

Facultad de Ingeniería
Escuela de Sistemas

DISERTACIÓN DE GRADO PREVIA A LA OBTENCIÓN DEL TÍTULO: INGENIERA DE SISTEMAS Y COMPUTACIÓN

TEMA:

DESARROLLO DE UNA HERRAMIENTA WEB DE APOYO PARA EL TRATAMIENTO DE NIÑOS CON TRASTORNO POR DÉFICIT DE ATENCIÓN E HIPERACTIVIDAD (TDAH)

AUTORA:

SHIRLEY VALERIA DE LA TORRE LEÓN

DIRECTORA:

ING. SUSANA MASAPANTA

QUITO, 2020

Contenido

1. Capítulo 1: TDAH 10
1.1. Definición 10
1.1.1. Orientación 12
1.1.2. Atención enfocada o focal 12
1.1.3. Atención sostenida 12
1.1.4. Atención selectiva 13
1.1.5. Atención alternada 13
1.1.6. Atención dividida 13
1.2. Síntomas 14
1.3. Prevalencia 15
1.4. Causas 17
1.5. Tratamiento 18
1.5.1. Tratamiento farmacológico 18
1.5.2. Psicoeducación 19
1.5.3. Intervenciones psicológicas 20
2. Capítulo 2: Aplicaciones Web 26
2.1. Definición 26
2.1.1. Aplicaciones del lado del cliente 27
2.1.2. Aplicaciones del lado del servidor 27
2.2. Backend 28
2.3. Frontend 28
2.4. Framework 29
2.4.1. Definición 29
2.4.2. Angular 29

2.4.3. Laravel 30
2.5. Modelo Vista-Controlador 30
2.6. Lenguajes de programación 32
2.6.1. PHP 32
2.6.2. JavaScript 33
3. Capítulo 3: Metodología Scrum 34
3.1. ¿Qué es una metodología? 34
3.2. Scrum 35
3.3. Roles de Scrum 36
3.3.1. Propietario del producto (*Product Owner*) 37
3.3.2. Equipo de Desarrollo (*Development Team*) 37
3.3.3. Scrum Master 38
3.4. Artefactos de Scrum 38
3.4.1. Lista de Producto (*Product Backlog*) 38
3.4.2. Lista de Pendientes del Sprint (*Sprint Backlog*) 39
3.4.3. Incremento 39
3.5. Sprint 40
4. Capítulo 4: Análisis, Diseño y Desarrollo de la aplicación 43
4.1. Análisis 43
4.1.1. Introducción 43
4.1.2. Descripción General 45
4.1.3. Requisitos Específicos 47
4.2. Diseño 65
4.2.1. Diseño Arquitectónico 65
4.2.2. Diagrama de Clases 66

4.2.3. Diccionario de Datos 69
4.2.4. Diagramas de Interacción 72
4.3. Desarrollo 84
4.3.1. Introducción 84
4.3.2. Estándares de programación 84
5. Conclusiones y Recomendaciones 101
5.1. Conclusiones 101
5.2. Recomendaciones 102
Referencias 104

Contenido de Tablas

Tabla 1. Capacidades cognitivas 11
Tabla 2. Prevalencia del TDAH por subtipos 16
Tabla 3. Diagnóstico del TDAH por género. 17
Tabla 1.Características de Usuario. 46
Tabla 2. Excepciones F0.1. 51
Tabla 3. Excepciones F1.1. 52
Tabla 4. Excepciones F1.2. 54
Tabla 5. Excepciones F1.3. 54
Tabla 6. Excepciones F1.4.1. 55
Tabla 7. Excepciones F1.4.2. 56
Tabla 8. Excepciones F2.1. 58
Tabla 9. Excepciones F2.3.1. 59
Tabla 10. Excepciones F2.3.2. 60
Tabla 11. Excepciones F3.1.1. 61
Tabla 12. Excepciones F3.1.2. 62
Tabla 13. Excepciones F3.2. 63
Tabla 14. Requisitos de rendimiento. 63
Tabla 15. Requisitos de diseño. 64
Tabla 16. Atributos del sistema. 64
Tabla 17. Otros requisitos. 65
Tabla 18. Ingreso al sistema 98
Tabla 19. Gestión de Usuarios. 100
Tabla 20. Pruebas Gestión de Avance. 100
Tabla 21. Pruebas Gestión de Ejercicios. 101

Contenido de Figuras

Figura 1.Ejercicio 2 de atención. 22
Figura 2 . Ejercicio 1 de memoria. 23
Figura 3. Resultado ejercicio 1 de memoria. 23
Figura 4. Ejercicio 2 de memoria. 24
Figura 5. Ejercicio 4 de memoria. 24
Figura 6. Modelo Vista-Controlador 31
Figura 7. Elementos de una metodología. 34
Figura 1. Funciones del sistema. 49
Figura 2. F0 Ingreso al sistema. 50
Figura 3. F0.1 Iniciar sesión. 50
Figura 4. F1 Gestión de usuarios. 51
Figura 5. F1.1 Crear Usuario. 52
Figura 6. F1.2 Modificar Usuario. 53
Figura 7. F1.3 Eliminar Usuario. 54
Figura 8. F1.4.1 Consulta General Usuarios. 55
Figura 9. F1.4.2 Consulta Usuarios por parámetro. 56
Figura 10. F2 Gestión de avance. 57
Figura 11. F2.1 Modificar Avance. 57
Figura 12. F2.3.1 Consulta General Avance. 58
Figura 13. F2.3.2 Consulta por parámetro Avance. 59
Figura 14. F3 Gestión de Ejercicios. 60
Figura 15. F3.1.1 Consulta General Ejercicios. 61
Figura 16. F3.1.2 Consulta por parámetro Ejercicios. 61
Figura 17. F3.2 Realizar Ejercicios. 62
Figura 18. Diagrama Conceptual. 65
Figura 19. Diagrama de clases de vistas. 64
Figura 20. Diagrama de Clases de Controladores. 65
Figura 21. Diagrama de entidad-relación 66
Figura 22. Diagrama a nivel lógico 67

Figura 23. Diagrama a nivel físico. 68
Figura 24. Diagrama de Secuencia F0.1. 72
Figura 25. Diagrama de Secuencia F1.1. 73
Figura 26. Diagrama de Secuencia F1.2. 74
Figura 27. Diagrama de Secuencia F1.3. 75
Figura 28. Diagrama de Secuencia F1.4.1. 76
Figura 29. Diagrama de Secuencia F1.4.2. 77
Figura 30. Diagrama de Secuencia F2.1. 78
Figura 31. Diagrama de Secuencia F2.3.1. 79
Figura 32. Diagrama de Secuencia F2.3.2. 80
Figura 33. Diagrama de Secuencia F3.1.1. 81
Figura 34. Diagrama de Secuencia F3.1.2. 82
Figura 35. Diagrama de Secuencia F3.2. 83
Figura 36. Login. 85
Figura 37. Registro. 85
Figura 38. Página principal. 86
Figura 39. El Ahorcado. Elaboración propia 86
Figura 40. Cartas de memoria. 87
Figura 41. Semejanzas y diferencias. 87
Figura 42. Descifrar mensaje. 88
Figura 43. Detección de cantidades. 88
Figura 44. Sombras de figuras. 89
Figura 45. Contar colores. 89
Figura 46. Ordenar los números. 90
Figura 47. Cartas de memoria sonidos. 90
Figura 48. Detección de animales. 91
Figura 49. Historia1. 91
Figura 50. Historia 2. 92
Figura 51. Perfil de usuario. 92
Figura 52. Avance general. 93
Figura 53. Avance por ejercicio. 93

Figura 54. Avance por categoría. 94
Figura 55. Ejercicio correcto. 95
Figura 56. Ejercicio incorrecto. 95

Dedicatoria

Dedico con todo mi amor a mis padres, por haberme forjado como persona y darme la oportunidad de estudiar lo que me gusta. Por todo su esfuerzo y sacrificio para sacarnos adelante a mis hermanas y a mí, por siempre darnos lo mejor y hacernos crecer cada día en lo personal y profesional.

A mis hermanas, por todo su apoyo durante todo este camino, en el cual he tenido momentos fáciles y difíciles, pero ellas me han sabido dar la motivación para continuar y esforzarme cada día más.

A mis abuelitos Eduardo y Margarita, por todo su amor y comprensión en todo momento. Por brindarme apoyo cuando más lo necesitaba, por las palabras y consejos que me han ayudado a ser la persona que soy ahora.

Y a mi niña Canelita, por haberme acompañado cada noche de desvelo y hacerme sentir su compañía cada momento.

Agradecimiento

A mi padre Edgar, por todo su esfuerzo para permitir que termine mis estudios. Por todos los consejos que me han ayudado a superar cada etapa de mi vida y por su apoyo incondicional en cada una de ellas. Por siempre estar para mí cuando lo he necesitado y por ser un gran ejemplo para mí.

A mi madre Alexa, por su apoyo incondicional en cada momento de este largo camino. Por cada palabra de aliento y por todo su cariño a cada momento. Gracias por todo el sacrificio que ha hecho para permitir que siga mis sueños y siempre impulsándome para que vuele alto.

A mi tía Pau, por todo su ayuda y apoyo incondicional. Gracias por ser una persona extraordinaria que nos ha ayudado de varias maneras.

A mis compañeros y maestros, por enseñarme cada día algo nuevo. A mis maestros, gracias por compartir su conocimiento con sus estudiantes, con su sabiduría me motivaron a superarme profesionalmente. A mis compañeros, que algunos se han convertido en grandes amigos, gracias por mostrar cada parte de ustedes, siempre me llevo lo mejor de cada uno.

A mi directora de tesis, Ing. Susana Masapanta, quien con su experiencia y conocimiento me supo guiar de la mejor manera para que pueda realizar este trabajo.

1. Capítulo 1: TDAH

1.1. Definición

El Trastorno por Déficit de Atención e Hiperactividad (TDAH) en uno de los trastornos más frecuentes durante la infancia y adolescencia. Por lo general, se diagnostica después de los 4 y antes de los 12 años, donde la mayoría de los casos se los encuentran entre los 8 y 10 años. Sin embargo, existen algunos casos que son diagnosticados durante la adolescencia o incluso, durante la adultez. (Sulkes, 2018).

Alda, Serrano, Ortiz, y San (2015) afirman que este trastorno afecta entre el 3 al 10% de los niños que se encuentran en la edad escolar[1]. No obstante, también afecta a varios adultos, ya que, si no es diagnosticado y tratado adecuadamente, los síntomas persisten en la vida adulta en un 60% de los casos.

Parekh (2017) de la Asociación Americana de Psiquiatría, afirma que: "los síntomas del TDAH incluyen falta de atención (no poder mantener la concentración), hiperactividad (movimiento excesivo que no se ajusta al entorno) e impulsividad (actos apresurados que ocurren en el momento sin pensar)". Así, la hiperactividad se la identifica cuando existe una demasía de movimiento en diferentes situaciones, por ejemplo, los niños se levantan de su lugar, no mantienen la calma, hablan en exceso, hacen mucho ruido, entre otras cosas. Por otro lado, la inatención o la falta de atención se caracteriza por la fácil distracción y olvido de las cosas, lo que dificulta seguir y organizar instrucciones. Por último, la impulsividad, se identifica por la impaciencia. Los niños no controlan sus acciones y no miden las consecuencias de sus actos, los que padecen TDAH tienden a interrumpir mientras otra persona está hablando y suelen tener problemas con su conducta (Alda et al., 2015).

El TDAH es un trastorno que afecta a la persona que lo padece y a los que pertenecen a su círculo social, puesto que tiene efectos en la parte emocional, social y cognitiva. Esta última contiene los procesos mentales que hacen posible realizar tareas, como la recepción, elaboración y procesamiento de información. (NeuronUP, 2018)

[1] Según la OMS, la edad escolar es aquella comprendida entre los seis y doce años.

Tabla 1. Capacidades cognitivas

Capacidad Cognitiva	Definición
Percepción	Capacidad de adquirir, procesar y dar sentido a la información recibida por los sentidos. [a]
Atención	Capacidad de selección y concentración en información relevante. [a]
Aprendizaje	Obtención de conocimientos nuevos a través de experiencias. [b]
Velocidad de procesamiento	Habilidad de procesar rápida y automáticamente la información. [b]
Memoria	Capacidad de registrar y almacenar información y recuperarla cuando sea necesaria. [a]
Comprensión	Proceso de analizar y reflexionar los datos recibidos. [b]
Lenguaje	Capacidad de utilizar un sistema de signos lingüísticos para expresar pensamientos o sentimientos de forma escrita, oral o por gestos. [a]
Orientación	Capacidad de ser consiente de sí mismo y del entorno que le rodea. Puede ser: personal, temporal y espacial. [c]
Razonamiento	Habilidad de obtener conclusiones, resolver problemas y aprender de nuevos sucesos. [b]
Gnosias	Capacidad del cerebro de identificar información que se ha aprendido antes por medio de los sentidos. [c]
Praxias	Capacidad de realizar movimientos de manera consiente con el fin de lograr un objetivo o plan. [b]

Nota: **Tabla1. Capacidades cognitivas. Elaboración propia. Referencias: [a] (CogniFit, 2020)**, **[b] (Bitbrain, 2018)**, **[c] (NeuronUP, 2018).**

Una de las capacidades más importantes es la atención, debido a que es primordial para la memoria o aprendizaje. Es decir, es la que se encarga que la información llegue al cerebro para poder procesarla. Sohlberg y Mateer (2001) identificaron varios niveles de atención:

orientación o arousal, atención enfocada o focal, atención sostenida, atención selectiva, atención alternada y atención dividida. (Citado por Ostrocky-Solís et al, 2004, p.46).

1.1.1. Orientación

Capacidad de mantener un nivel de conciencia y alerta con sí mismo y su entorno, lo que permite captar estímulos y seguir órdenes. Para conseguir la orientación es necesario la combinación de varias capacidades como memoria, recepción y atención. (Ostrocky-Solís et al., 2004, p.46-47). Existen tres tipo de orientacion: personal, temporal y espacial. Estos aspectos son los que permiten conocer la edad, día, mes, año, ciudad, etc., en el que se encuentra la persona.

1.1.2. Atención enfocada o focal

Capacidad de orientar la atención a estímulos visuales, auditivos y táctiles para poder responderlos. Una persona con deficiencia de atención es incapaz de filtrar estímulos externos o internos y, por ende, se distrae con aquellos que no son importantes y en algunos casos se demora en responder o necesita que se repita la pregunta. (Ostrocky-Solís et al., 2004, p.47-48).

Este tipo de atención tiene una relación con la percepción, puesto que es la que permite identificar aquellos estímulos que son relevantes de los que no lo son. Poder identificar a qué estímulos prestar atención mejora la productividad y desempeño al realizar cualquier actividad. (Rodriguez Puerta, 2018). Todas las personas tienen diferente nivel de concentración en un solo estímulo, incluso varios factores pueden afectar al momento de centrarse en un estímulo como el ruido, motivación, cansancio, entre otros. Para poder identificar a una persona que carece de atención focalizada, un profesional debe realizar varias pruebas que determinen el nivel de dicha capacidad y, en el caso que el paciente muestre dificultad para realizarlas se debe crear un plan de tratamiento.

1.1.3. Atención sostenida

Habilidad de mantener la atención o respuesta a un estímulo específico durante un largo período de tiempo. Torres (2008) señala que el déficit de atención sostenida se debe a la falta de la capacidad de selección, control, orientación y atención; en el caso de los niños que están en sus primeros años, se puede presentar dificultad en el lenguaje. (Citado por Klimenko, 2009).

Las personas que padecen este déficit, también llamado vigilancia, presentan dificultad para terminar una tarea, puesto que solo se enfocan en un estímulo durante corto período de tiempo, como segundos o pocos minutos, y luego dejan de realizar la actividad que estaban haciendo o empiezan otra dejando la anterior incompleta.

1.1.4. Atención selectiva

Capacidad de identificar y, como su nombre lo indica, seleccionar los estímulos relevantes e inhibir los distractores. En otras palabras, la atención selectiva sirve como un filtro, por el cual pasa solo la información que se considera importante para realizar cualquier actividad e ignora lo demás. En el caso de las personas con TDAH, tienen un déficit en este nivel de atención, lo que ocasiona que se sientan atraídos por estímulos externos como sonidos o imágenes llamativas, o incluso, internos como los recuerdos o sentimientos. (Cruz Velarde, 2019).

