Math.random() * 10) + 1);

var posX2 = 160;
var posY2 = 160;
var velocidadX2 = Math.floor((Math.random() * 10) + 1);
var velocidadY2 = Math.floor((Math.random() * 10) + 1);
function setup() {
  createCanvas(600, 400);
  }

function draw() {// Colorea la mesa
  background(0, 180, 0);

    posX1 = posX1 + velocidadX1;
    posY1 = posY1 + velocidadY1;
    posX2 = posX2 + velocidadX2;
    posY2 = posY2 + velocidadY2;
    if(posX1>=600){velocidadX1 = velocidadX1 * -1;}
    if(posX1<=0){velocidadX1 = velocidadX1 * -1;}
    if(posY1>=400){velocidadY1 = velocidadY1 * -1;}
    if(posY1<=0){velocidadY1 = velocidadY1 * -1;}

    if(posX2>=600){velocidadX2 = velocidadX2 * -1;}
    if(posX2<=0){velocidadX2 = velocidadX2 * -1;}
    if(posY2>=400){velocidadY2 = velocidadY2 * -1;}
    if(posY2<=0){velocidadY2 = velocidadY2 * -1;}

    fill(255,255,255);

    if((Math.abs(posX2 - posX1) <= tamanoBolas) && (Math.abs(posY2 - p
      fill(0,0,0);}
      //Pinta las bolas
    ellipse(posX1, posY1, tamanoBolas, tamanoBolas);
    ellipse(posX2, posY2, tamanoBolas, tamanoBolas);
    }
```

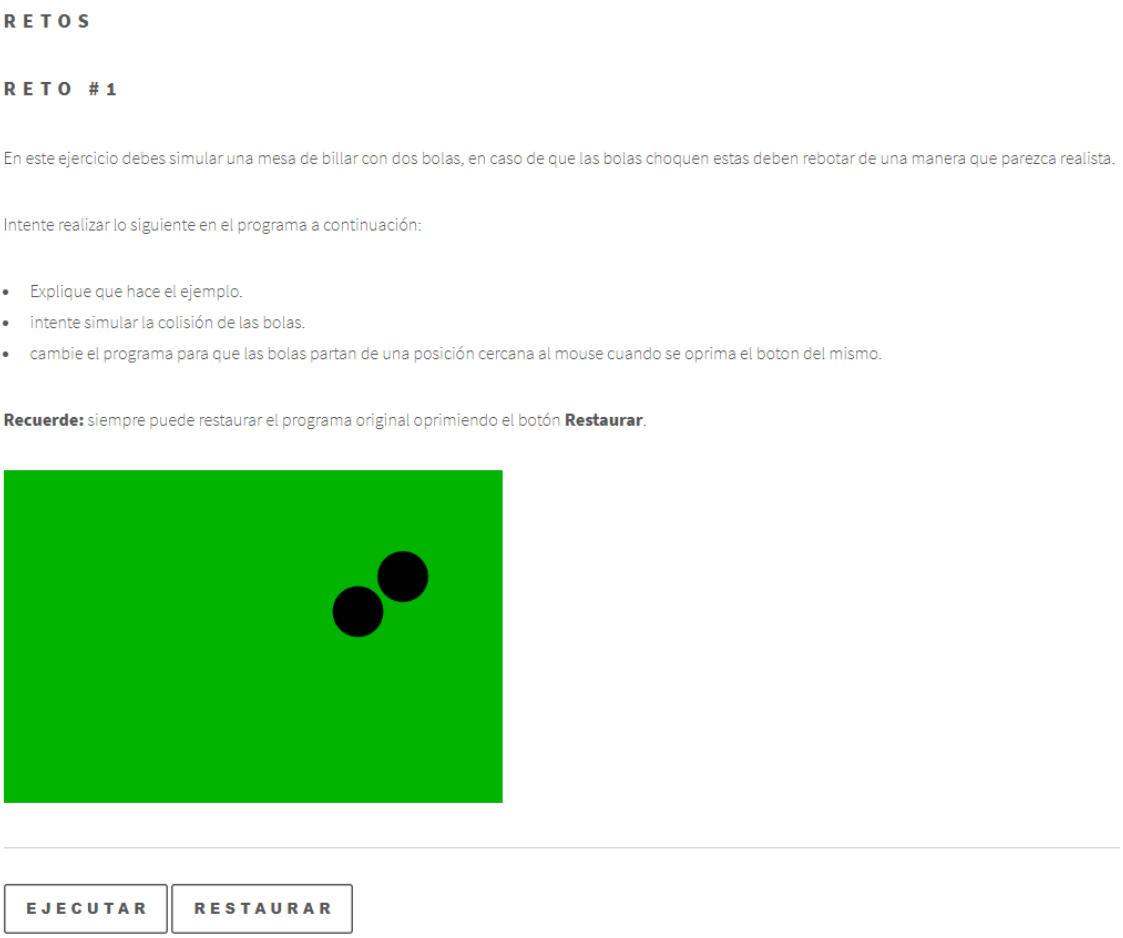

Figura 5.5: Reto #1 del módulo Retos. Dos esferas colisionando causando cambio de color.

5.2.2. Haskell

Primeros Pasos

El módulo primeros pasos (Ver figura: 5.6) contiene la información inicial con la cual los usuarios se empaparán del lenguaje de programación Haskell, en este se pretende que mediante la ayuda de un tutor con experiencia el usuario pueda afianzar los conocimientos básicos e iniciales sobre Haskell, estos incluyen operaciones simple como suma, resta, multiplicación y división así como operaciones de tipo booleanas, todo esto acompañado de ejemplos interactivos que ayudan a entender como funciona Haskell como lenguaje de programación, cómo funciona y para qué sirve. A diferencia de los módulos

de JavaScript, en este módulo no se presentan retos en específico, la idea de este módulo es introducir y presentar los conceptos básicos de programación funcional, en cambio en los módulos correspondientes a Haskell se integra un intérprete que tiene la capacidad de analizar y evaluar expresiones de Haskell así como guardar variables en memoria durante la sesión de los usuarios (Ver figura: 5.7) . Se recomienda un fuerte acompañamiento por parte de un tutor, no tanto por su nivel de complejidad sino para asegurar unas bases sólidas de aprendizaje para con los usuarios, de igual manera se espera que se deje un trabajo en casa o sesión de clase planteado con dificultad y extensibilidad apropiadas para que los usuarios se interesen cada vez más por los módulos de esta plataforma.

Figura 5.6: Explicación y lectura inicial de los primeros pasos de en Haskell.

Figura 5.7: Prueba en consola de primeros pasos en Haskell

Listas

Las listas son la estructura de datos más utilizada y pueden ser utilizadas de diferentes formas para modelar y resolver problemas, en esta sección se explica una base sobre listas, cadenas de texto que también son tratadas de la misma forma que las listas y operaciones sobre listas (Ver figura: 5.8), se explica desde la inicialización de listas, ya sean vacías o definidas, listas de listas, también se mostraran listas de manera mas compleja como lo son las listas por comprensión las cuales son definidas por ciertos argumentos para definir los elementos que pertenecen a ella, las listas por comprensión son muy similares a los conjuntos definidos de forma extensa, de igual manera se tratan las operaciones sobre las listas,como obtener su longitud, un elemento, generar su reverso, entre muchos otros, igual que en la sección de primeros pasos no se plantea un reto, mas si se plante el uso del interprete descrito anteriormente para servir de acompañamiento al tutor y guiar el trabajo de clase y de cada sesión por medio de la experimentación, así como el ensayo y error (Ver figura: 5.9).