1.1.5. Atención alternada

Capacidad de cambiar la atención entre diferentes estímulos y moverse entre varias actividades que requieran de capacidades cognitivas distintas. Las personas que tienen déficit en este nivel presentan dificultad para cambiar de actividad después de que han comenzado una, requiriendo claves para poder hacerlo. (Ostrocky-Solís et al., 2004, p. 49-50).

Para la atención alternada se necesita la aplicación de los niveles de atención anteriores: enfocada, sostenida y selectiva, ya que para poder variar entre tareas se requiere que se identifiquen estímulos, mantener la atención en cada una de las actividades y tener la capacidad de separar la información correspondiente a cada una de ellas, a esto se le denomina flexibilidad mental. Para poder reconocer un déficit en este nivel se realizan pruebas de memoria de trabajo, lo que hace referencia a la memoria de corto plazo que permite almacenar información temporalmente y procesarla. (Ramos Usuga, 2018)

1.1.6. Atención dividida

Es la capacidad de realizar varias actividades al mismo tiempo, es decir, se cambia el foco de atención flexiblemente y se da respuestas a diferentes estímulos simultáneamente, para ello se distribuye las capacidades de atención en cada una de las tareas que se realizan. Las personas con TDAH padecen de un déficit de atención dividida por lo cual se les dificulta al realizar varias cosas a la vez. (Velarde, 2019)

1.2. Síntomas

El TDAH se puede identificar cuando el niño se encuentra en la etapa escolar y presenta varios patrones de comportamiento en sus entornos como el hogar o la escuela. El DSM-5[2] clasifica los síntomas del TDAH en dos categorías: 1. falta de atención y 2. hiperactividad e impulsividad. Dentro de las mismas se encuentran varias conductas como la falta de atención, problemas al organizar actividades, incapacidad de quedarse quieto en un lugar durante un largo período de tiempo.

La Asociación Americana de Psiquiatría o APA por sus siglas en inglés, clasifica al TDAH en tres tipos: desatento, hiperactivo o impulsivo y por último la combinación de ambas. Para poder identificar una se necesita conocer los síntomas del paciente e información adicional de las personas que hacen parte de su círculo social como sus padres o profesores, esto hará que un profesional pueda realizar una evaluación y determinar un diagnóstico.

Para el tipo desatento, mínimo seis de los siguientes síntomas o cinco en caso de personas mayores a 17 años, son frecuentes:

- Poca atención a detalles y es descuidado en sus tareas.
- Presenta dificultad para mantenerse concentrado en una terea durante un largo período de tiempo.
- No presta atención cuando alguien le dirige la palabra.
- Problemas al seguir instrucciones y al realizar las actividades propuestas.
- Conflicto en el momento de organizar las labores pendientes.
- Evita realizar tareas que requieren esfuerzo mental sostenido.
- Pérdida de objetos.
- Fácil distracción.
- No recuerda los trabajos que tiene por hacer.

Para el tipo hiperactivo o impulsivo, mínimo seis de los siguientes síntomas o cinco en caso de personas mayores a 17 años, son frecuentes:

[2] *Diagnostic and Statistical Manual of Mental Disorders* o Manual Diagnóstico y Estadístico de los Trastornos Mentales. DSM es el manual utilizado por médicos e investigadores, donde se encuentran descripciones, síntomas y otros criterios, para diagnosticar y clasificar trastornos mentales.

- Se mueve demasiado en su asiento. En algunos casos empieza a golpear con sus manos o pies.
- Le cuesta permanecer sentado por un largo período de tiempo.
- Corre, trepa, salta en lugares donde no debería hacerlo.
- Le cuesta realizar actividades o juegos que requieren tranquilidad o silencio.
- Todo el tiempo está en movimiento.
- Por lo general, brinda las respuestas antes de que se termine de preguntar.
- Es impaciente
- Interrumpe conversaciones o actividades de otras personas.
- Dificultad para mantener silencio, habla en exceso.

Algunos expertos en el área establecen los siete años como edad mínima para diagnosticar, debido a que antes de esta edad los niños aún no presentan patrones de conducta fijos, es decir no se han desarrollado completamente sus reflejos o capacidad de analizar su entorno. (Fundación CADAH, s.f.)

1.3. Prevalencia

El TDAH es uno de los trastornos que más afecta durante la infancia a nivel mundial. Caballero L., Moya A., Romero S. & Hernández M. (2019) afirman que este trastorno afecta al 5% de los niños a nivel mundial. Por otra parte, la doctora Eliana Rodillo realizó un estudio que arrojó que la prevalencia de niños y adolescentes varía entre 5,9% - 7,1% y en adultos jóvenes es de 5%.

Por otro lado, el 30% de los que han sido diagnosticados siguen presentando todos los síntomas en su vida adulta y el 65% solo muestran los más importantes. Algunos de los síntomas que se presentan en la edad adulta son impaciencia, cambio de humor, falta de concentración, dificultad para realizar tareas o incapacidad de mantener las relaciones. (Sulkes, 2018).

En el Ecuador, el Ministerio de Educación registró, hasta el 2017, 7918 niños y adolescentes con TDAH en el sistema público. (Rosero, 2017). Por otra parte, el Instituto de Neurociencias de la Junta de Beneficiencia de Guayaquil (JBG), a mediados del 2016 indicó que se presentaron 592 nuevos casos en lo que iba del año. De esta cifra el 66,7% eran niños y 33,3%

niñas. Victoria Silva, psiquiatra infantil del Insituto menciona que es importante detectar a tiempo este trastorno, de lo contrario se pueden presentar problemas en la adolescencia como baja autoestima, depresión, ansiendad, entre otros. (Valetanga, 2016)

Un estudio realizado en estudiantes entre 14 a 18 años de la ciudad de Quito arrojó que el 2,85% de los evaluados padece del subtipo hiperativo/impulsivo, el 6,5% el subtipo inantento y el 7,3% tiene el subitpo combinado. (Ramos, Bolaños, & Ramos, 2015)

Tabla 2. Prevalencia del TDAH por subtipos

	Mundial	**Ecuador**
Inatento	3,4 – 3,6%[a]	6,5%[b]
Hiperativo/impulsivo	1,1 – 1,3 %[a]	2,85%[b]
Combinado	2,2 – 2,3%[a]	7,3%[b]

Nota. Tabla 2. Prevalencia del TDAH por subtipos. Elaboración propia.
Referencias: [a] González et al. (2015, p.58). [b]Ramos et al. (2015, p. 13-19).

El diagnóstico del TDAH ha ido en aumento los últimos años, según especialistas, esto se debe a que hay una mayor conciencia sobre el trastorno y sus síntomas. Las personas que rodean a los niños (profesores, padres o doctores) están más informadas, lo que provoca que se pueda identificar cuando una persona tiene TDAH.

The Journal of Clinical Psychiatry o Revista de Psiquiatría Clínica, diario oficial de la ASCP[3] publica artículos revisados por profesionales de la psiquiatría que hablan sobre el TDAH, depresión, trastorno bipolar y otros trastornos mentales. La revista publicó un estudio en el cual se identificó un incremento del 43% en la tasa de diagnóstico del TDAH. El trastorno afecta más a la raza blanca, sin embargo, el estudio mostró un aumento del 83% entre los hispanos. Sean Cleary, especialista a cargo de la investigación, indicó que el aumento se debe a un avance en la salud y educación. Una mayor conciencia sobre el trastorno en padres y profesionales ha permitido identificar y diagnosticar más casos, lo que ha permitido brindar un

[3] Sociedad Americana de Patología Clínica *o American Society for Clinical Pathology* por sus siglas en inglés. Asociación profesional que reúne a varios patólogos y especialistas de laboratorio.

tratamiento adecuado para reducir las consecuencias negativas en la etapa adulta (Alonso, 2015).

Tabla 3. Diagnóstico del TDAH por género.

	% de aumento
Mujeres	55%[a]
Hombres	40%[a]

Nota. Tabla 3. Diagnóstico del TDAH por género. Elaboración propia. Referencias: [a] Alonso (2015). Datos recuperados de un estudio que analizó a 190.000 niños entre los años 2000 a 2011.

1.4. Causas

Los expertos hasta el momento no han encontrado las causas concretas del TDAH, sin embargo, existen varios factores que están relacionados con este trastorno. Ramos, Casas y Bosch (2009) en su libro "Comprender el TDAH en adultos" señalan que existen estudios científicos que muestran que el TDAH puede estar relacionado con agentes biológicos. Dentro de estos, los autores identifican dos tipos de factores:

- Los de origen genético, en este caso son los genes los que indicarán la probabilidad de tener el trastorno. Estudios han confirmado que, si uno de los padres lo padece, existe un 56% de probabilidad de que su hijo tenga TDAH y, por otra parte, el 25% de los niños que padecen este trastorno tienen un padre o madre que también lo sufre. Según la APA[4]: "tres de cada cuatro niños con TDAH tienen un familiar con el trastorno".
- Los que están relacionados con los sistemas de neurotransmisión. Aquí la dopamina[5] y noradrenalina[6] juegan un papel importante. Algunas de las funciones de la dopamina es la regulación de la memoria, procesos cognitivos asociados al aprendizaje y toma de decisiones. (Regader, 2019)

[4] Asociación Estadounidense de Psicología o *American Psychological Association*. Es una organización científica y profesional de psicólogos fundada en 1892.
[5] Neurotransmisor que influye en el funcionamiento del cerebro, procesos emocionales, cognitivos y vitales.
[6] Hormona del sistema nervioso central y periférico que aumenta la presión arterial y el ritmo cardíaco y que actúa como neurotransmisor.

Una investigación israelí, dirigida por la doctora Aviva Mimouni-Bloch, indica que los bebés amamantados tienen menos probabilidades de adquirir TDAH. Los investigadores aclaran que no es una causa exacta, pero que si existe relación entre el consumo de leche materna y una menor incidencia del trastorno. Los datos del estudio arrojaron que los niños que consumieron leche de fórmula tenían tres veces más probabilidades de padecer TDAH que los que recibieron leche materna. (Organización Panamericana de la Salud, 2013)

Por otro lado, se ha encontrado conexión entre el TDAH con agentes externos o ambientales. Estos pueden ser la ingesta de drogas, alcohol, tabaco o demasiado estrés de la madre durante el embarazo, complicaciones en el parto, tener hipertensión o diabetes durante la gestación, nacimiento prematuro o que el niño pese menos de 2kg, exposición a químicos, entre otras.

El estrés, ansiedad o depresión son factores que tienen una relación inversa con la atención (principal síntoma del TDAH), es decir que a mayor existencia de estos será menor la atención que tenga la persona. Según estudios, esto puede traer consecuencias negativas como desórdenes psicológicos y emocionales, problemas al relacionarse con otras personas, autoestima baja. (Ostrocky-Solís, Gómez Pérez, Chayo-Dichy, & Flores Lázaro, 2004)

1.5. Tratamiento

Actualmente no hay cura para el TDAH, sin embargo, existen varios tratamientos que sirven para reducir o controlar los síntomas. Ya que se trata de un trastorno que afecta a las acciones cotidianas de las personas que lo padecen, es necesario que un especialista realice un plan de tratamiento personalizado para cada uno, de esa manera se puede afrontar de una forma más eficaz los aspectos de conducta, de interacción social con la familia, escuela, etc.

En el tratamiento se busca controlar la inatención y la hiperactividad que son los principales síntomas del TDAH, con ello se logra mejorar la calidad de vida del niño que padece el trastorno, subir el rendimiento académico y mejorar las relaciones con personas que se encuentran en su entorno.

1.5.1. Tratamiento farmacológico

En 1987 la Academia Americana de Pediatría (APP) indicó que el uso de medicamento en niños debe ser la última de las alternativas de tratamiento, sin embargo, en 1996 rectificó que la medicación debe estar incluida dentro del plan principal de tratamiento en niños

mayores de cinco años, señalando que debe ser acompañada de terapias psicosociales. Por el otro lado, a los niños menores de cinco años se recomienda que solo se debe aplicar las intervenciones psicosociales y en este caso, el medicamento sería una alternativa si lo anterior no da los resultados esperados. (Soutullo Esperón & Mardomingo Sanz, 2010, págs. 66-67)

Este tipo de tratamiento es el más usado, debido a que se utilizan medicamentos estimulantes y no estimulantes (estos últimos no suelen ser tan comunes), que ayudan a minimizar la existencia de síntomas como impulsividad, agresividad y la atención. Se ha demostrado que se obtiene buenos resultados son la medicación, no obstante, puede presentar efectos secundarios en algunas personas. (Bitbrain, 2018). Algunos de estos efectos pueden ser la pérdida de apetito, pérdida de peso, insomnio o irritabilidad. Por lo general, estos efectos suelen corregirse con el paso del tiempo sin necesidad de una intervención, en el caso de que esto no suceda, el profesional debe revisar la dosis del medicamento o en el peor de los casos, debe optar por cambiar de tratamiento. (Loro López et al., 2009)

1.5.2. Psicoeducación

Se enfoca a educar e informar sobre el TDAH a maestros, padres o cualquier adulto que tengan relación con niños que padecen el trastorno. Es importante que conozcan las causas, síntomas, los posibles tratamientos y sus efectos secundarios para que se pueda atender de mejor manera al paciente. Otro aspecto fundamental es conocer sobre los riesgos de no seguir un tratamiento, ya que puede ocasionar que en la vida adulta padezca depresión, agresividad o incluso ser más propenso al consumo de sustancias. (Soutullo Esperón & Mardomingo Sanz, 2010, pág. 66).

Dentro de este tipo de tratamiento se encuentran las intervenciones familiares y educativas. Las primeras enseña a los padres lineamientos de cómo se debe tratar al niño para que no se sienta regañado o juzgado (Bitbrain, 2018) y algunos aspectos que se tendrían que incorporar en el hogar como asignar tareas, agregar reglas específicas pero fácil de seguir o un plan de recompensa cuando el niño cumpla ciertas actividades.

El bajo rendimiento académico es uno de los principales efectos del TDAH porque afecta a las funciones ejecutivas y cognitivas, por lo tanto, las intervenciones educativas buscan

capacitar a los profesores para que puedan adaptar sus clases y faciliten el aprendizaje de los niños. Es recomendable que los pacientes reciban apoyo escolar de manera individual para tratar de mejor manera sus síntomas, se les debe enseñar a organizar sus tareas y planificar su tiempo.

1.5.3. Intervenciones psicológicas

Se busca trabajar las capacidades cognitivas que se encuentran afectadas como la atención, memoria o la concentración. Para la rehabilitación o estimulación cognitiva se utilizan varios ejercicios que desarrollen dichas capacidades, estos tienen niveles de dificultad y pueden ser realizados por niños, adolescentes y adultos.

Una técnica de estimulación cognitiva es la neuropsicología, rama de la Psicología que se encarga del estudio y rehabilitación de capacidades cognitivas[7](Ostrocky-Solís et al., 2004, pág. 21). El primer paso es realizar la evaluación neuropsicológica, en la cual se va a determinar las capacidades que han sido afectadas e identificar qué tanto alteran a las actividades diarias del paciente. Una vez identificados los déficit se procede a realizar la rehabilitación, por medio de la cual se busca mejorar las deficiencias encontradas.

Existen varios ejercicios y actividades de estimulación cognitiva que mejoran las capacidades mentales y permite corregir o, en algunos casos, prevenir algunos déficit. Según estudios, al realizar estos ejercicios se crean conexiones en el cerebro, permitiendo que se desarrolle de mejor manera y así contrarrestar las consecuencias de distintos trastornos. (Ansón Artero et al., 2015). Dentro del plan de rehabilitación se debe tomar en cuenta varios aspectos:

- Autocontrol: se realaciona con la impulsividad, lo cual es fundamental para aquellas personas que padecen TDAH. Aquí se trabaja el aspecto de pensar antes de actuar, analizar las consecuencias de los actos y tomar en cuenta la situación y entorno en que se encuentran para saber como reaccionar.
- Concentración: hace relación a la atención sostenida, es decir que el niño debe adquirir la capacidad de mantener un enfoque a una actividad durante un largo

[7] Capacidades Cognitivas. Ver Tabla 1.

período de tiempo. (Meca Martínez, 2013). Algunos ejercicios que se pueden realizar son[8]:

- Realizar una lectura de letras o números, y en el momento de escuchar el carácter indicado debe realizar una acción. Por ejemplo, si se trabaja con letras, en el momento que escuche la letra "o" debe aplaudir.
- Mostrar un conjunto de números, con ello exiten varias posibilidaes para trabajar con los niños:
  - Seleccionar los números que al sumar den una cantidad señalada. Este ejercicio puede ir incrementando su dificultad, dependiendo del progreso del paciente.
  - Ordenar los números de manera descendente.
  - Ordenar los números de manera ascendente.
- Mostrar de manera desordenada los días de la semana o los meses del año. El niño debe ordenar de manera reversa.
- Juego del ahorcado.