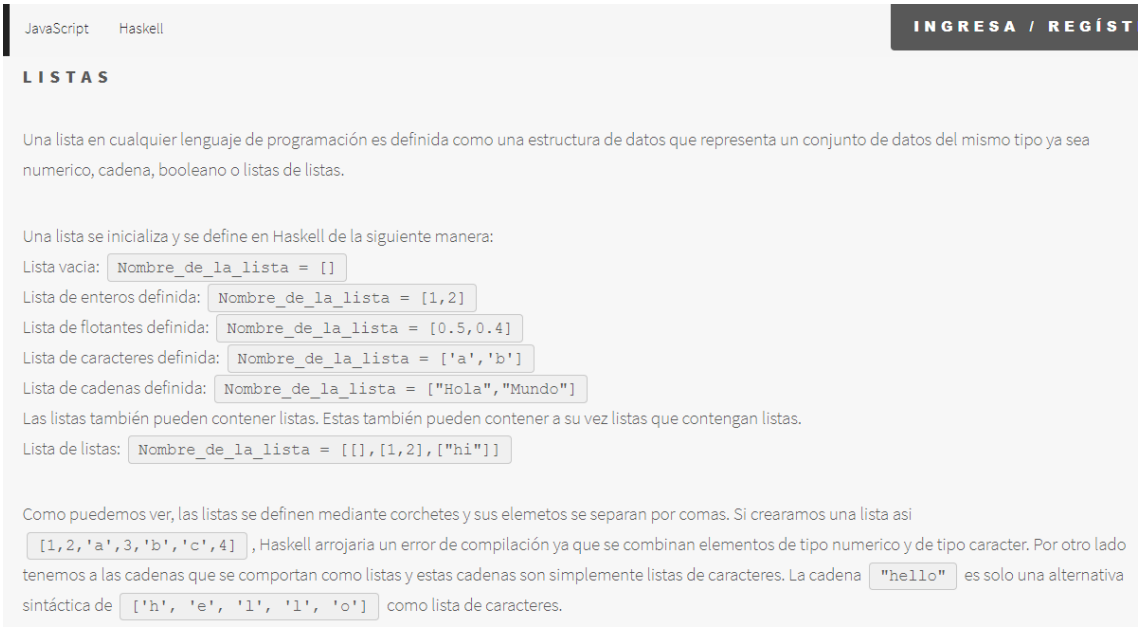

Figura 5.8: Explicación y lectura inicial de los primeros pasos para listas en Haskell.

Figura 5.9: Prueba en consola de listas en Haskell

Funciones

El principal propósito del módulo de Funciones (Ver figura: 5.10) es introducir a los usuarios a las funciones, se recomienda que el tutor a cargo empiece con las funciones más familiares para todos, las funciones matemáticas. La idea es que inicialmente se introduzcan estos conceptos a los usuarios sin necesidad de que tengan una aproximación al lenguaje como tal, ya que este lo permite. En el módulo se muestra una parte de teoría con ejemplos a ejemplos realizar con respecto a funciones matemáticas, la idea es que el tutor a cargo se encargue de mostrar otros tipos de funciones una vez entendidas as funciones matemáticas. El módulo inicialmente puede no ser difícil

si se introduce con cuidado, la idea es que estos conceptos queden lo más claros posibles para que en el futuro puedan ser usados por los usuarios de la manera correcta (Ver figura: 5.11).

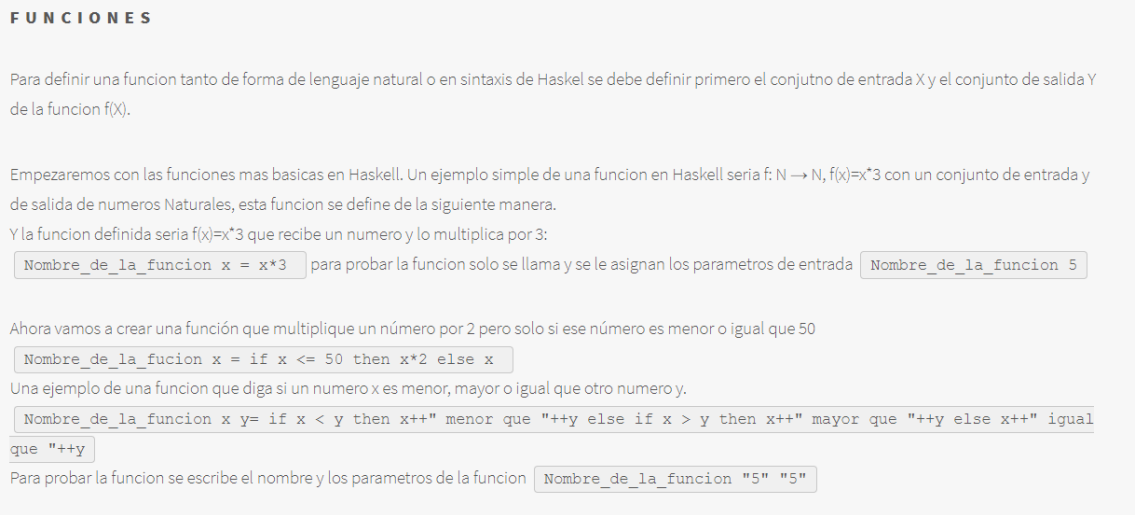

Figura 5.10: Explicación y lectura inicial de los primeros pasos para Funciones en Haskell.

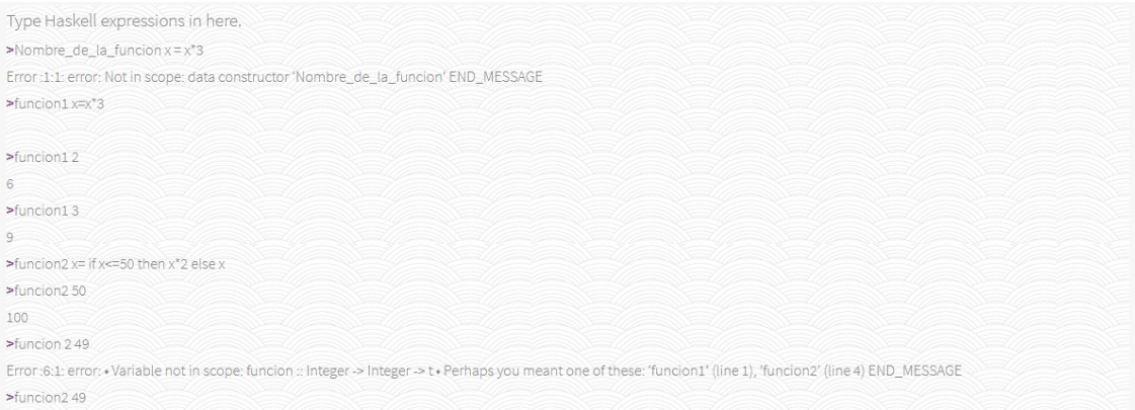

Figura 5.11: Prueba en consola de Funciones en Haskell

Recursión

El módulo de Recursión (Ver figura: 5.12) es uno de los más importantes, ya que su principal finalidad es introducir a los usuarios de manera natural al concepto de recursión desde el punto de vista lógico usando el potencial de Haskell para recibir estas expresiones de manera natural. Aquí es ideal que

el tutor se encargue de introducir la teoría y que el usuario la complemente con la teoría expuesta en la plataforma, este es uno de los módulos más complicados ya que introduce las teorías iniciales a la programación funcional. En este módulo se pretende dar un espacio de tiempo amplio para que los conceptos sean clarificados por parte del tutor y así lograr ser afianzados por los usuarios con sólidas bases para practicar y ejecutar ejercicios simples que le permitan al usuario afianzar conocimientos (Ver figura: 5.13).

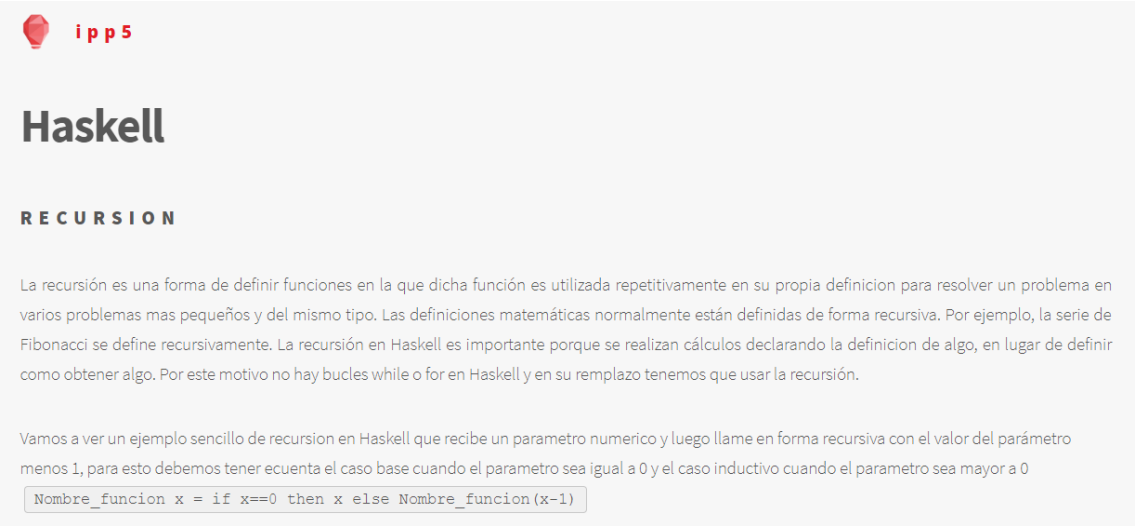

Figura 5.12: Explicación y lectura inicial de los primeros pasos para Recursion en Haskell.