- Atención: el déficit de atención es el principal síntoma que se presenta en el TDAH. Se deben realizar ejercicios que fortalezca la capacidad de filtrar estímulos irrelevantes para centrar la atención en una tarea específica, algunos ejemplos son:
  - Mostrar una matriz que contenga distingos caracteres. El niño debe colorear de diferente color cada uno de los caracteres.

    Ejemplo: Colorear de color verde la celda que contenga la letra "a", de azul la letra "b", de rojo la letra "c" y de amarillo la letra "d".

[8] Ejercicios basados del libro "¿Problemas de atención? Un programa para su estimulación y rehabilitación" (Ostrocky-Solís, Gómez Pérez, Chayo-Dichy, & Flores Lázaro, 2004)

a	b	a	b	c
d	c	b	c	d
b	a	a	d	a
d	b	d	b	d
a	c	d	a	c

Figura 1.Ejercicio 2 de atención.
Elaboración propia.

- Leer un texto pequeño y luego responder preguntas referentes al mismo. Este ejercicio puede ir incrementando de dificultad, haciendo preguntas que necesiten respuestas más detalladas.
- Sopa de letras donde deba encerrar ciertas palabras relacionadas. Por ejemplo los días de la semana o los meses del año.
- Encontrar 7 diferencias entre dos dibujos.

- Memoria: es una de las capacidades que sufre más daño por el paso del tiempo. Para entrenar la memoria se pueden realizar varios ejercicios como:
  - Asignar un símbolo a cada letra del abecedario. Colocar los símbolos para que el niño decifre el mensaje oculto.
    Ejemplo:

A	B	C	D	E	F	G	H	I
☺	✋	♦	●	😐	☹	✎	✂	🔔

J	K	L	M	N	Ñ	O	P	Q
🕮	☎	🕯	☪	☼	✈	✋	👎	💧

R	S	T	U	V	W	X	Y	Z
❄	☯	✉	✞	⌘	■	□	❍	★

Figura 2 . Ejercicio 1 de memoria.
Elaboración propia.

Encontrar el mensaje oculto:

✋	✞	😐	☼	✋	☯

●	🔔	☺	☯

Figura 3. Resultado ejercicio 1 de memoria.
Elaboración propia.

- Presentar al niño un conjunto de símbolos dentro de una tabla, en el cual se le pide hacer una marca cada que visualice un símbolo en específico.

  Ejemplo: marcar con una ‘X’ cuando vea una cara feliz.

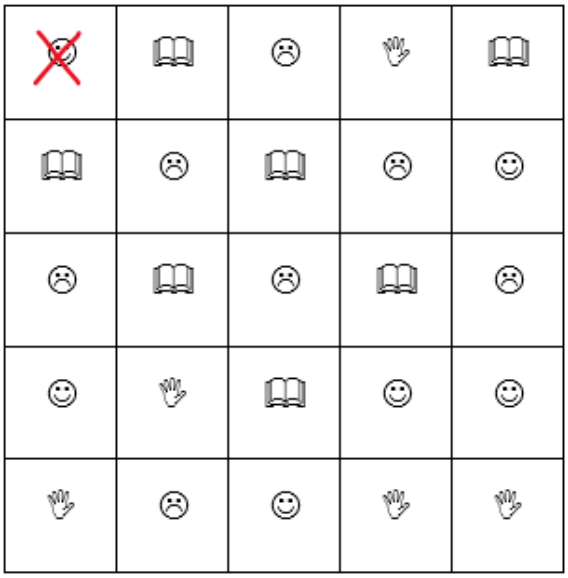


Figura 4. Ejercicio 2 de memoria.
Elaboración propia.

- Se visualiza una imagen de un objeto y posterior debe reconocer la sombra que pertenece a dicho objeto.
- Presentar un pequeño texto para que el niño lo lea, luego presentar casillas vacías con el mismo número de caractéres del texto, el niño debe colocar primero las vocales en el lugar correspondiente, y luego las letras restantes.

  Ejemplo: *"Julia y Marta pasean juntas en el parque"*

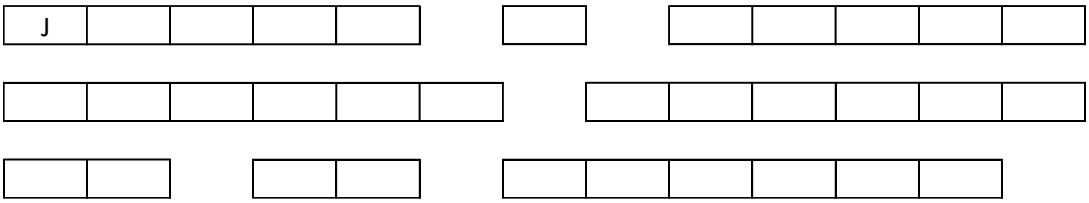


Figura 5. Ejercicio 4 de memoria.
Elaboración propia.

Los ejercicios mencionados anteriormente ayudan a desarrollar varias capacidades cognitivas. Sin embargo, hay que tener en cuenta que estos deben ser diseñados para que se los realice en pequeños

lapsos de tiempo, debido a que un niño con TDAH presenta dificultad para permanecer sentado o quieto realizando una tarea por mucho tiempo.

2. Capítulo 2: Aplicaciones Web

2.1. Definición

Una aplicación web es un tipo especial de aplicación que se encuentra alojada en un servidor, lo que permite poder acceder a ella desde cualquier navegador. Por tal motivo, no es necesario descargar e instalar en el dispositivo. A diferencia de las aplicaciones nativas, las aplicaciones web son desarrolladas de manera global para todas las plataformas, es decir que son compatibles con todos los sistemas operativos.

El uso de este tipo de aplicaciones va en aumento debido a las múltiples ventajas que brinda, algunas de ellas son:

- Ejecución desde cualquier dispositivo conectado a internet
- Multiplataforma
- Disponibilidad
- Actualizaciones automáticas
- No se requiere una instalación para su uso
- Los datos e información se encuentran almacenados en servidores
- Accesibilidad

Las aplicaciones web utilizan la arquitectura de cliente-servidor, la cual es una arquitectura de red. El cliente es aquel que demanda un servicio o recurso al servidor por medio de una petición HTTP (*HyperText Transfer Protocol*). Se pueden identificar dos tipos de clientes: inteligentes y tontos. Los primeros poseen una estructura de hardware y software bien estructurada para trabajar independientemente, y los clientes tontos son aquellos que no poseen dicha estructura y necesitan de un servidor para funcionar. (Luján Mora, 2002).

Por otra parte, los servidores son los que proveen el servicio o recurso que el cliente solicita. Estos esperan la petición para responder código HTML de una página web, en caso de no encontrar los datos correspondientes enviará un mensaje de error. (Vilajosana Guilén & Navarro Moldes, 2019).

Este tipo de arquitectura utiliza un modelo de tres capas:

1. Capa de presentación: también se la conoce como capa cliente. Es la que se refiere a la entrega del usuario, es decir, tiene una interacción con él a través de la interfaz de la aplicación.
2. Capa lógica de negocio: es la que se encarga de la validación de datos, procesamiento de contenido dinámico y generación de información que se va a entregar al usuario. En esta capa se atienden las peticiones del cliente y se comunica con la capa de datos para pedir o almacenar datos.
3. Capa de datos: en esta capa se encuentran los datos que son utilizados en la aplicación, es decir que está relacionada con la manera en la que se almacena la información y cómo se accede a ella. Para interactuar con la base de datos, por lo general es por medio de una API.

Esta arquitectura es utilizada por varias entidades alrededor del mundo que desean comunicación con otras a través de la red. Como ya se mencionó cuenta con dos partes que son cliente y servidor, por tal motivo se pueden identificar dos tipos de aplicaciones:

2.1.1. Aplicaciones del lado del cliente

El código se ejecuta en la máquina del usuario, es decir el servidor proporciona la aplicación, y un navegador es aquel que se encarga de ejecutar a la misma. Para este tipo de aplicaciones los clientes deben tener un navegador compatible con las especificaciones de la aplicación, o también es conocida como *script*. Los lenguajes del lado del cliente suelen ser HTML, CSS y JavaScript, entre los más comunes. Además, que existen *frameworks* que facilitan el trabajo de desarrollo como Angular o Bootstrap. (Berenguel Gómez, 2015)

2.1.2. Aplicaciones del lado del servidor

A diferencia de las aplicaciones del lado del cliente, aquí el servidor es el que se encarga de ejecutar la aplicación, la cual forma un código HTML que es devuelto al mismo para ser enviado al cliente a través del protocolo HTTP. (Vilajosana Guilén & Navarro Moldes, 2019). A ser el servidor el que ejecuta el *script*, se reducen los problemas de compatibilidad con los navegadores clientes. Algunos lenguajes del lado del servidor pueden ser PHP, Java, Python, C#, entre otros. Los *scripts* que se encuentran en el lado del servidor son más

seguros, al contrario que los que están en el lado del cliente, que son archivos planos, lo que los hace más vulnerable a cualquier ataque.

2.2. Backend

Parte del servidor, es decir es la que se encarga que funcione correctamente la capa lógica de la aplicación. En ella se reciben, procesan y envían los datos para ser presentados al cliente. En el *backend* se crea la conexión con la base de datos, se aplican funciones lógicas y se gestiona la seguridad de la aplicación. (Arjonilla, 2020)

Elementos del *backend*:

- **Servidor:** puede ser Apache, XAMPP, Nginx, IIS, entre otros.
- **Base de datos:** algunas son generalmente utilizadas son MySQL, SQL Server, Oracle y PostgreSQL. (Luna, Peña Millahual, & Iacono, 2018, págs. 1-2).
- **Lenguaje de programación:** El *backend* utiliza tecnologías como Python, Ruby, NodeJS, Java, .Net, PHP, etc.
- **API:** *Rest* o *Soap*

2.3. Frontend

Se encuentra en el lado del cliente, es decir la parte de la aplicación que tiene una interacción directa con el usuario. Se utiliza tecnologías web como HTML, CSS o JavaScript para desarrollar los componentes con los que va a interactuar el usuario y se pueden utilizar *frameworks* como React, Angular, Bootstrap, entre otros.

Antes la complejidad se encontraba en el lado del servidor (*backend*), si se requería algún cambio en la página web se necesitaba volver a cargar a la misma, sin embargo, llegó AJAX (*Asynchronous JavaScript y XML*). Esta tecnología es un conjunto de técnicas que permite crear aplicaciones asíncronas, eso quiere decir que puede recibir o enviar información al servidor sin necesidad de recargar la página, ya que estos procesos se los realiza en segundo plano. (Ponguta, 2019)

2.4. Framework

2.4.1. Definición

Es el entorno de trabajo que está compuesto por varios componentes y herramientas que permiten el desarrollo de una aplicación. "Un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta" (Gutiérrez, 2014).

Los *frameworks* brindan varios beneficios, como la optimización de código y tiempo de desarrollo. Otro beneficio es que, al momento de utilizar un *framework* durante el desarrollo de una aplicación, el programador se debe enfocar en el cumplimiento de los requerimientos del software, más no en la programación de bajo nivel que hace funcional al mismo.

2.4.2. Angular

Es un *framework* para el frontend que utiliza el modelo Vista-Controlador (MVC) y fue desarrollado por Google. Mediante este marco de trabajo se pueden realizar, de una manera más sencilla, páginas SPA o *Single-Page-Applications* (Galán, 2020). Este tipo de páginas tienen todos los componentes en un solo archivo HTML, ofreciendo una experiencia más rápida al navegar por el sitio o aplicación web.

Características:

- Velocidad y rendimiento: trasforma el código para máquinas virtuales de JavaScript, la primera vista de la aplicación en PHP, Node.js, .NET u otros servidores son representados en páginas HTML y CSS; utiliza *Component Router*, el cual sirve para que solo se cargue el código que sea necesario.
- Productividad: facilita la creación de la interfaz de la aplicación con una sintaxis sencilla; utiliza CLI Angular, herramientas de línea de comando que permite la instalación de componentes.

Angular brinda varios beneficios:

- Dentro del *framework* se utilizan varias bibliotecas que ofrecen más funcionalidades, lo que permite reducir el consumo de recursos.

- Utilización de APIS para la creación de etiquetas HTML, esto permite reutilizar código y ahorrar tiempo.
- Es un software de código abierto, lo que permite un desarrollo continuo y una gran comunidad de soporte. (Angular, 2020)

2.4.3. Laravel

Marco de aplicación web que puede servir tanto para *backend* como a *frontend*, en él se desarrolla en PHP, por lo que aprovecha las actualizaciones y mejoras de este lenguaje de programación. El creador de Laravel, Taylor Otwell, lanzó la primera versión en el 2011 y desde la versión 3 ha ido aumentado su popularidad hasta convertirse en uno de los *frameworks* más utilizados en la actualidad. (Bean, 2015, págs. 1-2)

Laravel cuenta con varias características:

- *Blade:* gestiona las plantillas de las vistas que son utilizadas en la aplicación, facilitando la reutilización de las que sean necesarias y permite acceder a las variables de PHP.
- *Eloquent:* sistema que permite ingresar, modificar, eliminar o modificar los datos de la base de datos. Esta herramienta facilita el trabajo de programación, puesto que realiza las solicitudes en SQL.
- *Routing:* gestiona las rutas que se utilizan en la aplicación, se pueden manejar por grupo y por tipo de sufijos o prefijos.
- *Middlewares:* controladores que pueden realizarse antes o después de peticiones, aumentando el nivel de seguridad y accesibilidad de los datos.
- Comunidad y documentación: al ser un software libre, cuenta con una gran comunidad en la cual se puede encontrar soporte y algunos paquetes ya hechos que pueden resultad útiles al momento de programar. Así mismo, la documentación de Laravel es muy completa y detalla cada uno de sus componentes.

2.5. Modelo Vista-Controlador

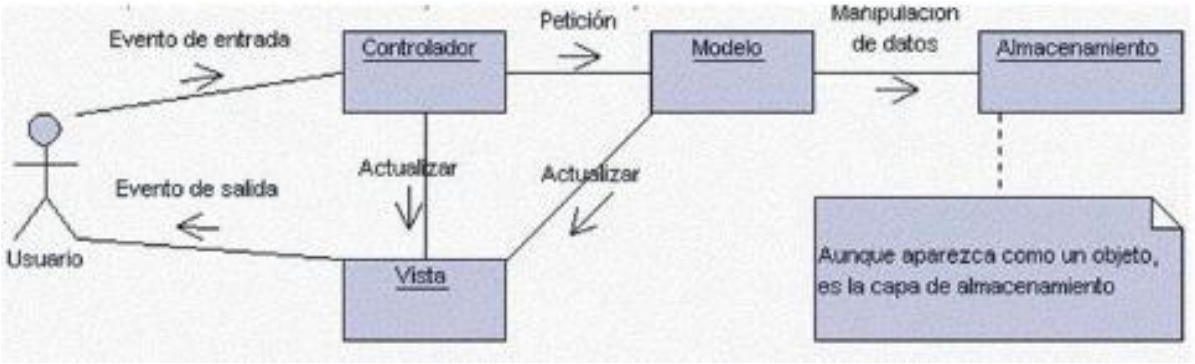
Tipo de arquitectura de software que apareció en la década de los 70´s como parte del lenguaje de programación *Smalltalk* (Brad Green, 2013, pág. 3) Este lenguaje fue creado para fines educativos y fue pionero en permitir depuración en tiempo de ejecución, lenguaje de máquina

virtual y la compilación JIT o *just-in-time*, esta última es una técnica para optimizar el rendimiento del software. Aunque, hoy en día *Smalltalk* no es un lenguaje popular, se sigue utilizando para enseñar a programar a personas que con poca experiencia. (Eng, 2020).

La idea del modelo es separar los datos, la interfaz y la lógica de una aplicación tres componentes diferentes:

1. Modelo: Contiene la gestión de datos y la lógica de negocio. Se encarga de realizar consultas a la base de datos, ya sea para ingresar, modificar, eliminar o consultar información.
2. Vista: Es la interfaz de usuario. La vista contiene la representación de los datos que el usuario envía y recibe, y los componentes con los que interactúa.
3. Controlador: Es la conexión entre el modelo y la vista. El controlador se encarga de recibir las peticiones del usuario, pedir la información necesaria al modelo y enviarla a la vista.