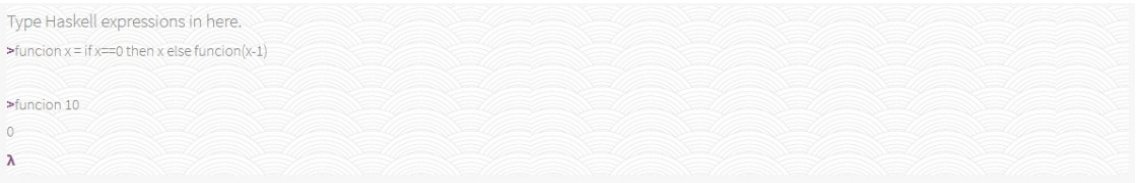

Figura 5.13: Prueba en consola de Recursion en Haskell

Capítulo 6

Arquitectura

En el desarrollo del proyecto de grado se realizaron estudios técnicos y tecnológicos para poder suplir las necesidades y requerimientos de este, dentro de las cuales se especificaba un sistema escalable en el tiempo, código de calidad documentado y tecnologías modernas. Esto acompañado de las necesidades tecnológicas de la implementación del Intérprete de Haskell en línea nos llevó a buscar las mejores soluciones disponibles actualmente a nivel de lenguajes de programación, así como adaptar las actuales herramientas tecnológicas de la nube para dicho fin para obtener los siguientes resultados.

6.1. Intérprete JavaScript

En el desarrollo de este proyecto, y para efectos de practicidad a la hora de enseñar el lenguaje JavaScript de manera interactiva se hace necesaria la creación de un Intérprete en línea embebido a la plataforma que nos permita hacer uso de todas sus capacidades no solo lógicas sino de generación visual. En este caso se hace uso de una librería externa denominada p5.js la cuál nos permite, mediante un editor y validador de JavaScript desarrollado en HTML, simular una consola de este para generar código compilable que se puede enviar para mostrar gráficos y así diseñar los módulos de aprendizaje y enseñanza de manera dinámica y gráfica para su mejor comprensión. Así mismo se hace uso del lenguaje sin ningún tipo de librería o intermediario para realizar las funciones sencillas como lo son las validaciones lógicas y los cálculos para así enriquecer la explicación de la teoría y generar también un ambiente adaptable (responsivo) en plataformas web y móviles permitiéndonos así, ofrecer las mismas capacidades de proceso tanto en computadores de escritorio como en celulares personales.

6.2. Intérprete de Haskell

Uno de los grandes retos en el proyecto fue la ideación e implementación de los simuladores de Intérprete en línea para ejecutar y evaluar código de Hhaskell en la plataforma mientras se realizan estudios teóricos, dado que la documentación sobre este tipo de aplicativos es limitada se llegó a unas vagas interpretaciones y aproximaciones de las cuales salieron 3 prototipos, todos ellos implementados y testeados en ambientes de desarrollo y con sujetos de prueba amenos al proyecto.

6.2.1. Intérprete mediante conexión SSH

El primer prototipo es un Intérprete completo que hace el uso de una máquina remota sobre conexión ssh para poder ejecutar el Intérprete de Haskell (stack ghci) en su sesión, en este el usuario se conecta a la terminal del equipo remoto utilizando las credenciales de usuario y contraseña para poder acceder a la terminal de la máquina anfitrión y así poder ejecutar los comandos respectivos de Haskell. En este caso la implementación se realizó utilizando NodeJS como server - side applicative el cual nos permite ejecutar proyectos sobre el servidor mediante NodeJS, este aplicativo contiene los elementos necesarios para comportarse como una terminal equivalente a la de máquinas Linux y es una modificación del proyecto con licencia pública WebSSH2 alojado en el sistema de repositorios github bajo el link: https://github.com/billchurch/WebSSH2. Este proyecto fue adaptado para poder ejecutarlo desde la plataforma ipp5 mediante un iframe que establece la conexión con la aplicación mediante los servidores de backend y así poder realizar el uso de la misma para permitir ejecutar código de Hhaskell en línea.

6.2.2. Intérprete mediante evaluador de expresiones de Haskell

El segundo prototipo se realiza bajo los lineamientos de su lenguaje, es decir utilizando Hhaskell como herramienta para poder analizar y ejecutar expresiones de haskell. En este caso se hace el uso del proyecto público de chrisdone ubicado en github: https://github.com/tryhaskell/tryhaskell el cual utiliza Haskell y el framework de aplicativos web para haskell snap para crear una página que simula una terminal que acepta expresiones Hhaskell para evaluarlas y retornar un resultado, en este caso la herramienta se ve limitada ya que utiliza un módulo de Haskell llamado mueval que solo permite ejecutar expresiones seguras, dando a entender como seguras aquellas expresiones que no ponen en riesgo la integridad de la máquina y por ende esta

no permite realizar expresiones complejas como funciones propias ni guardar variables en tiempo de ejecución para su posterior uso. En este caso, al igual que en el anterior prototipo, se implementó una modificación de dicho proyecto para adaptarlo y servirlo en la plataforma, en este se hace el uso de un contenedor que tiene instalado el Intérprete de haskell para ejecutar la aplicación y servirla para que sea consumida mediante un iframe embebido en cada parte práctica de las secciones del curso de haskell, en este caso la aplicación es servida por uno de los contenedores M2 de AWS(Amazon Web Services) y recibe las peticiones mediante javascript y Ajax, luego son enviadas directamente al aplicativo de haskell que se encarga de realizar la evaluación de la expresión y determinar si puede o no ejecutarla, de ser posible ejecutarla utiliza el módulo de haskell mueval para retornar el resultado, de no ser posible retorna un mensaje de error.

6.2.3. Intérprete mediante JavaScript y servidor REST java con Spring Boot

El tercer prototipo fue diseñado y desarrollado como un servidor web utilizando un estilo de arquitectura REST(Representational State Transfer) en lenguaje Java con el framework Spring Boot para facilitar la inyección de código usando Maven como herramienta para gestionar la automatización de la construcción del proyecto, en el servidor web se desarrollaron dos servicios(URLs) que al ser consumidos generan una respuesta. El servidor al recibir una solicitud GET de otra aplicación genera una salida dependiendo del comando Haskell digitado por el cliente, lo que hace el servidor al recibir tal comando es ejecutar un proceso en java que simula un comando de consola en el sistema operativo que ejecuta el Intérprete GHCi(Haskell) y a este proceso se le asigna un BufferedWrite para poder escribir los comandos recibidos por cliente en el proceso del compilador y un BufferedReader para leer la salida que genera el proceso del compilador al escribirle los comandos del cliente; esta salida es un String que contiene la salida generada por el compilador que se devuelve a la aplicación que hizo la petición GET. Esta otra aplicación en nuestro caso es el frontEnd que consiste en la parte visual del Intérprete donde el usuario ingresa los comandos Haskell; este se conecta y consume los servicios del servidor web que al recibir el String que contiene la salida del compilador que ejecuta un filtro y muestra la salida adecuadamente para el usuario con JavaScript. En este caso la aplicación es servida por uno de los contenedores M2 de AWSAmazonn Web Services) que es una plataforma de servicios de nube que ofrece potencia de cómputo y recibe las peticiones del cliente de JavaScript.

6.3. Estructura física

Para poder cumplir con los requerimientos técnicos y tecnológicos del proyecto optamos por distribuir la plataforma en servidores en la nube (específicamente AWS) que se configuraron para ser escalables con respecto al tráfico que manejen disponiéndose de balanceadores de carga que redirigen el trafico a los centros de cómputo más desocupados y creando nuevas instancias de ser necesarias. Así mismo se configuraron contenedores que alojan específicamente los contenidos estáticos de la plataforma (imágenes, html, estilos) para garantizar un flujo entre servidor y cliente óptimo y veloz, dejando de lado los procesadores lógicos en contenedores externos dedicados para este trabajo. Sobre lo anterior se presenta el siguiente esquema para su mejor entendimiento:

En el esquema se presenta la arquitectura sobre la cual está montada la plataforma ipp5, donde el usuario es redirigido por un DNS de alto rendimiento llamado Route53 que lo lleva a un servidor cloud front encargado de redirigir el tráfico de los usuarios hacia el balanceador de carga que solicita los recursos a los contenedores M3 lógicos y S3 de almacenamiento para entregar al cliente el acceso a la plataforma. Así mismo la plataforma se comunica con contenedores lógicos S3 para poder ejecutar los servicios de backend correspondientes a las implementaciones de los Intérprete de haskell.

6.3.1. Documentación y fuentes

Todo el código escrito para la realización de esta plataforma está dispuesto de manera pública en el repositorio en la nube github sobre el siguiente link: https://github.com/dnielben/ipp5

6.4. Diseño de pruebas

En este capitulo se recolecta la información correspondiente a las pruebas realizadas a la plataforma ipp5 para determinar su nivel de calidad y comportamiento en la simulación de un ambiente real.

6.4.1. Definición de las pruebas

Pruebas de Carga

Las pruebas de carga corresponden a un ambiente de pruebas configurado para determinar y validar la respuesta de una aplicación cuando es sometida

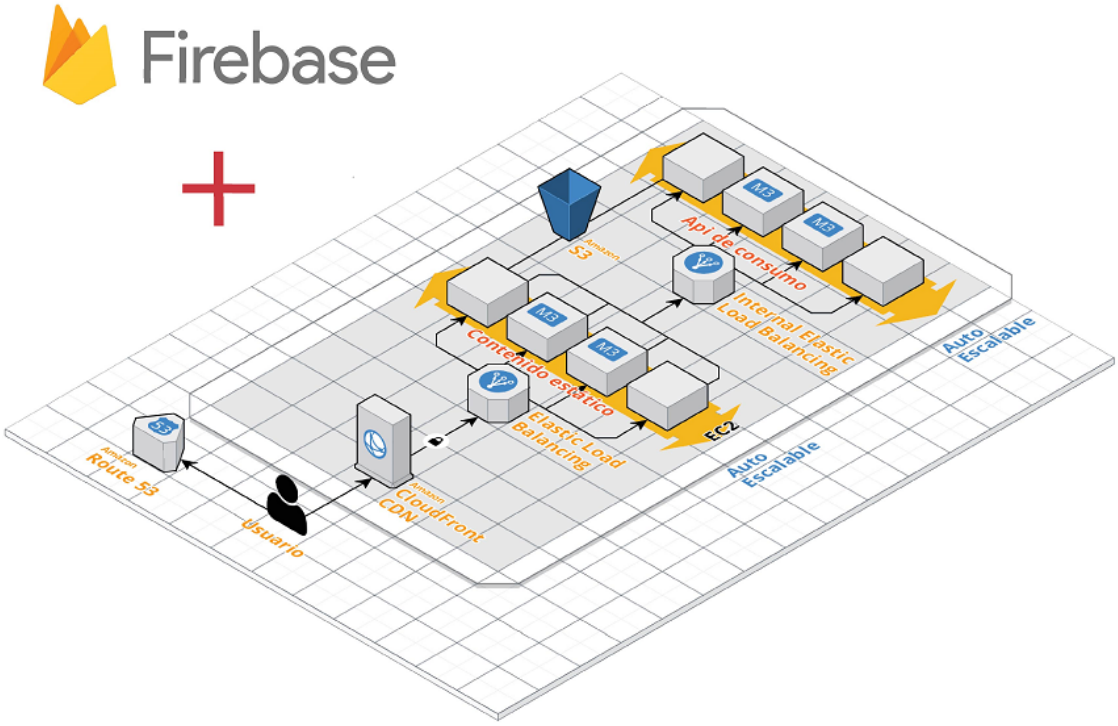

Figura 6.1: Arquitectura de la Plataforma

a una carga de usuarios y/o transacciones que se espera en el ambiente de producción. Por ejemplo: Verificar la correcta respuesta de la aplicación ante el alta de 100 en forma simultánea.

Pruebas de Stress

Las pruebas de stress corresponden a un ambiente de pruebas configurado para encontrar el volumen de datos o de tiempo en que la aplicación comienza a presentar fallas o se vuelve incapaz de responder a las peticiones. Son pruebas de stress ó de rendimiento que superan los límites esperados de los servidores de backend en un ambiente de producción. Por ejemplo: encontrar la cantidad máxima y usuarios simultáneos, en que la aplicación deja de responder o responde de manera errónea.

Pruebas de usabilidad

Las pruebas de usabilidad corresponden a un ambiente de pruebas configurado para verificar el correcto funcionamiento de los elementos interactuantes de una página web con el fin de determinar errores de funcionalidad, mal usos de textos, errores ortográficos entre otros. Por ejemplo: Verificar el correcto funcionamiento y visualización de un módulo nuevo agregado a una página web.

6.4.2. Configuración de las pruebas

Herramientas

• JMeter Para la realización de las pruebas de Carga y de Stress se utilizó la aplicación libre JMeter. JMeter es una aplicación Java desarrollada para realizar pruebas de comportamiento funcional y de rendimiento permitiendo hacer uso de sus módulos para pruebas Web, SOAP, TCP y objetos Java entre otros.

• Excel Para la generación de informes se hace uso de la herramienta Excel, que permite organizar la información para ofrecer métodos de visualización de los datos obtenidos tras ejecutar las pruebas de Carga y de Stress.

• Git Para la corrección de errores y manejo de versiones se hace uso de la aplicación Git que, conectada con el servidor de repositorios en la nube GitHub permite realizar un manejo y anotación de errores además de un correcto seguimiento de las versiones previas.

Pruebas de Carga y pruebas de Stress

Para la realización de las pruebas de Carga y de Stress se configuraron los servidores de AWS con un (1) balanceador de carga y tres (3) instancias que replicaban el backend de ipp5 con el fin de determinar sus capacidades de procesamiento, en cada una de estas instancias se ejecutan los procesos correspondientes al Intérprete y evaluador de expresiones hecho en Haskell y a la sesión destinada para conexión mediante SSH.

Las pruebas de Carga se establecen en un ambiente de producción donde se esperan no más de 40 usuarios en simultáneo. Estas pruebas al igual que las pruebas de Stress se ejecutaron utilizando grupos incrementales en el tiempo, empezando con un grupo de 3 usuarios que en simultáneo envían peticiones a las páginas que cargaban el Intérprete de Haskell y la conexión SSH, cada conjunto de usuarios enviaba peticiones con 0 segundos de diferencia durante un tiempo definido (5 minutos), cuando este tiempo finalizaba se procedía a incrementar el siguiente grupo de la siguiente manera:

Primer grupo: 3 usuarios simultáneos (Prueba de Carga) Segundo grupo: 7 usuarios simultáneos (Prueba de Carga) Tercer grupo: 10 usuarios simultáneos (Prueba de Carga) Cuarto grupo: 20 usuarios simultáneos (Prueba de Carga) Quinto grupo: 50 usuarios simultáneos (Prueba de Stress)

Pruebas de usabilidad

Para la realización de las pruebas de Usabilidad se hace uso de los desarrolladores con el fin de determinar y corregir los errores que se pueden presentar en la aplicación, en casos posteriores previos a entrega a producción se hace necesario pasarlo sobre usuarios reales que prueben la plataforma y permitan establecer sugerencias y encontrar nuevas incidencias a corregir.

6.4.3. Resultados de las pruebas

6.4.4. Pruebas de Carga y Stress

A continuación se muestran los resultados de las pruebas de Carga y Stress, el resultado de estas fue muy satisfactorio dentro de las primeras ejecuciones y al final de estas puesto que tuvo un crecimiento en procesamiento (evidenciando el uso del balanceador de carga) a lo largo de las pruebas, cumplió a cabalidad el requerimiento de las pruebas de carga simulando la cantidad de peticiones esperadas en un ambiente de producción y presentando pocos fallos en las pruebas de Stress.