Figura 6. Modelo Vista-Controlador



Nota. Adaptado de Patrón Modelo-Vista-Controlador (pág. 49), por Díaz Y. & Fernández Y, 2012, Interrelación entre los elementos del patrón MVC.

La utilización del modelo al desarrollar una aplicación trae varias ventajas. Una de ellas es que se puede crear cada componente por separado, realizar las respectivas pruebas y unir en tiempo de ejecución. (Díaz González & Fernández Romero, 2012, pág. 48)

2.6. Lenguajes de programación

2.6.1. PHP

PHP (*Hypertext Pre-Processor*) se considera un lenguaje interpretado. Los lenguajes interpretados son aquellos que, por medio de una herramienta llamada intérprete, traduce el código de manera secuencial y la verificación de errores se realiza en tiempo de ejecución. Inicialmente, PHP se creó para registrar las visitas hechas a páginas personales, ahora se ha convertido en uno de los lenguajes de *script* más utilizados (Cobo, 2005, pág. 15). Este lenguaje tiene una de las curvas de aprendizaje más bajas y contiene una sintaxis similar a lenguajes como Java o C.

PHP apareció en 1994 y fue creado por *PHP Group*. Originalmente fue diseñado para el sistema operativo Linux, sin embargo, ha evolucionado y se ha convertido en un lenguaje multiplataforma, debido a que trabaja sobre el servidor web, es compatible para todos los sistemas operativos. (Cobo, 2005, págs. 99-100).

Pérez J. indica que PHP es un lenguaje de lado de servidor y es utilizado para la programación web. Se puede escribir código en PHP dentro de archivos HTML para realizar páginas web dinámicas y, gracias a la rapidez que ofrece, su popularidad ha ido en aumento los últimos años. (Citado por Mina, M. A. E., & Cedeño, A. Y. S., 2018, pág. 32).

PHP posee varias ventajas, algunas de ellas son:

- Curva de aprendizaje baja
- Rapidez
- Multiplataforma, es compatible con sistemas operativos como Linux, Windows y MacOS
- Compatibilidad con la mayoría de las bases de datos como MySQL, Oracle, PostgreSQL, etc.
- Es software de código libre, por lo que tiene un mejoramiento continuo y cuenta con un gran soporte y documentación.
- Cuenta con algunas propiedades de lenguajes orientados a objetos como declaración de clases y utilización de la herencia.
- Integración con código HTML o CSS.

- Permite implementar componentes seguros a las aplicaciones, por lo que disminuye la probabilidad de sufrir algún ataque.
- Permite el manejo de excepciones.

2.6.2. JavaScript

Apareció en la década de los 90's gracias a la necesidad de un lenguaje que se ejecute en el navegador. En esta época empezaron a desarrollarse las primeras aplicaciones, por lo que el contenido de las páginas web iba aumentando su complejidad. Fue creado por Brendan Eich y, en un principio, el lenguaje se llamó *LiveScript*, sin embargo, sus creadores decidieron cambiar su nombre a JavaScript para su primer lanzamiento con el Navegador Netscape 3.0. (Pérez, 2019)

JavaScript es un lenguaje interpretado muy considerado en la actualidad, debido a que se han ido desarrollando varios motores de interpretación que transforman JavaScript a lenguaje de máquina para que la computadora lo pueda ejecutar, estos motores mejoraron el rendimiento de las aplicaciones y aceleraron el procesamiento de código, de tal manera que JavaScript se ha posicionado como una de las mejores opciones para desarrollar en el entorno web (Gauchat, 2012).

Ventajas de JavaScript:

- Es un lenguaje fácil de utilizar y cuenta con varios efectos visuales.
- Multiplataforma.
- Sirve para desarrollar aplicaciones y sitios web dinámicas.
- Es multiparadigma, es decir, soporta distintos paradigmas de programación como: la orientada a objetos, la imperativa y la funcional.
- Es rápido debido a que se ejecuta en el lado del cliente. Es soportado por varios navegadores, los cuales interpretan el código para mostrar en la página web.
- Se mantiene el constante cambio y se integra con facilidad con otras tecnologías.

3. Capítulo 3: Metodología Scrum

3.1. ¿Qué es una metodología?

Una metodología define técnicas y métodos que deben ser empleados para realizar un proyecto. La mayoría de las metodologías cuentan con métodos, herramientas, técnicas, documentación, control y fases; por ende, el cumplimiento de todos estos componentes hace que el desarrollo de software sea predecible. (Trigas Gallego, 2012)

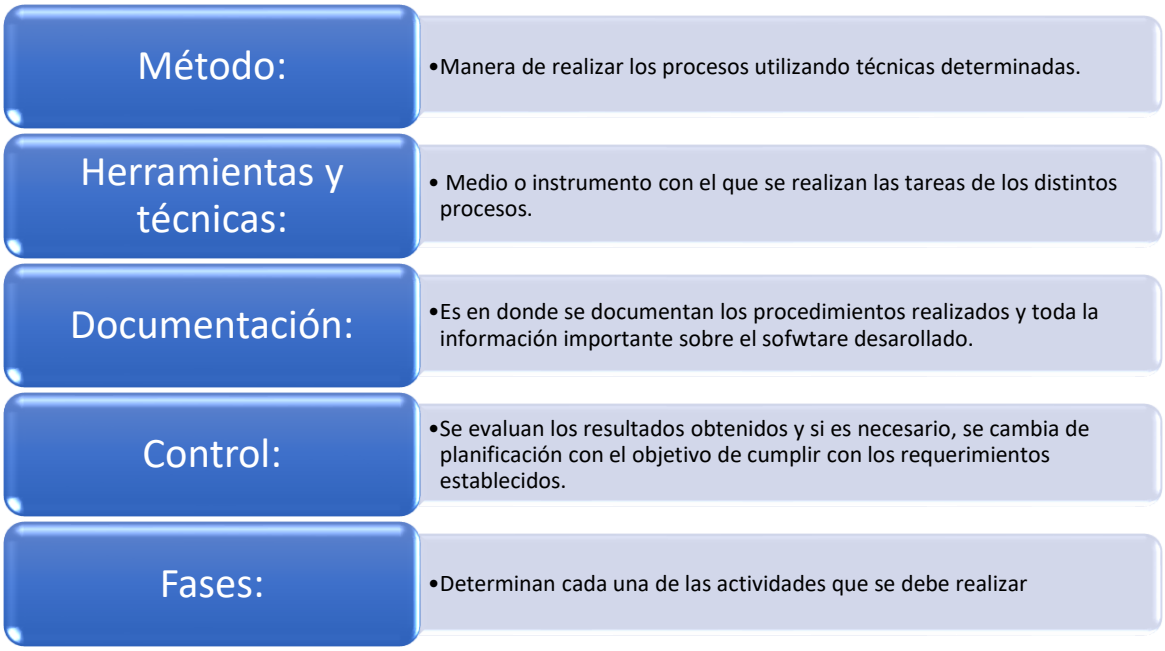


Figura 7. Elementos de una metodología.
Elaboración propia.

Fuente: Trigas G., M. Metodología Scrum. 2012. Gestión de Proyectos Informáticos. Adaptado de: http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf

Existen metodologías tradicionales y ágiles, las primeras se caracterizan por hacer el tiempo de desarrollo más largo y sin opción a cambios, puesto que si existe cualquier modificación en los requerimientos iniciales el costo del proyecto aumentaría. Este tipo de metodologías ofrecen documentación demasiado larga, identificación de las tareas al comienzo del proyecto y aplicación del modelo en cascada, en el cual no se puede volver a una fase anterior.

En la actualidad el desarrollo orientado a la web ha ido aumentando, por lo que se necesita metodologías que permitan cambios constantes. Las metodologías ágiles se caracterizan por

ser iterativas y mantener una buena comunicación entre los miembros del equipo y el cliente. Un aspecto para tomar en cuenta en este tipo de metodologías es que el recurso más importante es el humano, tanto los encargados del desarrollo y los representantes del cliente, con quienes se establecen los requerimientos y, de existir alguna modificación se la pueda realizar sin complicaciones (Trigas Gallego, 2012). En estas metodologías se utilizan el ciclo PDCA (planificar – hacer – verificar – actuar), el cual ayuda a situaciones de cambio y permite la mejora continua. Una ventaja de estas metodologías es que los cambios se los toma como algo necesario para que el producto que se está desarrollando sea de calidad y pueda satisfacer las necesidades del cliente, por tal motivo, en cada ciclo se tiene una fase de *feedback* o *postmortem*, en la cual se analizan los resultados obtenidos y si existe algún cambio se vuelve a planificar un nuevo ciclo donde se resuelva ciertas modificaciones.

3.2. Scrum

Es un marco de trabajo que funciona de manera empírica, es decir el conocimiento se va adquiriendo a través de la experiencia y la toma de decisiones es en base a lo que ya es conocido. Scrum trabaja de manera iterativa e incremental, lo cual permite aumentar la productividad y reducir los riesgos.

Scrum se basa en tres pilares fundamentales:

- Transparencia: todos los aspectos del proyecto deben cumplir los mismos estándares y deben ser bien entendidos por los responsables, de esta manera todos los involucrados tienen claro los objetivos comunes por cumplir.
- Inspección: los miembros del equipo deben evaluar los resultados de cada aspecto del proyecto y asegurarse que van enfocados al cumplimiento de los objetivos.
- Adaptación: si en la inspección se identificaron aspectos que no brindan los resultados esperados, estos deben ser adaptados lo antes posible para minimizar los riesgos.

Para los dos últimos pilares, Scrum maneja cuatro aspectos: Scrum Diario, Planificación, Revisión y Retrospectiva del Sprint. (Schwaber & Sutherland, 2016, págs. 3-4). *Sprint* es la manera en que Scrum llama a una iteración, al ser una metodología ágil trabaja con pequeños ciclos para el desarrollo de software.

Scrum maneja cinco fases dentro de un ciclo de desarrollo:

- Concepto: se identifican las características del producto y se asignan responsables del desarrollo de cada una de ellas.
- Especulación: se establecen los costos y las fechas límites de entrega de versiones. En cada iteración se definen los requerimientos y se listan las funcionalidades que se espera obtener.
- Exploración: se agregan al producto las funcionales identificadas en la fase anterior.
- Revisión: se realizan pruebas del producto y se verifica que esté de acorde con los objetivos planteados.
- Cierre: se entrega la versión del producto en la fecha establecida y se establece los cambios que se deban realizar en el siguiente ciclo.

> Una característica particular de la metodología Scrum es que se realizan reuniones diarias, en ellas se revisa el progreso realizado por cada uno de los integrantes del equipo de desarrollo. (Trigas Gallego, 2012, pág. 33)

3.3. Roles de Scrum

Todas las personas involucradas en un proyecto forman parte del Equipo Scrum, en él se encuentran los desarrolladores, dueño de producto y el Scrum Máster. Para que el equipo tenga éxito en sus metas, cada integrante debe estar comprometido a trabajar en conjunto para lograr los objetivos y contar con la flexibilidad ante cualquier cambio que se presente en el desarrollo del producto. El equipo elige la mejor manera para trabajar en el proyecto y no dependen de ninguna persona externa al equipo. (Schwaber & Sutherland, 2016, pág. 5)

El equipo trabaja de manera iterativa e incremental, es decir, en cada iteración entregan una versión del producto, en donde cada vez se acercan más a la versión final, manteniendo una versión útil del mismo en cada ciclo.

En el Equipo Scrum se encuentran las personas comprometidas en el proyecto y proceso: Propietario del producto (*Product Owner*), *Scrum Master* y el Equipo de desarrollo. Por otra parte, las personas involucradas, es decir que no son parte del proceso Scrum, pero son las que brindan retroalimentación al equipo de desarrollo para poder planificar las iteraciones, estas son: Usuarios, son los destinatarios finales del producto; *Stakeholders*, reciben algún beneficio

de la aplicación y están involucrados en la revisión del *Sprint*; y los *Managers*, aquellos que brindan apoyo en la toma de decisiones y establecimiento de objetivos.

3.3.1. Propietario del producto (*Product Owner*)

Es aquel que comunica los requerimientos del cliente y de tomar todas las decisiones para cumplir con las mismas. Las necesidades son escritas en el *Product Backlog* y se las ordena por prioridad, esto ayuda a maximizar la productividad del equipo y el valor del producto. El propietario del producto debe asegurarse que los puntos dentro del listado del Producto o *Product Backlog* sean claros para todos y reflejen claramente los objetivos.

Product Owner es representado por una sola persona a la cual los otros miembros del equipo deben hacer caso y seguir los lineamientos escritos en el *Product Backlog*, si se requiere algún cambio en estos elementos, él debe de aprobarlos y, por consiguiente, escribirlos en la Lista. (Schwaber & Sutherland, 2016, págs. 5-6).

3.3.2. Equipo de Desarrollo (*Development Team*)

Equipo de cinco a nueve personas encargadas de realizar las tareas escritas en el *Product Backlog*, se organizan autónomamente para poder llegar a los objetivos. Al final de cada Sprint, el equipo de desarrollo entrega una versión del producto que puede ser lanzada a producción como producto terminado y generan el *Sprint Backlog* o Lista de Pendientes de *Sprint*.

Características del Equipo de Desarrollo:

- Es multifuncional, es decir, sus integrantes cuentan con las habilidades requeridas para terminar un *Sprint* y generar el producto.
- Los miembros del equipo se organizan para realizar las tareas asignadas y nadie fuera de este puede indicar como realizar los elementos de la Lista del Producto.
- No existe una jerarquía, es decir, todos cumplen el rol de desarrolladores.
- A pesar de que se dividan las tareas a cada miembro, todos son los responsables del resultado.

3.3.3. Scrum Master

Persona encargada de supervisar que el desarrollo del producto vaya de manera adecuada, verificar que la metodología sea aplicada correctamente y eliminar cualquier factor que afecte a la creación del producto. Este rol cumple el papel de intermediario entre el Equipo de Desarrollo y personas externas al equipo, puesto que él modera cuando se debe tener interacción con el equipo.

El *Scrum Master* ayuda al Dueño de Producto a manejar de manera correcta la Lista de Producto y a que ésta contenga información que sea entendible, asegurar que todo el Equipo tenga claro los objetivos, planificación y alcance del producto y facilitar los recursos que sean requeridos. Por otra parte, el apoyo hacia el equipo de desarrollo es en la organización de este, colaboración para el desarrollo de un producto de calidad y entrenamiento para nuevos entornos que sean necesarios adoptar para el proyecto. (Scrum.org, 2020)

3.4. Artefactos de Scrum

3.4.1. Lista de Producto (*Product Backlog*)

Lista de requerimientos o funcionalidades que se almacenan por orden de prioridad. El cliente y el *Scrum Master* son los encargados de identificar los requerimientos, tanto funcionales como no funcionales, de la lista, y este último se encarga de estimar los costos relacionados para cada requerimiento e identificar los riesgos señalando su plan de solución. Sin embargo, el Dueño de Producto es el responsable de la Lista de Producto, de todo su contenido y del orden de los requerimientos identificados. La lista debe ser realizada antes de comenzar un *Sprint*, sin embargo, solo debe contener los objetivos principales del proyecto porque los secundarios pueden ser identificados durante el desarrollo (Trigas Gallego, 2012, pág. 37).

La Lista de Producto va a existir siempre que el producto exista, es decir, que a medida que el producto se va desarrollando pueden ir incrementando los elementos de la lista y estos pueden ser incluidos por cualquier miembro del equipo en la reunión de Planificación de *Sprint*. A pesar de que cualquier involucrado en el proyecto puede sugerir incluir otros elementos, el encargado de aumentar, ordenar y priorizar dichos elementos es el Dueño de Producto.

La Lista de Producto contiene la descripción de cada una de las funcionalidades del producto a las que se les denomina historias de Usuario. Estas se crean con la intervención del cliente y del equipo y se van desarrollando a lo largo de la realización del proyecto. Las historias de usuario contienen tres fases que se las conoce como "Las 3C", y estas son: *Card*, descripción de la historia; *Conversation*, conversación entre las partes involucradas para plantear el objetivo; y, *Confirmation*: determinación de los detalles y el límite (Trigas Gallego, 2012, págs. 37-38).

3.4.2. Lista de Pendientes del Sprint (*Sprint Backlog*)

El Equipo de Desarrollo elige los elementos de la Lista de Producto que serán desarrollados en el *Sprint* y establece un plan para alcanzar el Objetivo de *Sprint*, estos dos componentes forman el *Sprint Backlog*. Se deben establecer los detalles de las funcionalidades que serán entregadas en el próximo Incremento, pero estas no pueden ser cambiadas una vez que comienza el desarrollo del *Sprint* si ponen en riesgo el objetivo. Cada elemento de la lista es asignado a un responsable para su elaboración y se establecen fechas límites de entrega, de esta manera se pueden identificar fácilmente los riesgos o problemas que puedan existir y emplear el plan de solución (Schwaber & Sutherland, 2016, págs. 15-16).

La Lista de Pendientes ordena por prioridad las tareas que son seleccionadas en la reunión de Planificación de Sprint, a lo largo del desarrollo el equipo puede identificar más tareas adicionales, pero estas deben ser aprobadas y entendidas por todos los involucrados en la creación del producto (James & Walter, 2010).

3.4.3. Incremento

Es el resultado de las tareas terminadas del *Sprint Backlog* más el valor de los resultados de los anteriores *Sprints*. El incremento debe ser un producto funcional independientemente que el cliente lo quiera lanzar a producción o no. Sin embargo, el cliente puede hacer observaciones sobre el Incremento y, en el caso que existan cambios, se realiza una planificación de *Sprint* para el inicio de otro ciclo.

3.5. Sprint

Es la clave de la metodología y se considera un bloque de tiempo en el cual se genera un incremento del producto. Un *Sprint* no comienza hasta que el anterior termine y en cada uno de ellos contienen los Scrum Diarios, trabajo de desarrollo, Revisión de *Sprint* y la Retrospectiva de *Sprint*. (eclass, 2020)

Un *Sprint* es un subproyecto que dura generalmente un mes y genera un producto, si este dura más tiempo existe la posibilidad de aumentar los riesgos y complejidad del proyecto. Cuando se inicia un *Sprint* no se pueden realizar cambios que alteran al Objetivo de *Sprint*, pero se puede cambiar el alcance mediante una negociación entre el Dueño de *Sprint* y el equipo de desarrollo.

Eventos de un *Sprint*:

- Cancelación: No es común, sin embargo, en el caso que suceda únicamente el Dueño de Producto puede hacerlo. Las posibles causas de la cancelación son cuando el objetivo deja de ser funcional, el Sprint no brinda un valor a la organización o esta cambia de metas o tecnologías que impidan seguir con el progreso del proyecto. Cancelar un *Sprint* debe ser en casos especiales ya que consumen recursos, puesto que se debe volver a realizar el proceso de planificación y estimación.

  Al momento de cancelar un *Sprint* se deben revisar los elementos de la Lista de Producto que fueron realizados y verificar si el incremento puede ser lanzado a producción. Aquellos elementos que no fueron realizados se vuelven a colocar en la lista.

- Planificación: se realiza durante la reunión llamada “Planificación de *Sprint*”, en donde debe estar presente todo el Equipo Scrum. Durante la planificación se deben responder las preguntas del qué y cómo se va a realizar el trabajo durante el *Sprint*, para ello se necesita la colaboración del *Scrum Master*, Dueño de Producto y el Equipo de Desarrollo para determinar los elementos que se trabajaran en el ciclo. Además, el *Scrum Master* debe asegurarse que los objetivos sean entendidos por todos para que al final se cumplan el objetivo del *sprint* (Rehkopf, 2020).

Una vez que se ha establecido el objetivo de *Sprint* se debe establecer el plan para identificar los elementos de la Lista de pendientes y establecer el calendario que se debe cumplir para entregarlos. El plan y los elementos se denominan *Backlog* de *Sprint*.

El Equipo de Desarrollo pueden negociar los elementos si consideran que hay algún inconveniente con lo establecido, en este caso el Dueño de Producto puede reorganizar los elementos de la lista o conseguir más personas que colaboren al equipo si considera necesario. Al final de la reunión, cuando han llegado a un acuerdo, el equipo debe explicar cómo se han organizado para cumplir con todos los elementos de la Lista de Producto dentro de las fechas establecidas.

Elementos de un *Sprint*:

- Objetivo de *Sprint* (*Sprint Goal*): es la meta que se establece en la reunión de planificación. El objetivo define el por qué se crea el incremento del producto y ayuda a que todos los miembros del equipo desarrollen en un solo sentido y no presenten resultados separados y diferentes a lo establecido.
- Scrum Diario (*Daily Scrum*): reunión de no más de 15 minutos que se realiza en la misma hora y lugar todos los días. En ella se trata sobre lo que los miembros del Equipo de Desarrollo han hecho durante las últimas 24 horas y qué es lo que harán las siguientes. Adicional a esto se puede encontrar solución a ciertos inconvenientes que se han encontrado o tratar de identificar los factores que pueden interferir en el desarrollo del producto.

  El responsable de esta reunión es todo el Equipo de Desarrollo y el *Scrum Master* solo se encarga de verificar que solo los miembros del equipo participen en la reunión y que esta no dure más de los establecido, que es 15 minutos. (Schwaber & Sutherland, 2016, págs. 11-12)

- Revisión de *Sprint* (*Sprint Review*): como su nombre lo indica, se revisa el trabajo realizado durante el *Sprint* y se muestra los resultados a los interesados antes de entrar a la etapa de producción (Rehkopf, 2020). Los que deben asistir a la revisión son el Equipo Scrum, los interesados en el producto y el Dueño de Producto. Este último se

encarga de informar que elementos se han realizado y cuáles no, además de informar los problemas encontrados en el *Sprint* y explicar la manera en que estos fueron resueltos.

- Retrospectiva de *Sprint* (*Sprint Retrospective*): La retrospectiva se debe realizar después de la revisión y antes de la planificación del *Sprint*, debido a que se identifican las áreas que se pueden mejorar para el siguiente ciclo y crear un plan de mejoras.

  El *Scrum Master* es el responsable de esta reunión y en ella se deben analizar el manejo de recursos utilizados en el último *Sprint*, identificar los elementos realizados y la manera de mejorarlos, por último, se establece un plan para aplicar las mejoras establecidas.

4. Capítulo 4: Análisis, Diseño y Desarrollo de la aplicación

4.1. Análisis

El objetivo de la fase de análisis es identificar los requerimientos de la aplicación, así como asignar las tareas a los integrantes del equipo de trabajo y los recursos que sean necesarios para la elaboración de las tareas asignadas.

Se presentará la descripción de los requerimientos del producto a desarrollarse, los cuales deben ser entendidos y aprobados por las partes involucradas. La especificación se basa en el estándar IEEE 830-1998, el cual brinda recomendaciones para realizar el documento Especificación de Requerimientos de Software -ERS o SRS por sus siglas en inglés.

El documento tiene como objetivo reunir los requerimientos funcionales y no funcionales de la aplicación web a desarrollar, de esta manera la persona encargada en desarrollar tendrá claro el objetivo y comportamiento del producto. En este trabajo, el proceso de análisis y levantamiento de requerimientos se realiza en colaboración de un profesional en el área de psicología, quien brinda información sobre los parámetros que deben cumplir los ejercicios implementados en la aplicación.

El sistema busca solucionar la falta de una aplicación web que contenga ejercicios para el tratamiento de niños con TDAH, los cuales pueden ser realizados desde cualquier dispositivo que tenga conexión a internet.

4.1.1. Introducción

El documento de especificación de requerimientos de software contiene la descripción de las funcionalidades del sistema, con el objetivo de detallar cada componente. Se compone de 4 secciones: propósito, ámbito del sistema, referencias y visión general del documento.

4.1.1.1. Propósito

El propósito del documento es definir los requerimientos funcionales y no funcionales de una aplicación web para el tratamiento de niños con TDAH. Este documento se realiza para asegurar un total entendimiento de cada una de las funcionalidades a desarrollar.

4.1.1.2. Ámbito del Sistema

La aplicación desarrollada en el presente trabajo tiene como objetivo brindar una herramienta de apoyo para el tratamiento de niños con TDAH. En ella se encontrarán actividades a realizar por los niños, las cuales siguen los parámetros establecidos por un psicólogo. El sistema tendrá las siguientes características:

- Los ejercicios desarrollados en la aplicación estarán agrupados de acuerdo con la capacidad cognitiva que se busca estimular. De esta manera se podrá abordar con técnicas de intervención de forma específica.
- Los gráficos o texto que se utilicen deben ser grandes y coloridos para llamar la atención de los niños.
- Los ejercicios establecidos deben ser diseñados para que no duren más de 10 minutos, debido a que la aplicación será utilizada por niños con déficit de atención.

4.1.1.3. Definiciones, acrónimos y abreviaturas

Definiciones:

- **Actor:** persona que interactúa con el sistema. Cada actor cuenta con diferentes permisos para el acceso e interacción con las diferentes funcionalidades.
- **Gestión:** manejo de una entidad[9] a la que podrá ingresar, modificar, eliminar o consultar información.
- **Bases de datos:** almacén de datos de manera ordenada para su posterior uso.
- **Casos de Uso:** diagrama UML en el que se pueden identificar las funcionalidades del sistema y los actores que están relacionados con cada una de ellas.
- **Usuario:** se refiere a la persona que realiza los ejercicios de tratamiento a través de la aplicación.
- **Administrador:** persona que tiene total acceso a las funcionalidades del sistema desarrollado.
- **MySQL:** motor de base de datos que se utiliza para el manejo de información.

[9] Unidad de base de datos que contiene información y que está compuesta por un conjunto de atributos y propiedades que la caracterizan.

Acrónimos:

- **MVC:** Modelo Vista-Controlador
- **ERS:** Especificación de Requerimientos de Software
- **TDAH:** Trastorno por Déficit de Atención e Hiperactividad
- **IDE:** Entorno de desarrollo integrado

Abreviaturas:

- **F:** Funcionalidad
- **SW:** Software
- **HD:** Hardware
- **BDD:** Bases de datos
- **N.A.:** No aplica

4.1.1.4. Referencias

N.A.

4.1.1.5. Visión general del documento

A continuación, se hará una descripción general del producto, en la cual se puede encontrar: perspectiva y funciones del producto, características de los usuarios, restricciones, suposiciones, dependencias y requisitos futuros. Siguiente a esto se identificarán los requisitos específicos, para ello se establecerán las interfaces externas, funciones, requisitos de rendimiento, restricciones de diseño, atributos del sistema y otros requisitos.

4.1.2. Descripción General

En esta sección del documento se indica si el sistema es dependiente o independiente de otros sistemas y las funcionalidades que este va a contener. Por otra parte, se establecen los tipos de usuarios que van a tener interacción con el sistema y sus características.

A continuación, se establecen las restricciones, son aquellas limitaciones como políticas de la empresa, limitaciones de hardware, entre otras. Por último, se describen las suposiciones, dependencias y futuros requisitos del sistema.

4.1.2.1. Perspectiva del Producto

El sistema es independiente, es decir, no es parte de un sistema mayor. El sistema se compone de procesos que guarden información sobre el avance de cada usuario en los ejercicios realizados.

4.1.2.2. Funciones del Producto

F0 Ingreso al sistema: el usuario ingresa al sistema con sus credenciales previamente creadas. Si no se han creado, se realiza el registro de usuario con un nombre de usuario y contraseña.

F1 Gestión de usuario: Se realiza el ingreso, modificación, eliminación y consulta de usuarios.

F2 Gestión de avances: se realiza el ingreso, modificación y consulta del avance de cada usuario.

F3 Gestión de Ejercicios: se realiza la consulta y ejecución de los ejercicios existentes en el sistema.

4.1.2.3. Características de los Usuarios

Tipo de usuario	Usuario
Nivel educacional	Básica
Experiencia	Baja
Experiencia técnica	Baja

Tabla 4.Características de Usuario.
Elaboración propia.

Tipo de usuario	Administrador
Nivel educacional	Superior
Experiencia	Alta
Experiencia técnica	Alta

Tabla 2. Características de Administrador.
Elaboración propia.

4.1.2.4. Restricciones

- Una vez creado un usuario, no se podrá crear otro con el mismo correo electrónico.
- Para el desarrollo del sistema se utilizará JavaScript como lenguaje de programación para el backend con el uso del framework Angular, y PHP como lenguaje de programación para el backend con el uso del framework Laravel.
- La metodología para el desarrollo de software es Scrum.
- Se trabajará con VisualCode como IDE y se utilizará PowerDesigner para la creación de los diagramas.
- Al tratarse de una aplicación web, se utilizará el navegador Chrome, Firefox, Microsoft Edge, Samsung Internet y Safari para los respectivos controles.

4.1.2.5. Suposiciones y Dependencias

Se asume que los requerimientos señalados en este documento son estables y satisface las necesidades de los usuarios.

La aplicación podrá ser ejecutada desde cualquier navegador, sea de escritorio o móvil.

Se asume que se trabajará con la aplicación sobre navegadores actualizados para aprovechar las características del sistema. Para un mejor funcionamiento de la aplicación de debe contar con la última versión estable de cada navegador:

- Google Chrome: 86.0.4240.183
- Firefox: 82.0.2
- Microsoft Edge: 86.0.622.38
- Opera: 71.0.3770.228

4.1.3. Requisitos Específicos

En esta sección se presentan los requisitos con un mayor detalle, con el objetivo de que los desarrolladores entiendan claramente las necesidades del sistema. Cada requisito debe reflejar el comportamiento de los usuarios finales que interactúan con el sistema.

Se va a desarrollar la aplicación en un entorno web, la cual servirá para apoyo del tratamiento de niños con Trastorno con Déficit de Atención e Hiperactividad (TDAH).

4.1.3.1. Interfaces Externas

- (Código RIE01) Interfaz de usuario: los gráficos y textos que se utilicen para los ejercicios de rehabilitación deben ser llamativos, para ello se deben utilizar colores y el tamaño de letra debe ser grande.
- (Código RIE02) Interfaz de software y hardware: la interfaz de software dependerá del navegador en donde se ejecute la aplicación. Por otro lado, no se requiere una interfaz de hardware para esta aplicación.
- (Código RIE03) Interfaces de comunicaciones: No se requiere una interfaz de comunicación.

4.1.3.2. Funciones

F0 Ingreso al Sistema: Los actores ingresan al sistema dependiendo del rol que tengan (usuario o administrador). En el cual tendrán diferentes funciones y vistas.

F1 Gestión de usuarios: El perfil de administrador podrá crear, borrar y consultar los usuarios existentes en el sistema. Y si es perfil usuario, podrá acceder a su propia información con posibilidad de editar sus datos y eliminar el perfil.

En el caso que el usuario ya cuente con un avance, este sólo podrá desactivar la cuenta.

F2 Gestión de avance: Cada usuario puede acceder a la información de avance, de esta manera se puede apreciar el progreso de cada uno de los ejercicios.

F3 Gestión de ejercicios: Los usuarios pueden consultar y realizar los ejercicios existentes en el sistema. Por cada realización de los mismo, se creará un avance que se va almacenando en la base de datos.