Los resultados fueron los siguientes:
Primer grupo: 3 usuarios simultáneos (Prueba de Carga). Figuras: 6.2 y 6.3
Segundo grupo: 7 usuarios simultáneos (Prueba de Carga). Gráficos: 6.4 y 6.5
Tercer grupo: 10 usuarios simultáneos (Prueba de Carga). Gráficos: 6.6 y 6.7
Cuarto grupo: 20 usuarios simultáneos (Prueba de Carga). Gráficos: 6.8 y 6.9
Quinto grupo: 50 usuarios simultáneos (Prueba de Carga). Gráficos: 6.11 y 6.11

RESUMEN	
Titulo	
N° de peticiones enviadas	353
Promedio de tiempo de respuesta (ms)	1524.5779
Mediana de tiempo de respuesta	1397
Minimo tiempo de respuesta (ms)	823
Máximo tiempo de respuesta (ms)	4884
Promedio de latencia (ms)	303.645892
Mediana de latencia (ms)	278
Minima latencia (ms)	178
Máxima latencia(ms)	2426
Promedio de tiempo de conexión (ms)	60.3427762
Mediana de tiempo de conexión (ms)	70
Minimo de tiempo de conexión (ms)	0
Máximo tiempo de conexión (ms)	817
Porcentaje de error	100

Figura 6.2: Tabla de resultados: 3 usuarios simultáneos

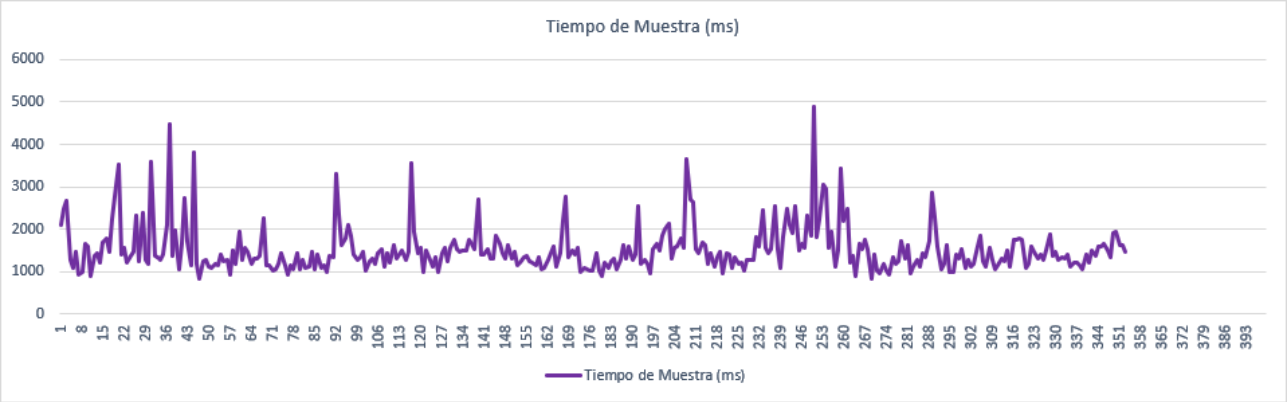

Figura 6.3: Gráfica de resultados: 3 usuarios simultáneos

RESUMEN	
Titulo	
N° de peticiones enviadas	737
Promedio de tiempo de respuesta (ms)	1686.82361
Mediana de tiempo de respuesta	1487
Minimo tiempo de respuesta (ms)	823
Máximo tiempo de respuesta (ms)	5732
Promedio de latencia (ms)	335.860244
Mediana de latencia (ms)	286
Minima latencia (ms)	0
Máxima latencia(ms)	3339
Promedio de tiempo de conexión (ms)	93.2713704
Mediana de tiempo de conexión (ms)	74
Minimo de tiempo de conexión (ms)	0
Máximo tiempo de conexión (ms)	3121
Porcentaje de error	0

Figura 6.4: Tabla de resultados: 7 usuarios simultáneos

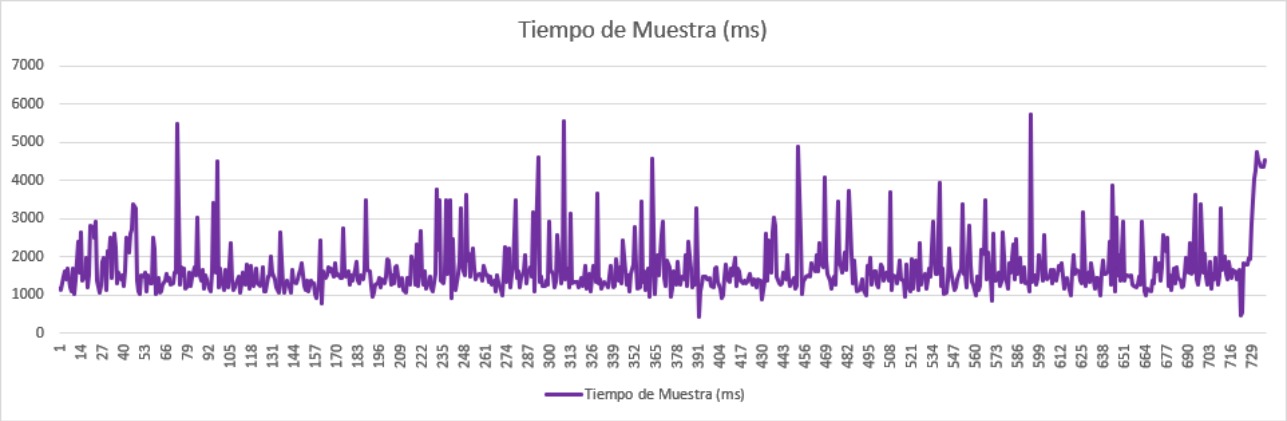

Figura 6.5: Gráfica de resultados: 7 usuarios simultáneos

RESUMEN	
Titulo	
N° de peticiones enviadas	1309
Promedio de tiempo de respuesta (ms)	1359.8961
Mediana de tiempo de respuesta	1285
Minimo tiempo de respuesta (ms)	38
Máximo tiempo de respuesta (ms)	4910
Promedio de latencia (ms)	367.772345
Mediana de latencia (ms)	325
Minima latencia (ms)	37
Máxima latencia(ms)	3709
Promedio de tiempo de conexión (ms)	64.157372
Mediana de tiempo de conexión (ms)	74
Minimo de tiempo de conexión (ms)	0
Máximo tiempo de conexión (ms)	3148
Porcentaje de error	0

Figura 6.6: Tabla de resultados: 10 usuarios simultaneos

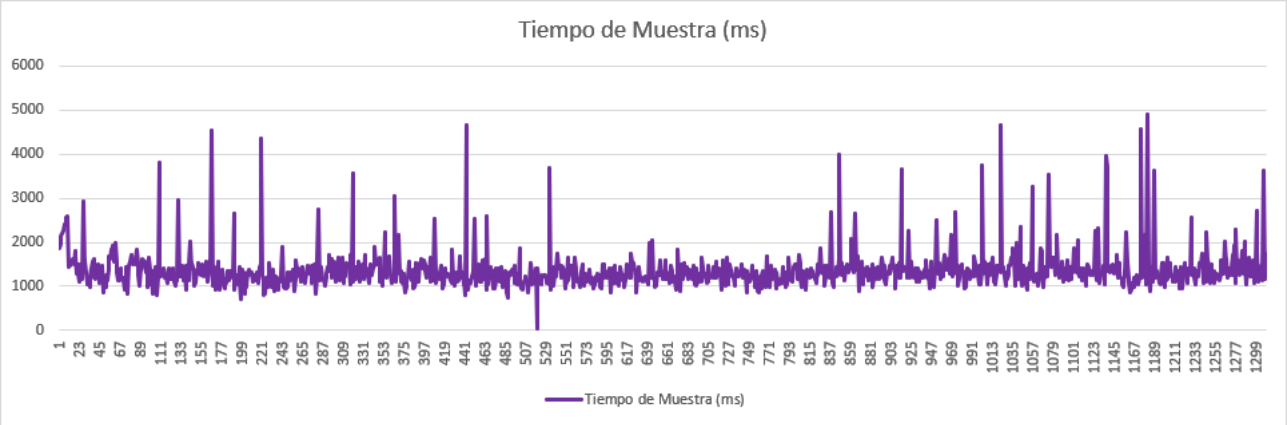

Figura 6.7: Gráfica de resultados: 10 usuarios simultaneos

RESUMEN	
Titulo	
N° de peticiones enviadas	1553
Promedio de tiempo de respuesta (ms)	2304.50741
Mediana de tiempo de respuesta	1943
Minimo tiempo de respuesta (ms)	33
Máximo tiempo de respuesta (ms)	21013
Promedio de latencia (ms)	457.164842
Mediana de latencia (ms)	339
Minima latencia (ms)	0
Máxima latencia(ms)	9353
Promedio de tiempo de conexión (ms)	171.74179
Mediana de tiempo de conexión (ms)	99
Minimo de tiempo de conexión (ms)	0
Máximo tiempo de conexión (ms)	21013
Porcentaje de error	0

Figura 6.8: Tabla de resultados: 20 usuarios simultaneos

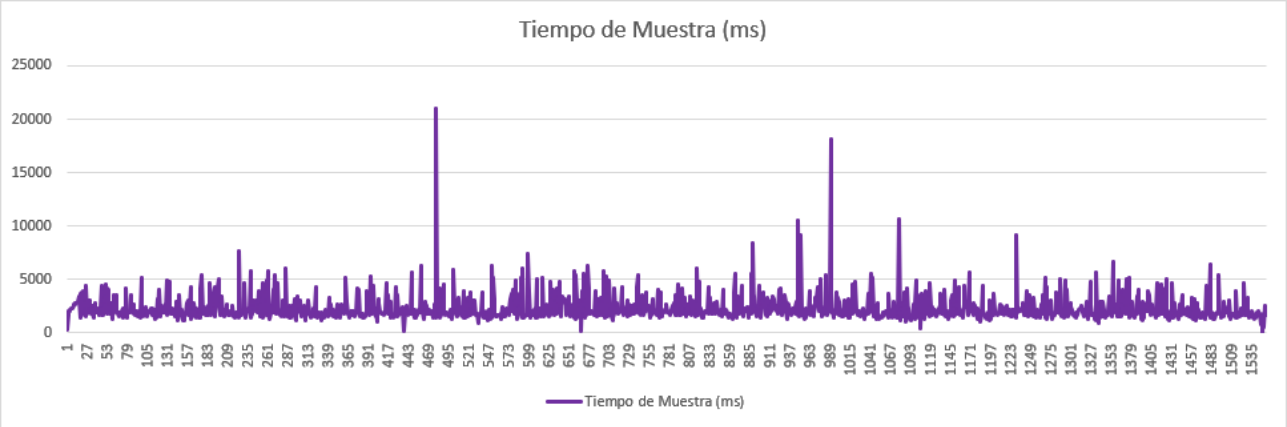

Figura 6.9: Gráfica de resultados: 20 usuarios simultaneos

RESUMEN	
Titulo	
N° de peticiones enviadas	1756
Promedio de tiempo de respuesta (ms)	4764.83542
Mediana de tiempo de respuesta	4126
Minimo tiempo de respuesta (ms)	391
Máximo tiempo de respuesta (ms)	34104
Promedio de latencia (ms)	849.323462
Mediana de latencia (ms)	461
Minima latencia (ms)	0
Máxima latencia(ms)	10644
Promedio de tiempo de conexión (ms)	459.530752