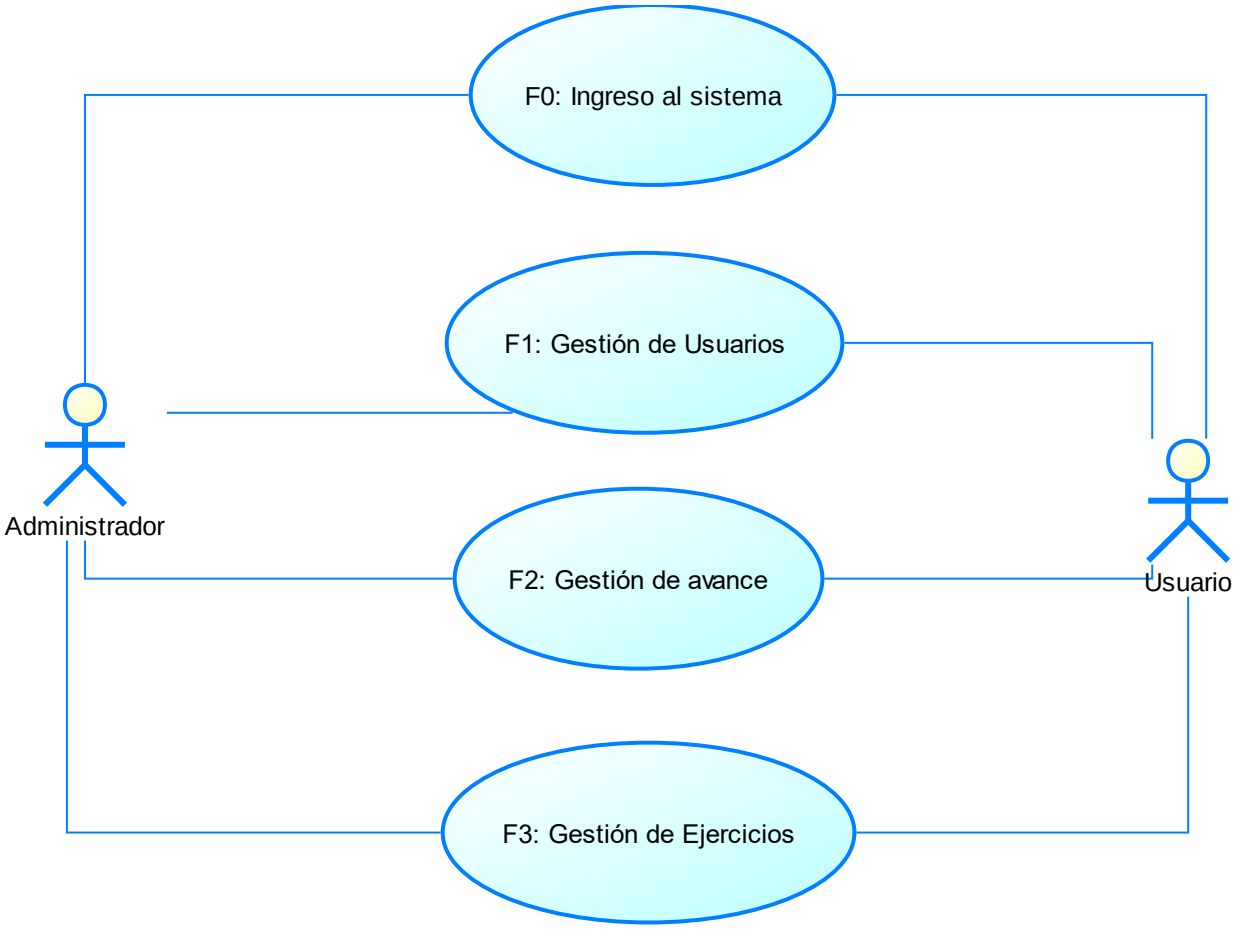


Figura 8. Funciones del sistema.
Elaboración propia.

4.1.3.2.1. F0: Ingreso al sistema

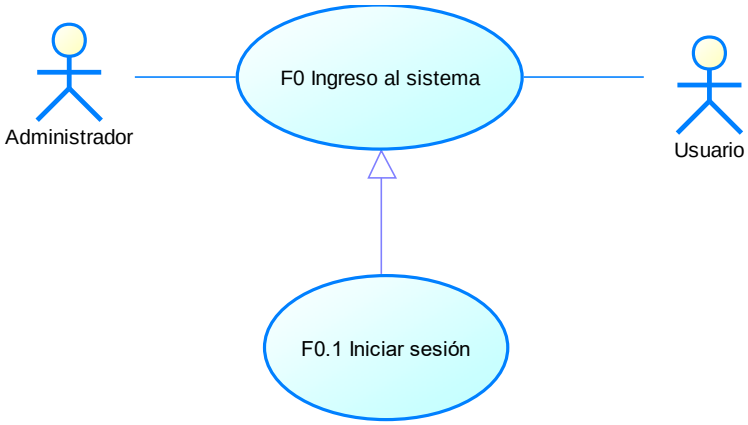


Figura 9. F0 Ingreso al sistema.
Elaboración propia.

A detalle:

F0.1 Iniciar sesión

Descripción: Permite al usuario autenticar sus credenciales en el sistema para tener acceso al mismo.



Figura 10. F0.1 Iniciar sesión.
Elaboración propia.

Actor: Usuario, Administrador

Flujo principal:

1. El sistema presenta pantalla principal
2. El actor presiona Iniciar Sesión
3. El sistema muestra pantalla de iniciar sesión
4. El actor ingresa correo electrónico
5. El actor ingresa contraseña

6. El actor presiona el botón “Ingresar”
7. El sistema verifica que el usuario y contraseña sean correctos (E1, E2)
8. El sistema da acceso a la aplicación web

Flujo alterno:

8. Ver caso de uso F1.1

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA
E2: Datos incorrectos	Usuario y/o contraseña incorrectos, ingrese nuevamente.

Tabla 5. Excepciones F0.1.
Elaboración propia.

4.1.3.2.2. F1: Gestión de usuarios

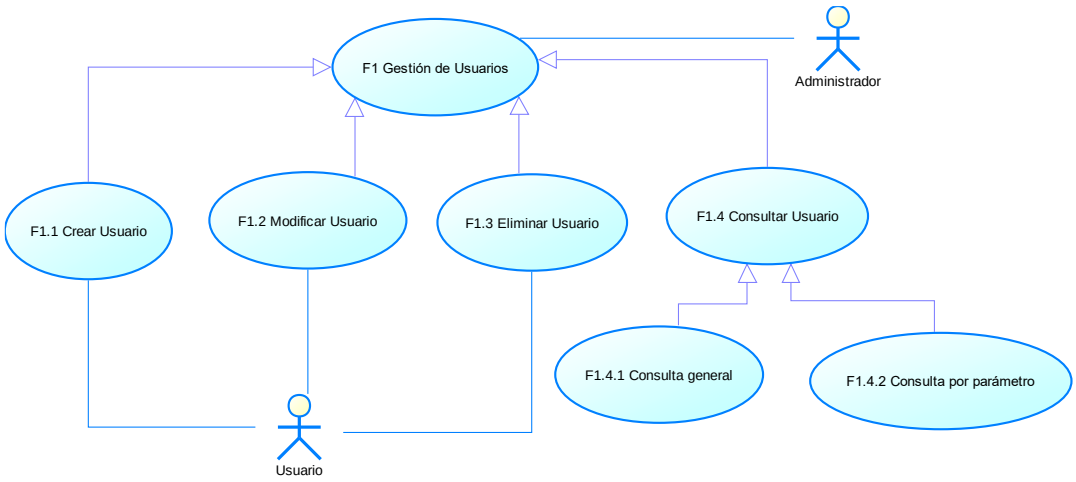


Figura 11. F1 Gestión de usuarios.
Elaboración propia.

A detalle:

F1.1 Crear Usuario

Descripción: Permite ingresar un nuevo usuario al sistema

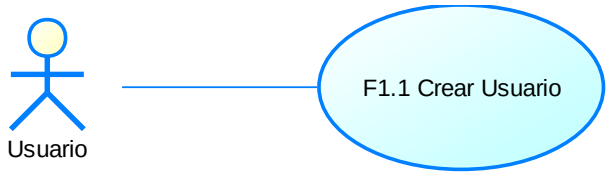


Figura 12. F1.1 Crear Usuario.
Elaboración propia.

Actor: Usuario

Flujo principal:

1. El actor presiona el botón “Registrarse” en el menú principal.
2. El sistema muestra en pantalla el formulario a llenar.
3. El actor ingresa:
    a. Nombre de usuario
    b. Correo electrónico (E2)
    c. Fecha de nacimiento (E2)
    d. Nivel
    e. Contraseña
    f. Confirmación de contraseña
4. El actor presiona el botón “Registrarse”
5. El sistema verifica existencia de usuario (E1)
6. El sistema almacena los datos del usuario (E1)

Flujo alterno:

6. Ver caso de uso F1.2 o F1.3

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA
E2: Problemas en la validación de tipo de dato	Ingrese los datos correctamente

Tabla 6. Excepciones F1.1.
Elaboración propia.

F1.2 Modificar Usuario

Descripción: Permite modificar un usuario que existe en el sistema

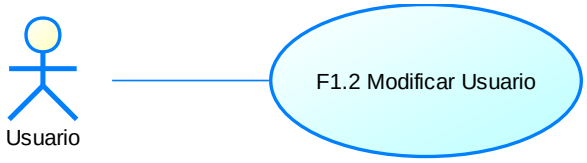


Figura 13. F1.2 Modificar Usuario.
Elaboración propia.

Actor: Usuario

Flujo principal

1. El actor presiona el botón “Perfil” en el menú principal
2. El sistema muestra en pantalla la vista del perfil de usuario
3. El sistema carga los datos del usuario (E1)
4. El actor presiona el botón “Editar”
5. El sistema muestra vista de editar perfil
6. El actor modifica los datos:
    a. Nombre de usuario
    b. Correo electrónico (E2)
    c. Fecha de nacimiento (E2)
    d. Nivel
    e. Contraseña
7. El actor presiona el botón “Guardar Cambios” (E1)
8. El sistema almacena la información

Flujo alterno:

8. Ver caso de uso F1.1

Excepciones:

Causa	**Mensaje**
E1: Problemas con BBDD	Comunicarse con su DBA
E2: Problemas en la validación de tipo de dato	Ingrese los datos correctamente

Tabla 7. Excepciones F1.2.
Elaboración propia.

F1.3 Eliminar Usuario

Descripción: Permite eliminar un usuario que existe en el sistema

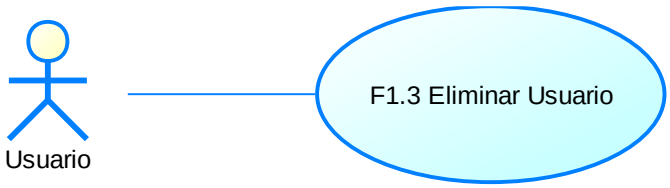


Figura 14. F1.3 Eliminar Usuario.
Elaboración propia.

Actor: Usuario

Flujo principal:

1. El actor presiona el botón “Perfil” del menú principal
2. El sistema muestra la vista del perfil de usuario
3. El sistema carga la información del usuario (E1)
4. El actor presiona el botón de “Eliminar” (E2)
5. El sistema elimina la información del usuario

Flujo alterno:

5. Ver caso de uso F1.1

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA
E2: Asegurarse de eliminar datos	¿Está seguro de que desea eliminar?

Tabla 8. Excepciones F1.3.
Elaboración propia.

F1.4.1 Consulta General Usuarios

Descripción: Permite realizar una consulta de manera general de todos los usuarios que existen en el sistema.



Figura 15. F1.4.1 Consulta General Usuarios.
Elaboración propia.

Actor: Administrador

Flujo principal:

1. El actor selecciona usuarios en el menú principal
2. El sistema muestra pantalla de Usuarios
3. El actor selecciona “Consulta General”
4. El sistema carga los datos de los usuarios del sistema (E1)
5. El sistema muestra la información (E1)

Excepciones:

Causa	**Mensaje**
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 9. Excepciones F1.4.1.
Elaboración propia.

F1.4.2 Consulta de Usuarios por parámetro

Descripción: Permite realizar una consulta por parámetro de usuarios que existen en el sistema.

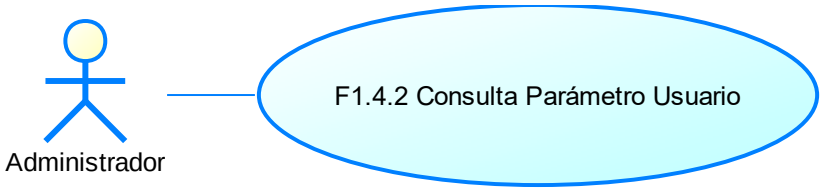


Figura 16. F1.4.2 Consulta Usuarios por parámetro.
Elaboración propia.

Actor: Administrador

Flujo principal

1. El actor selecciona usuarios en el menú principal
2. El sistema muestra la pantalla de Usuarios
3. El actor selecciona “Consulta por parámetro”
4. El actor selecciona el parámetro por buscar
5. El actor ingresa el valor del parámetro
6. El actor presiona “Buscar”
7. El sistema carga la información (E1)
8. El sistema muestra la información requerida

Excepciones:

Causa	**Mensaje**
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 10. Excepciones F1.4.2.
Elaboración propia.

4.1.3.2.3. F2 Gestión de avance

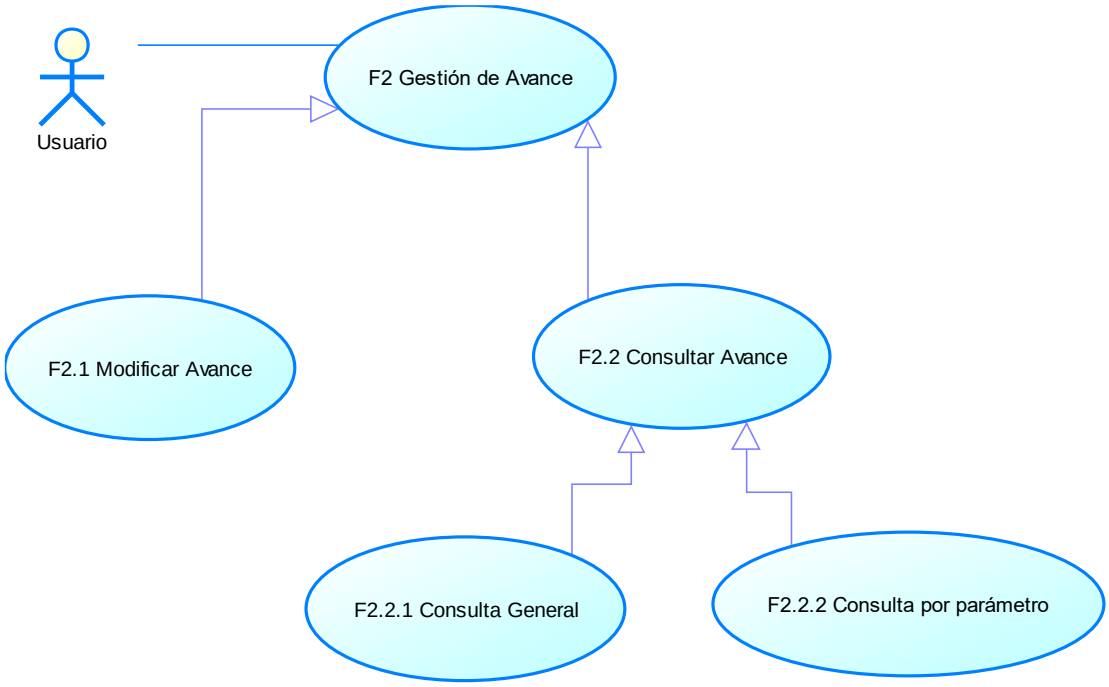


Figura 17. F2 Gestión de avance.
Elaboración propia.

A detalle:

F2.1 Modificar Avance

Descripción: Permite que el sistema actualice el avance de cada usuario cuando este realiza algún ejercicio.

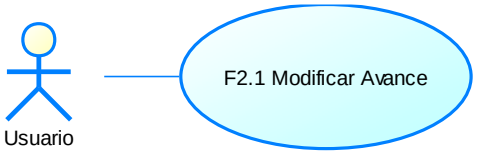


Figura 18. F2.1 Modificar Avance.
Elaboración propia.

Actor: Usuario

Flujo principal:

1. El actor ingresa a un ejercicio
2. El actor realiza el ejercicio (ver caso de uso F3.2)
3. El sistema calcula los resultados obtenidos
4. El sistema actualiza avance del usuario en la base de datos (E1)

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 11. Excepciones F2.1.
Elaboración propia.

F2.2.1 Consulta General de Avance

Descripción: Permite que el usuario consulte el avance de manera general.

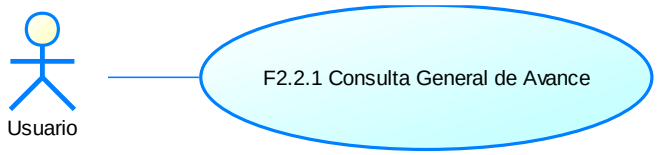


Figura 19. F2.3.1 Consulta General Avance.
Elaboración propia.

Actor: Usuario

Flujo principal:

1. El actor selecciona “Perfil” en el menú principal
2. El sistema muestra pantalla del perfil
3. El actor presiona “Mi avance”
4. El sistema muestra pantalla de avance (E1)
5. El sistema muestra datos del avance en una tabla de manera general (E1)

Flujo alterno:

4. Ver caso de uso F3.1

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 12. Excepciones F2.3.1.
Elaboración propia.

F2.2.2 Consulta por parámetro de Avance

Descripción: Permite que el usuario y administrador consulten el avance por parámetro.

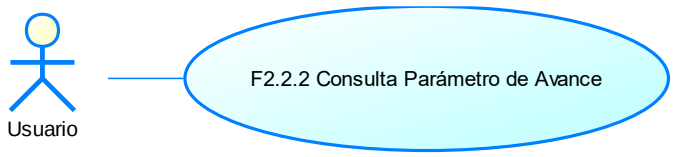


Figura 20. F2.3.2 Consulta por parámetro Avance.
Elaboración propia.