Mediana de tiempo de conexión (ms)	128
Minimo de tiempo de conexión (ms)	0
Máximo tiempo de conexión (ms)	21026
Porcentaje de error	0

Figura 6.10: Tabla de resultados: 50 usuarios simultaneos

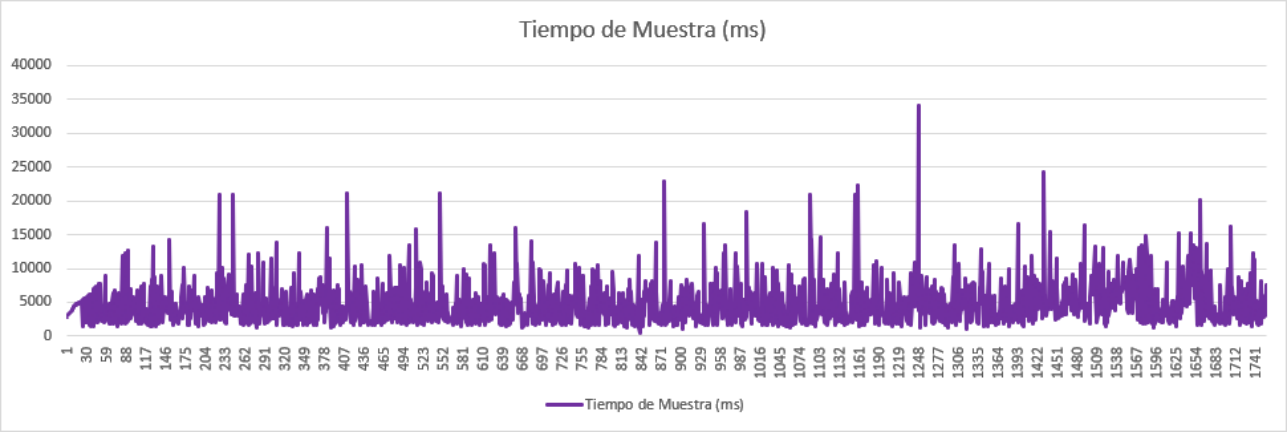

Figura 6.11: Gráfica de resultados: 50 usuarios simultaneos

Capítulo 7

Trabajo Futuro

El proyecto ipp5 se plantea para su posible expansión y su desarrollo, a partir de esta idea se desarrolla una serie de guías u objetivos a tratar para darle un posterior y mayor avance en futuros trabajos.
El trabajo propuesto a futuro se basa en 5 áreas, las cuales son el resultado de las investigaciones y la diversas áreas relacionadas, así como las alternativas que se están tratando actualmente y en desarrollo.

7.1. Redes Sociales

Las redes sociales, son de manera innegable una parte de la vida de la gran mayoría de jóvenes es su principal medio de comunicación, e incluso de expresión con sus amigos y familiares así como un común denominador, si bien muchas instituciones de educación se empeñan en restringir o prohibir su uso, las redes sociales son unas herramientas bastante representativas de la Web 2.0, en las que existen espacios de intercambio de información, lo que fomenta el aprendizaje colaborativo. Ante este panorama, se ha visto el caso de que muchos educadores tengan que adaptar a estos cambios tecnológicos e incluirlos en sus métodos de educación para que los estudiantes se vean realmente involucrados, al ser accesibles y gratuitas, y formar parte del día a día de gran parte de la juventud del mundo, resulta ser del interés colectivo pues están familiarizados con estas redes y son de entretenimiento para muchos. Quizá prefieren tener un debate por Twitter desde la comodidad de sus casas que en un salón de clases, al colaborar con contenido e información, otra de las ventajas es que estas plataformas contribuyen a la socialización entre las partes, pues muchas personas pueden sentirse cohibidas delante de más personas y más abiertas si están frente a un ordenador o teléfono móvil. Por lo tanto, existirá una conexión entre iguales en la que se reconocerá el

valor de cada dato que se comparta, lo que motivará aún más al aprendizaje. Para la plataforma ipp5, se busca realizar un aprovechamiento de las redes sociales para lograr un mayor estimulo, y aprovechamiento de los espacios fuera del aula, dentro del planteamiento se busca, el que las cuentas de usuario puedan ser creadas y enlazadas a los perfiles, así mismo poder generar puntajes, insignias, logros, reconocimientos, entre otros, los cuales puedan compartir en sus redes, para poder generar competencias e invitaciones para el uso del aplicativo. [10]

[1]

7.2. Retos propios y competencia

El Aprendizaje Basado en Retos tiene sus raíces en el Aprendizaje Vivencial, el cual tiene como principio fundamental que los estudiantes aprenden mejor cuando participan de forma activa en experiencias abiertas de aprendizaje, que cuando participan de manera pasiva en actividades estructuradas. En este sentido, el Aprendizaje Vivencial ofrece oportunidades a los estudiantes de aplicar lo que aprenden en situaciones reales donde se enfrentan a problemas, descubren por ellos mismos, prueban soluciones e interactúan con otros estudiantes dentro de un determinado contexto, el Aprendizaje Vivencial es un enfoque integrador del aprendizaje, que combina la experiencia, la cognición y el comportamiento (Akella, 2010).