Actor: Usuario, Administrador

Flujo principal:

1. El actor selecciona “Perfil” en el menú principal
2. El sistema muestra pantalla del perfil
3. El actor presiona “Mi avance”
4. El sistema muestra pantalla de avance (E1)
5. El usuario selecciona el parámetro a buscar, por ejercicio o categoría (E2)
6. El sistema muestra datos del avance por ejercicio o categoría(E1)

Flujo alterno:

6.Ver caso de uso F3.1

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA
E2: Tipo de dato incorrecto	Ingrese el dato correctamente

Tabla 13. Excepciones F2.3.2.
Elaboración propia.

4.1.3.2.4. F3 Gestión de Ejercicios

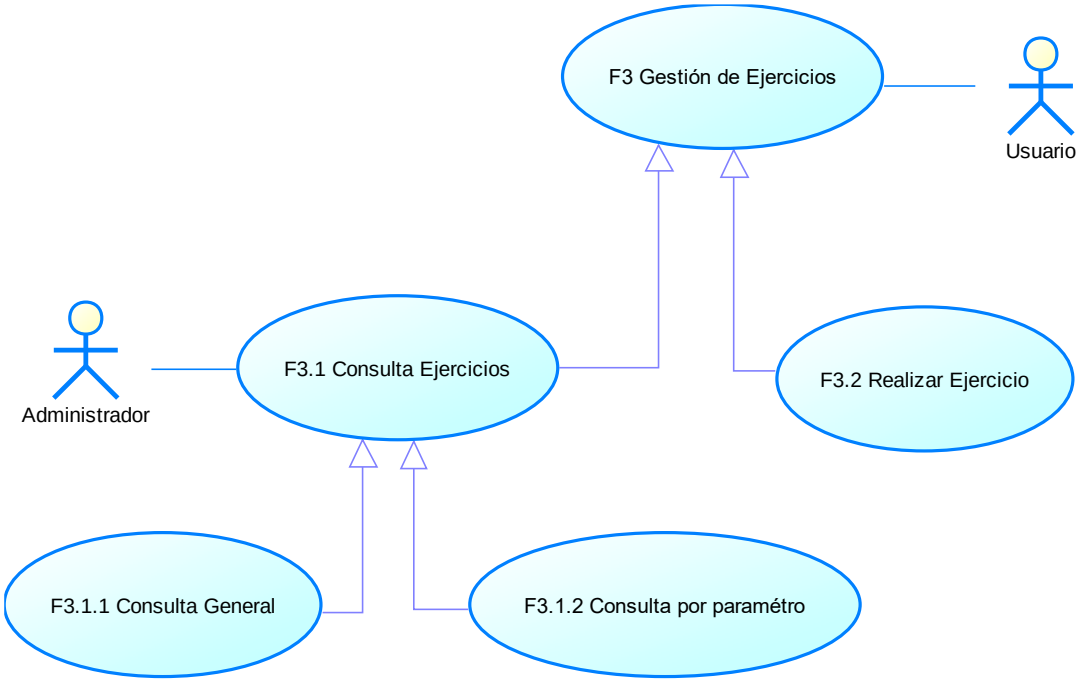


Figura 21. F3 Gestión de Ejercicios.
Elaboración propia.

A detalle:

F3.1.1 Consulta General de Ejercicios

Descripción: Permite realizar una consulta general de los ejercicios existentes en el sistema.

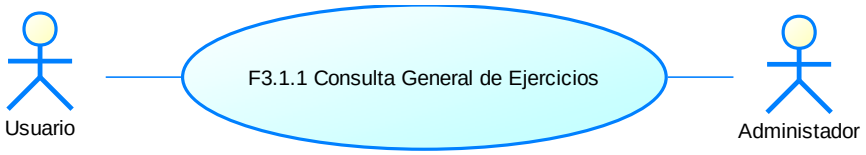


Figura 22. F3.1.1 Consulta General Ejercicios.
Elaboración propia.

Actor: usuario, administrador

Flujo principal:

1. El usuario presiona "Ejercicios" en el menú principal
2. El sistema muestra pantalla de ejercicios
3. El sistema carga información (E1)
4. El sistema muestra los ejercicios existentes en el sistema

Excepciones:

Causa	**Mensaje**
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 14. Excepciones F3.1.1.
Elaboración propia.

F3.1.2 Consulta por parámetro de Ejercicios

Descripción: Permite realizar una consulta por parámetro de los ejercicios existentes en el sistema.

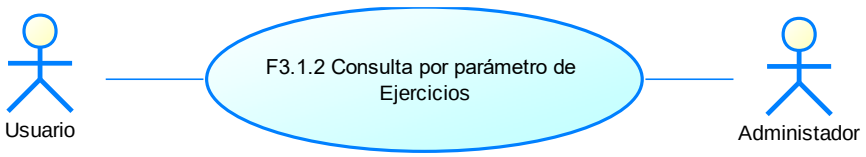


Figura 23. F3.1.2 Consulta por parámetro Ejercicios.
Elaboración propia.

Actor: usuario, administrador

Flujo principal:

1. El usuario presiona “Ejercicios” en el menú principal
2. El sistema muestra pantalla de ejercicios
3. El actor ingresa el nombre del ejercicio a buscar
4. El sistema carga información (E1)
5. El sistema muestra los ejercicios existentes en el sistema que coincidan con el parámetro ingresado

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 15. Excepciones F3.1.2.
Elaboración propia.

F3.2 Realizar Ejercicios

Descripción: Permite que el usuario realice cualquier ejercicio existente en el sistema.

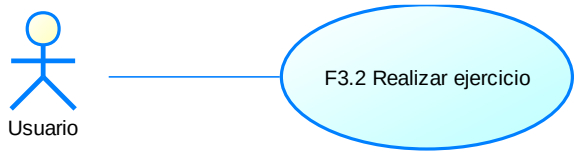


Figura 24. F3.2 Realizar Ejercicios.
Elaboración propia.

Actor: usuario

Flujo principal:

1. El usuario presiona “Ejercicios” en el menú principal
2. El sistema muestra pantalla de ejercicios
3. El actor presiona sobre el ejercicio que desea realizar
4. El sistema muestra pantalla del ejercicio seleccionado (E1)
5. El sistema muestra las indicaciones del ejercicio
6. El actor realiza el ejercicio

7. El sistema toma tiempo, fecha e intentos de realización del ejercicio resuelto.
8. El sistema muestra mensaje de resultado
9. El sistema almacena informacion (Ver caso de uso 2.1)

Excepciones:

Causa	Mensaje
E1: Problemas con BBDD	Comunicarse con su DBA

Tabla 16. Excepciones F3.2.
Elaboración propia.

4.1.3.3. Requisitos de Rendimiento

Descripción	Prioridad
El 90% de las solicitudes deben realizarse en menos de 5 segundos.	Media
El número de transacciones por segundo simultáneamente es de 50.	Media
Conexión a la red por medio de cable o Wifi.	Alta

Tabla 17. Requisitos de rendimiento.
Elaboración propia.

4.1.3.4. Requisitos de Diseño

Descripción	Prioridad
Contar con la última versión estable de un navegador: • Google Chrome: 86.0.4240.183 • Firefox: 82.0.2 • Microsoft Edge: 86.0.622.38 • Opera: 71.0.3770.228	Alta
Utilizar colores llamativos para botones e imágenes.	Alta

Utilizar tamaño de fuente grande para cualquier texto de la aplicación.	Alta
Aplicación *responsive*[10]	Alta

Tabla 18. Requisitos de diseño.
Elaboración propia

4.1.3.5. Atributos del sistema

Descripción	Prioridad
Existirán dos tipos de roles: Usuario y Administrador. Cada uno cuenta con diferentes características y acceso a funcionalidades de la aplicación.	Alta
La autentificación es por medio de un *login* con usuario y contraseña, con el cual se identificará el rol del usuario.	Alta
La contraseña de los usuarios debe ser mínimo de ocho caracteres, estas serán encriptadas antes de su almacenamiento en la base de datos.	Alta

Tabla 19. Atributos del sistema.
Elaboración propia.

4.1.3.6. Otros requisitos

Descripción	Prioridad
Utilización de Angular para el desarrollo del frontend de la aplicación.	Alta
Utilización de Laravel para el desarrollo del backend de la aplicación.	Alta
Utilización del gestor de base de datos MySQL para el almacenamiento de información.	Alta

[10] Una aplicación responsive es aquella que se adapta a cualquier dispositivo y se visualiza de manera correcta en cualquier tamaño de pantalla.

Manejo de la metodología Scrum para el desarrollo de la aplicación web.	Alta

Tabla 20. Otros requisitos.
Elaboración propia.

4.1.4. Apéndices

4.1.4.1. Diagrama Conceptual

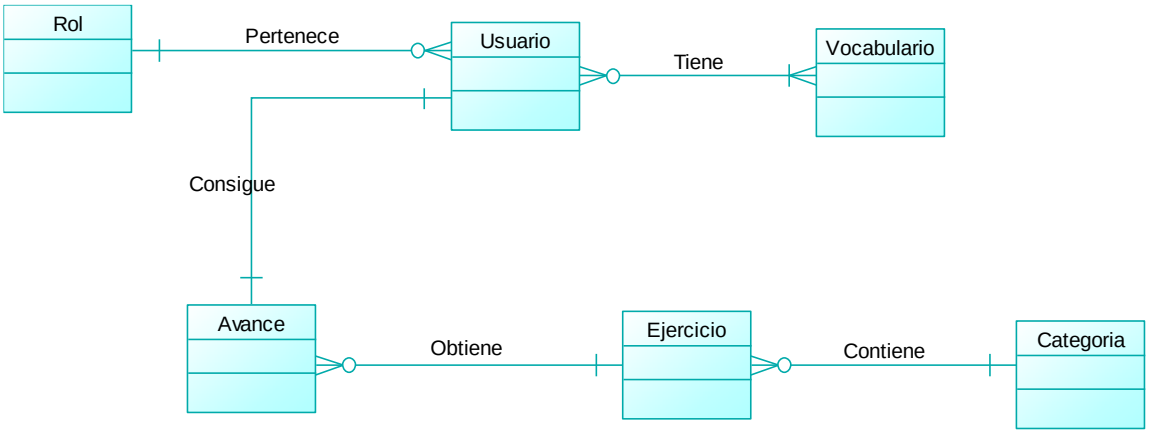


Figura 25. Diagrama Conceptual.
Elaboración propia.

4.2. Diseño

En la fase de diseño se indica la manera en la que la aplicación interactúa con el usuario. Para el desarrollo de esta aplicación web se utilizará el modelo de vista controlador, el cual será detallado en los siguientes puntos:

4.2.1. Diseño Arquitectónico

Para el desarrollo de la aplicación se utilizará el Modelo Vista-Controlador, el cual es uno de los patrones en el diseño de software más utilizados ya que se encarga de dividir la lógica del negocio con la vista, mejorando el mantenimiento de la aplicación.

Este patrón cuenta con tres componentes: Modelo, Vista y Controlador. Cada uno de ellos cumple una función diferente y permite que el desarrollo de aplicaciones sea más optimo,

además de que estos componentes separan los datos que maneja la aplicación, la lógica de control y la interfaz (Universidad de Alicante, 2020).

4.2.1.1. Modelo

Es un conjunto de objetos que se encargan de realizar consultas a la base de datos y contienen la lógica de negocio. A parte de realizar consultas, el modelo valida las actualizaciones y acceso a diferentes funcionalidades dependiendo del rol con el que se está trabajando. Una vez que ha procesado la información, la envía a la Vista.

4.2.1.2. Vista

Es la representación gráfica de la información, es decir es la interfaz de usuario. En este componente se encuentran de manera organizada los datos que el usuario recibe, y también es la parte por la cual el usuario ingresa información.

4.2.1.3. Controlador

Es aquel que escucha las peticiones del usuario y le informa al Modelo. Cada controlador cuenta con diferentes métodos para responder a las diferentes solicitudes que son recibidas desde la Vista y pedir la información necesaria al Modelo.

4.2.2. Diagrama de Clases

4.2.2.1. Diagrama de clases de vistas

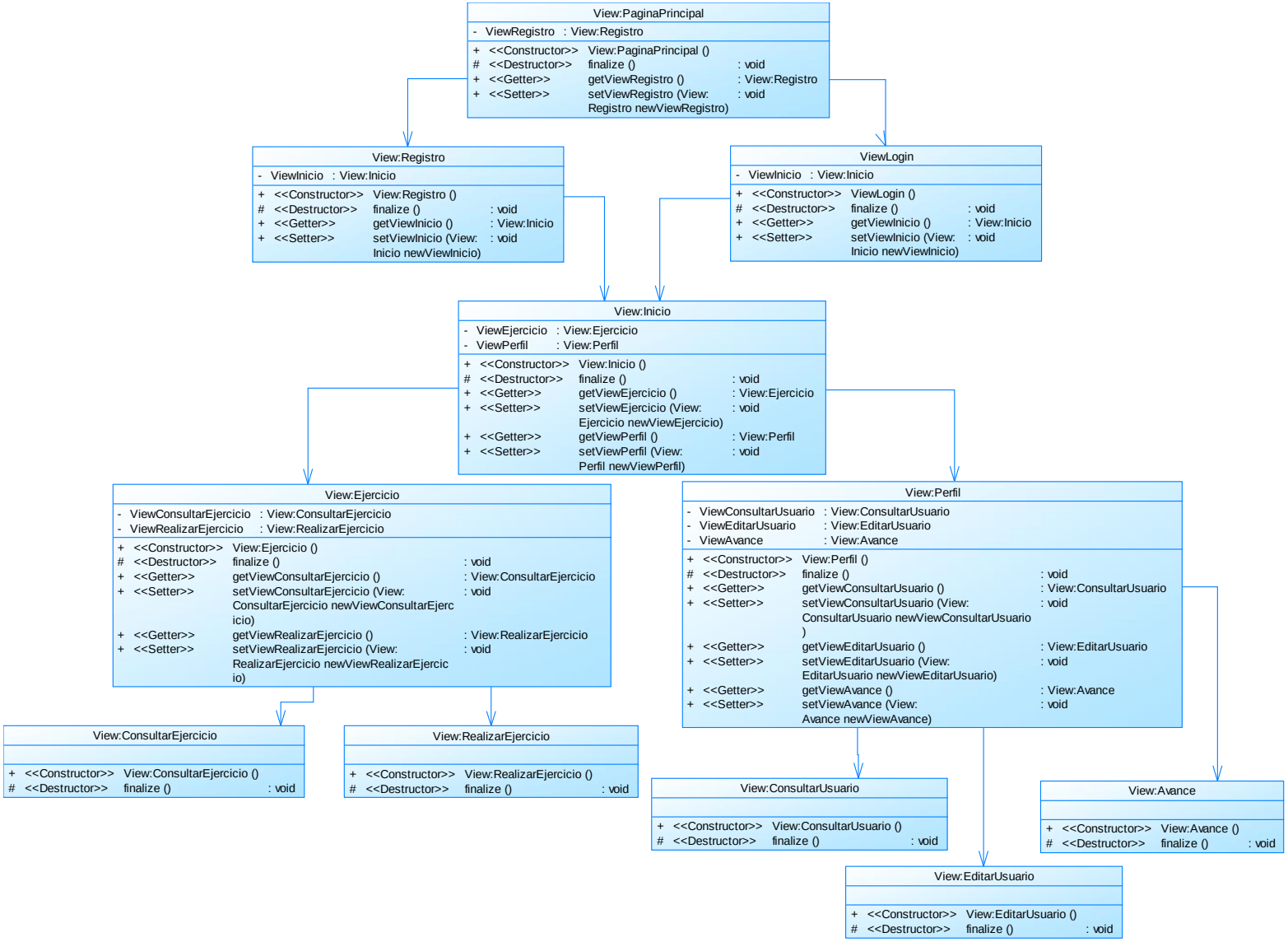


Figura 26. Diagrama de clases de vistas.
Elaboración propia.