Dentro del planteamiento y la estructura con la cual fue diseñada la plataforma, según y alineada a las distintas investigaciones, los retos creados y la competitividad pueden llegar a constituir un favor critico en el desarrollo de una experiencia de alta vivencia, esto aprovechando las otras opciones de desarrollo a futuro pueden dar lugar y cabida a competencia entre cursos, entre grupos de estudiantes, y las distintas posibilidades, con esto estimular el desarrollo de las habilidades. [26]

7.3. Aulas Virtuales

Las aulas virtuales son una nueva modalidad educativa que se desarrolla de manera complementaria o independiente a las formas tradicionales de educación, y que surge a partir de la incorporación de las TIC's en los procesos de enseñanza - aprendizaje. Actualmente se utiliza en muchas universidades, escuelas y organizaciones laborales en el que se produce la relación entre los distintos estudiante y el maestro para interactuar entre sí y acceder a la información, esto implica que el estudiante cuando accede a un aula virtual

debe experimentar o vivenciar situaciones potenciales de aprendizaje, de forma similar, a lo que le ocurre en los escenarios presenciales: por ejemplo, leer textos, formular preguntas, resolver problemas, entregar trabajos.[5]
La Cuarta Revolución Industrial que está suponiendo la transformación digital ha traído consigo un cambio en la forma de entender la mayoría de industrias. No hay sector que no esté influenciado por ella, incluido el de la educación. Con el desarrollo de las TIC y su incorporación al sector educativo se ha comenzado el camino hacia un nuevo modelo de aprendizaje. Las nuevas generaciones reclaman una educación en la que los alumnos jueguen un papel más participativo. Es el momento de dejar atrás esas clases en las que solo se escucha y toma notas. Conseguir que el alumno se involucre más en el proceso de aprendizaje es algo más sencillo gracias a la gran variedad de herramientas que facilitan las TIC. Desde el uso de las redes sociales para crear grupos de comunicación entre los alumnos y el profesor, hasta la creación de wikis y blogs para potenciar la creación de contenido para compartir.[4]
En la plataforma ipp5, se puede apuntar a futuro a ser una herramienta, que permita a cada maestro crear su propia aula virtual, y realizar seguimiento al avance tanto individual como grupal de sus alumnos, creación de retos, y seguimiento a estos mismo.

7.4. IA de apoyo y guía

Aunque la Inteligencia Artificial (IA) ya forma parte de nuestras vidas, aún resulta extraño oír hablar de ella en ámbitos como el de la educación, donde la realidad de las aulas avanza a un ritmo mucho más pausado que el de la tecnología. Sin embargo, es precisamente el campo educativo uno de los que más podría verse reforzado y transformado gracias a los nuevos sistemas de inteligencia artificial y su capacidad para contribuir a la personalización del aprendizaje.
De acuerdo con un estudio publicado por el McKinsey Global Institute, las tecnologías de inteligencia artificial pueden ser adaptadas para contribuir a alcanzar los objetivos cruciales de la educación, tales como mejorar la eficiencia de la enseñanza o proporcionar educación de calidad y a bajo costo para todos, así como contribuir a desarrollar las habilidades esenciales para el siglo en que vivimos.
La IA puede ser la respuesta a algunos de los problemas que enfrentan los sistemas educativos actuales, pues permiten la personalización del programa educativo de acuerdo a su perfil y necesidades, además de ofrecer herramientas inteligentes de tutoría que permiten enseñar y aprender habilidades específicas.[9]

En las ventajas y posibles formas de aplicación de la IA en la educación se pueden dar:

- La IA puede crear de forma autónoma hipótesis, localizar conexiones imprevistas, reducir el coste para obtener conocimiento y predictividad.
- Puede automatizar revisiones sistemáticas de literatura académica por lo que se puede utilizar para controlar el plagio y el mal uso de las estadísticas.
- La IA puede ayudar a resolver las diferencias que surgen a la hora de colaboraciones entre diferentes disciplinas.
- A medida que se vuelve más autónoma, la IA puede ayudar a la colaboración entre universidades e instituciones externas, como ejemplo: entre la investigación médica y las prácticas clínicas en el sector de la salud.
- La automatización de las labores más rutinarias y menos gratificantes para los profesores, como por ejemplo las calificaciones de pruebas pueden ser delegadas a esta.
- El uso de chatbots desarrollados como orientadores de forma exitosa ya está siendo de implementados en universidades, por ejemplo, la Universidad técnica de Berlín, para ayudar a los estudiantes a la planificación de los cursos.
- La realidad combinada y la visión por computadora pueden ser más estimulantes y por consiguiente crear más fidelización en los estudiantes, aparte de fomentar una mayor comprensión pues mejorarían el aprendizaje de una manera más intuitiva y adaptativa.

Para la plataforma ipp5, se busca realizar un chatbot, inteligente que en tiempo real pueda dar consejos, guías, estímulos entre otros, para permitir y lograr un mayor acercamiento y estimulo dentro de la misma, así mismo retro alimentaciones sobre el código y preguntas, el cual debería poder responder a las diversas dudas de los estudiantes, así como dar un enlace y un seguimiento personalizado, así como una simulación de un tutor virtual.

7.5. Conclusiones

Si bien se plantea una guía para posibles y futuros avances, debe tomar como una orientación según las investigaciones, mas no una regla o ley a

seguir, las posibilidades son muchas, las presentadas fueron a nuestro criterio los mas aptos, para un desarrollo que abarque las investigaciones así como su adaptación y adopción por parte de los estudiantes.

Capítulo 8

Conclusiones finales y Discusión

8.1. Teorías del aprendizaje

En el desarrollo del proyecto, e incluso en el mismo origen se planteó un diferenciador con respecto a otras plataformas de auto aprendizaje y apoyo a la enseñanza, este factor consiste en el análisis e investigación de teorías y áreas del conocimiento que logren dar un mejor norte al desarrollo del mismo. El conocimiento es un fenómeno con múltiples aspectos, es un fenómeno psicológico, sociológico y biológico como se encontró, su estudio se debe dar desde muchos puntos de vista, También en el estudio del conocimiento científico cabe esta perspectiva científica, representada ya de hecho por la ciencia de la ciencia, también constituir una teoría general del conocimiento (de tipo racionalista: el conocimiento científico como modelo más desarrollado de cualquier forma de conocimiento). No existe una única teoría del conocimiento, de ser así seria la ley del conocimiento, esto lo hace aún mas interesante y apasionante, ver el conocimiento y sus facetas desde muchas áreas y perspectivas así como se trabajo.
Para el desarrollo de la plataforma se optó por investigar las áreas del conocimiento no solo aplicado a la computación, o la enseñanza en programación, se abordo desde la parte biológica, y en general, esto permitió conocer, y abordar muchas investigaciones en el área para nutrir el desarrollo del proyecto.
Entre las diversas investigaciones y aprendizaje, se presentó una limitante a poder dejar plasmado tanto conocimiento, entre ellos el tiempo y los recursos, así como los diversos problemas encontrados en el camino, aun así queda una puerta abierta a un desarrollo futuro, todas estas investigaciones

y recopilaciones quedaron plasmadas, así como su posible y futura aplicación.

8.2. Plataforma ipp5, Arquitectura y tecnologías.

El desarrollo de la plataforma ipp5 dentro del contexto de arquitectura como herramienta de software nos permitió experimentar libremente diferentes herramientas y Frameworks así como aplicar nuestros conocimientos desde los más básicos hasta los más avanzados para poder demostrar nuestro potencial y crear una herramienta meramente fruto de la culminación de nuestros estudios y creatividad.
Una de las grandes ventajas de este proyecto fue su versatilidad, la que nos permitió elegir por cuenta propia cada uno de los elementos, frameworks, herramientas de diseño y plataformas que hacen de ipp5 una herramienta fácil de usar y de entender desde el punto de vista de usuario como de desarrollador, basando sus pilares en el tradicional HTML+css responsivo y un backend apoderado (tras varias versiones) en Java Spring luego de haber jugado con lenguajes como Javascript, Haskell y NodeJs, lo que demuestra que ipp5 como producto software se asemeja a un dispositivo plug and play que permite su fácil extensibilidad y manutención a través del tiempo y las tecnologías así como un desacople de la parte técnica como limitante de la imaginación, que fue factor fundamental al desarrollo de la plataforma y que nos permitió como estudiantes, descubrir nuevas tecnologías y trabajar libre y cómodamente durante el desarrollo del proyecto.
Si bien el avance del proyecto es amplio y las diferentes áreas abordadas fueron de gran interés y alcance, queda abierto a un sin fin de posibilidades, no solo a lo planteado en el capítulo de trabajo futuro, las posibilidades crecen con cada estudio e investigación en la enseñanza y las nuevas tecnologías.

Bibliografía

[1] Compiled by the ACM Education Board. *ACM Curricula Recommendations for Related Computer Science Programs in Vocational-Technical Schools, Community and Junior Colleges, and Health Computing*. Inf. téc. ACM Order No.: 201833. New York, NY, USA, 1983.

[2] Matti Luukkainen Arto Vihavainen Matti Paksula. "Extreme apprenticeship method in teaching programming for beginners". En: *Commun. ACM* 12.1 (mar. de 2011), págs. 93-98. ISSN: 978-1-4503-0500-6. DOI: 0.1145/1953163.1953196. URL: https://doi.org/10.1145/1953163.1953196.

[3] JosÃ©R. Banegas, Fernando RodrÃguez Artalejo y Juan del Rey Calero. "Popper y el problema de la inducciÃen epidemiologÃa". es. En: *Revista EspaÃde Salud PÃ* 74 (ago. de 2000), págs. 00-00. ISSN: 1135-5727. URL: http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1135-57272000000400003&nrm=iso.

[4] Matt Bower. "Virtual Classroom Pedagogy". En: *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education*. SIGCSE '06. Houston, Texas, USA: ACM, 2006, págs. 148-152. ISBN: 1-59593-259-3. DOI: 10.1145/1121341.1121390. URL: http://doi.acm.org/10.1145/1121341.1121390.

[5] Matt Bower. "Virtual Classroom Pedagogy". En: *SIGCSE Bull.* 38.1 (mar. de 2006), págs. 148-152. ISSN: 0097-8418. DOI: 10.1145/1124706.1121390. URL: http://doi.acm.org/10.1145/1124706.1121390.

[6] Campo Burgos. "HACIA UNA TEORÍA EDUCATIVA DESDE EL PENSAMIENTO DE KARL POPPER". En: 11 (ene. de 2011).

[7] Elizabeth F. Churchill, Anne Bowser y Jennifer Preece. "Teaching and Learning Human-computer Interaction: Past, Present, and Future". En: *interactions* 20.2 (mar. de 2013), págs. 44-53. ISSN: 1072-5520. DOI: 10.1145/2427076.2427086. URL: http://doi.acm.org/10.1145/2427076.2427086.

[8] S Combéfis y J Wautelet. "Programming trainings and informatics teaching through online contests". En: 8 (ene. de 2014), págs. 21-34. URL: https://www.researchgate.net/publication/283595158_Programming_trainings_and_informatics_teaching_through_online_contests.

[9] Theodore J. Crovello y Wayne Harvey. "Artificial Intelligence in Education (Panel Session)". En: *Proceedings of the 1985 ACM Annual Conference on The Range of Computing : Mid-80's Perspective: Mid-80's Perspective*. ACM '85. Chairman-Rogers, Jean B. Denver, Colorado, USA: ACM, 1985, págs. 52-. ISBN: 0-89791-170-9. DOI: 10.1145/320435.320454. URL: http://doi.acm.org/10.1145/320435.320454.

[10] Matt Glowatz y Orna O'Brien. "A Technological, Pedagogical and Content Knowledge (TPACK) Framework in the Context of a Social Networking Site Used for Academic Engagement". En: *Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services*. iiWAS '14. Hanoi, Viet Nam: ACM, 2014, págs. 35-39. ISBN: 978-1-4503-3001-5. DOI: 10.1145/2684200.2684346. URL: http://doi.acm.org/10.1145/2684200.2684346.

[11] Thomas T. Hewett y col. *ACM SIGCHI Curricula for Human-Computer Interaction*. Inf. téc. New York, NY, USA, 1992.

[12] Érick Osiris Leines Jiménez. *Introducción al aprendizaje humano*. 2016. URL: https://www.uaeh.edu.mx/scige/boletin/prepa4/n2/e1.html (visitado 08-02-2018).

[13] Monica M. McGill. "Learning to Program with Personal Robots: Influences on Student Motivation". En: *Commun. ACM* 12.1 (mar. de 2012), págs. 36-44. ISSN: 0001-0782. DOI: 10.1145/2133797.2133801. URL: https://doi.org/10.1145/2133797.2133801.

[14] Julie Mills y David Treagust. "Engineering Education, Is Problem-Based or Project-Based Learning the Answer". En: 3 (ene. de 2003).

[15] Iain Milne y Glenn Rowe. "Difficulties in Learning and Teaching Programming—Views of Students and Tutors". En: *Education and Information Technologies* 7.1 (mar. de 2002), págs. 55-66. ISSN: 1573-7608. DOI: 10.1023/A:1015362608943. URL: https://doi.org/10.1023/A:1015362608943.

[16] Brad A. Myers. "A Brief History of Human-computer Interaction Technology". En: *interactions* 5.2 (mar. de 1998), págs. 44-54. ISSN: 1072-5520. DOI: 10.1145/274430.274436. URL: http://doi.acm.org/10.1145/274430.274436.

[17] Manuel Rivas Navarro. *PROCESOS COGNITIVOS Y APRENDIZAJE SIGNIFICATIVO*. Madrid: Subdirección General de Inspección Educativa de la Viceconsejería de Organización Educativa de la Comunidad de Madrid, 2008. ISBN: 0-893-91332-4.

[18] Manuel Rivas Navarro. *Teorías cognitivas del aprendizaje*. Madrid: Subdirección General de Inspección Educativa de la Viceconsejería de Organización Educativa de la Comunidad de Madrid, 2008.

[19] Karl Raimund Popper. *The Two Fundamental Problems of the Theory of Knowledge*. Londres: Troels Eggers Hansen, 1979. ISBN: 978-84-309-3252-8.

[20] John R. Savery. "Overview of Problem-Based Learning: Definitions and Distinctions". En: 1 (mayo de 2006).

[21] T. R. da Silva y E. H. da Silva Aranha. "Online game-based programming learning for high school students — A case study". En: *2015 IEEE Frontiers in Education Conference (FIE)*. Vol. 00. Oct. de 2015, págs. 1-8. DOI: 10.1109/FIE.2015.7344131. URL: doi.ieeecomputersociety.org/10.1109/FIE.2015.7344131.

[22] Matti Luukkainen Stephen Paul Linder Michael J. Fromberger. "An instructional scaffolding approach to teaching software design". En: *Commun. ACM* 21.1 (mar. de 2011), págs. 93-98. ISSN: 978-1-4503-0500-6. DOI: 0.1145/1953163.1953196. URL: https://doi.org/10.1145/1953163.1953196.

[23] Edward L Thorndike. *Education Psychology*. 2.ª ed. New York: Routledge, 1923.

[24] Edward L Thorndike. *Psychology and the Science of Education: Selected Writings of Edward Thorndike*. 4.ª ed. Michigan: Teachers College Press, 1967.

[25] Edward L Thorndike. *The Principles of Teaching: Based on Psychology*. 1.ª ed. New York: Routledge, 1906.

[26] David H. Tobey, Portia Pusey y Josh Chin. "Cybersecurity Competitions in Education: Engaging Learners Through Improved Game Balance". En: *Proceedings of the 2015 ACM SIGMIS Conference on Computers and People Research*. SIGMIS-CPR '15. Newport Beach, California, USA: ACM, 2015, págs. 99-100. ISBN: 978-1-4503-3557-7. DOI: 10.1145/2751957.2751969. URL: http://doi.acm.org/10.1145/2751957.2751969.

[27] Christopher D. Wickens. *An Introduction to Human Factors Engineering*. New Jersey: Pearson Prentice Hall, 204. ISBN: 978-0131837362.