4.2.2.2. Diagrama de clases de controladores

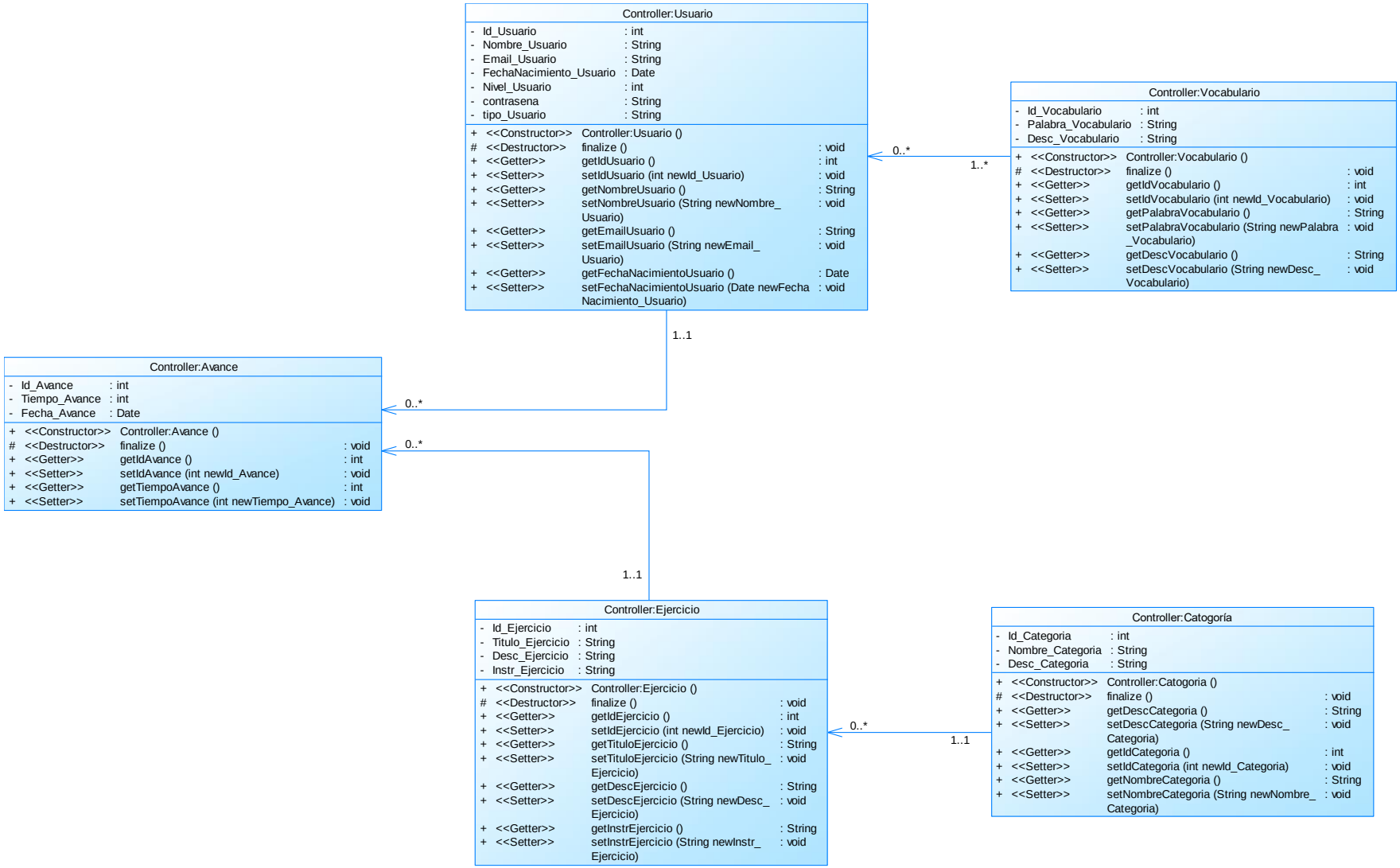


Figura 27. Diagrama de Clases de Controladores.
Elaboración propia.

4.2.2.3. Diagrama de entidad-relación

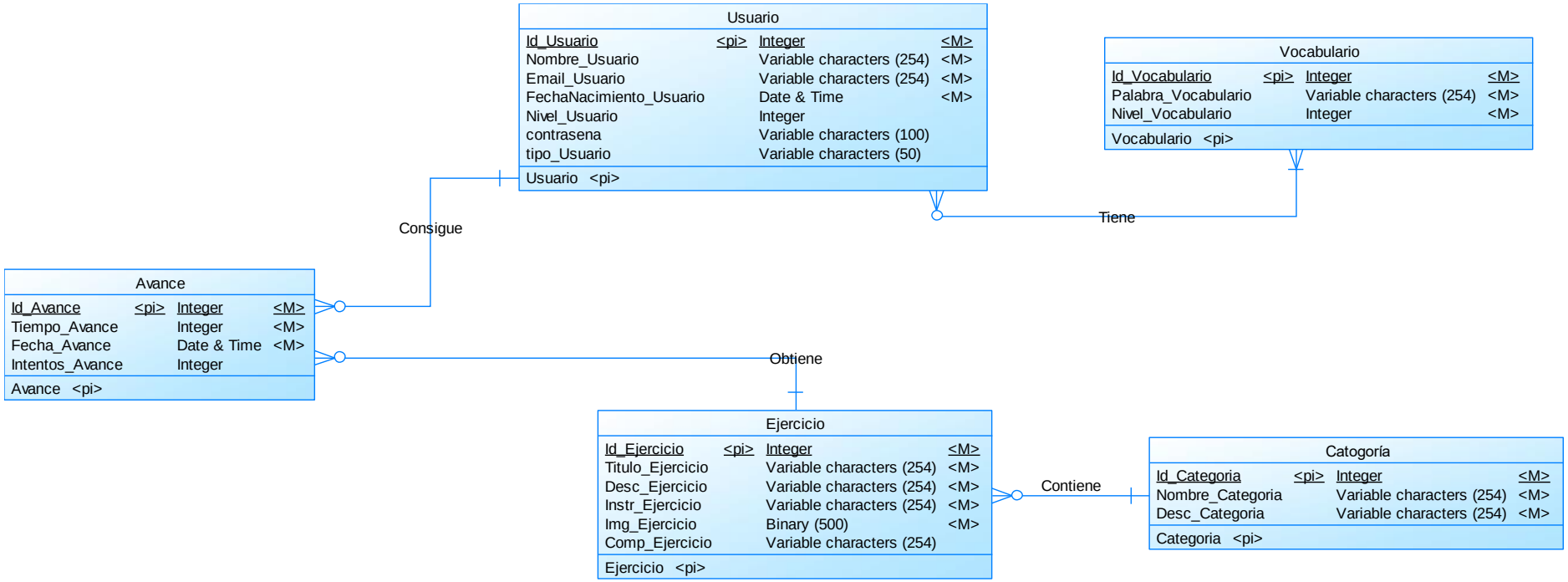


Figura 28. Diagrama de entidad-relación
Elaboración propia.

4.2.2.4. Diagrama de clases a nivel lógico

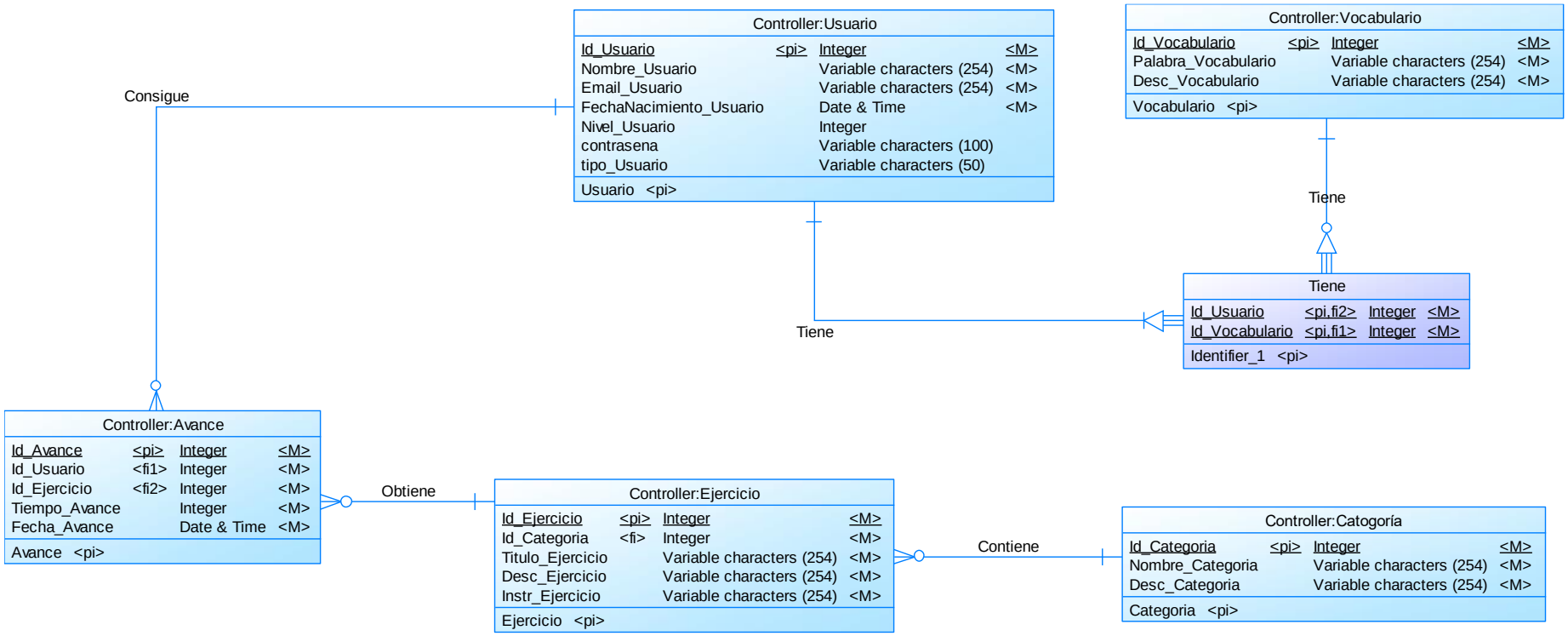


Figura 29. Diagrama a nivel lógico.
Elaboración propia.

4.2.2.5. Diagrama de clases a nivel físico

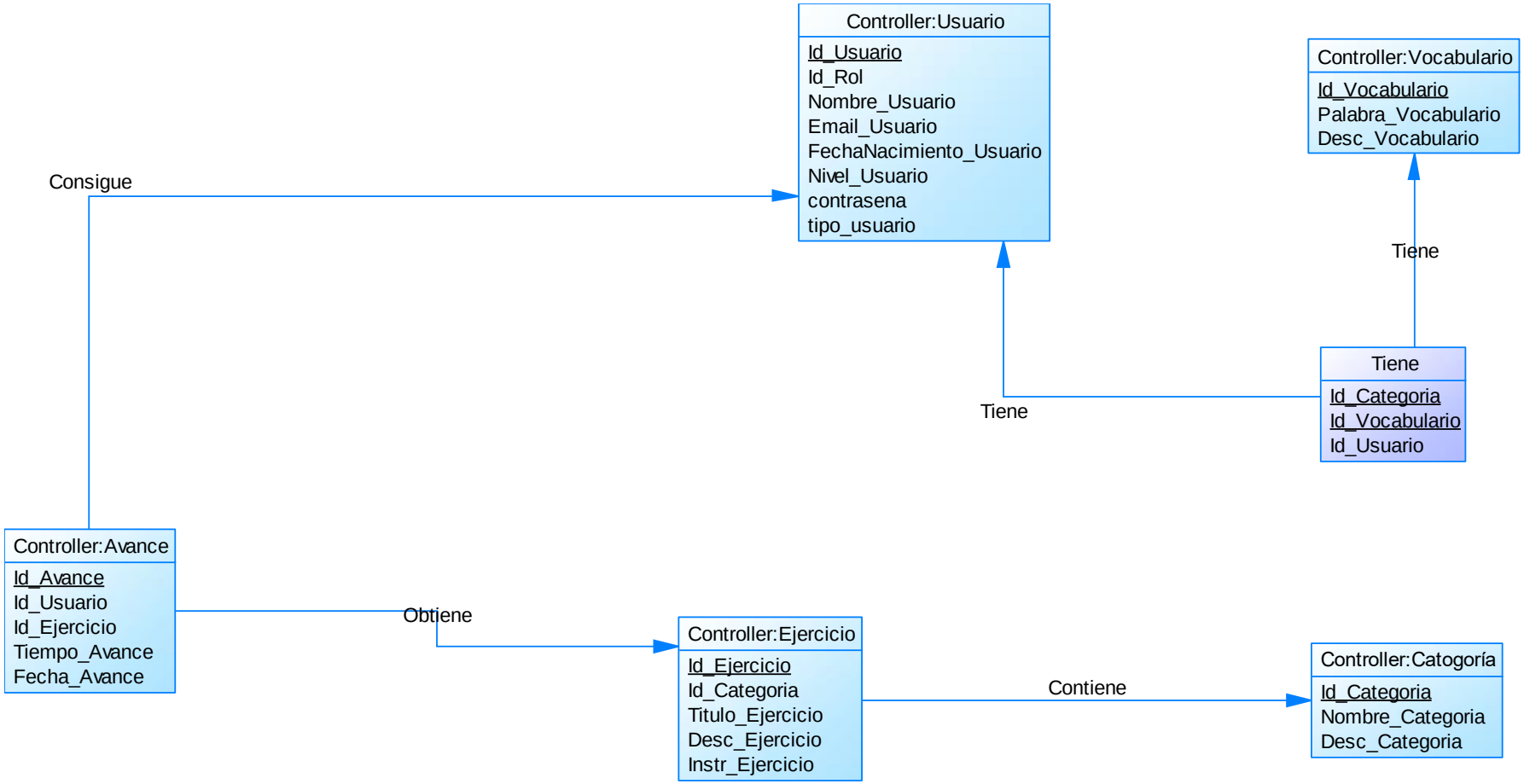


Figura 30. Diagrama a nivel físico.
Elaboración propia

4.2.3. Diccionario de Datos

1. **Lista de entidades**

Name	*Code*	*Parent*	*Generate*
Avance	Avance	Conceptual Data Model 'Clases'	X
Categoría	Categoria	Conceptual Data Model 'Clases'	X
Ejercicio	Ejercicio	Conceptual Data Model 'Clases'	X
Usuario	Usuario	Conceptual Data Model 'Clases'	X
Vocabulario	Vocabulario	Conceptual Data Model 'Clases'	X

1.1 Lista de datos de entidades

Name	*Code*	*Data Type*	*Length*
Comp_Ejercicio	Comp_Ejercicio	Variable characters(254)	254
contrasena	contrasena	Variable characters(100)	100
Desc_Categoria	descCategoria	Variable characters(254)	254
Desc_Ejercicio	descEjercicio	Variable characters(254)	254
Email_Usuario	emailUsuario	Variable characters(254)	254
Fecha_Avance	fechaAvance	Date & Time	
FechaNacimiento_Usuario	fechaNacimientoUsuario	Date & Time	
Id_Avance	idAvance	Integer	
Id_Categoria	idCategoria	Integer	
Id_Ejercicio	idEjercicio	Integer	
Id_Usuario	idUsuario	Integer	
Id_Vocabulario	idVocabulario	Integer	
Img_Ejercicio	Img_Ejercicio	Binary (500)	500

Name	*Code*	*Data Type*	*Length*
Instr_Ejercicio	instrEjercicio	Variable characters (254)	254
Intentos_Avance	Intentos_Avance	Integer	
Nivel_Usuario	Nivel_Usuario	Integer	
Nivel_Vocabulario	descVocabulario	Integer	
Nombre_Categoria	nombreCategoria	Variable characters (254)	254
Nombre_Usuario	nombreUsuario	Variable characters (254)	254
Palabra_Vocabulario	palabraVocabulario	Variable characters (254)	254
Tiempo_Avance	tiempoAvance	Integer	
tipo_Usuario	tipo_Usuario	Variable characters(50)	50
Titulo_Ejercicio	tituloEjercicio	Variable characters (254)	254

Nota: No existe dominios ni precisión para los datos.

1.2 Lista de identificadores de identidades

Name	*Code*	
Usuario	USUARIO	Entity 'Usuario'
Ejercicio	EJERCICIO	Entity 'Ejercicio'
Categoria	CATEGORIA	Entity 'Catogoría'
Vocabulario	VOCABULARIO	Entity 'Vocabulario'
Avance	AVANCE	Entity 'Avance'

1.3 Lista de relaciones

Name	*Code*	*Parent*	*Entity 2*	*Entity 1*
Consigue	consigue	Conceptual DataModel 'Clases'	Avance	Usuario

Contiene	contiene	Conceptual DataModel 'Clases'	Ejercicio	Catogoría
Obtiene	obtiene	Conceptual DataModel 'Clases'	Avance	Ejercicio

Name	*Code*	*Parent*	*Entity 2*	*Entity 1*
Tiene	tiene	Conceptual DataModel 'Clases'	Usuario	Vocabulario

4.2.4. Diagramas de Interacción

4.2.4.1. F0: Ingreso al sistema

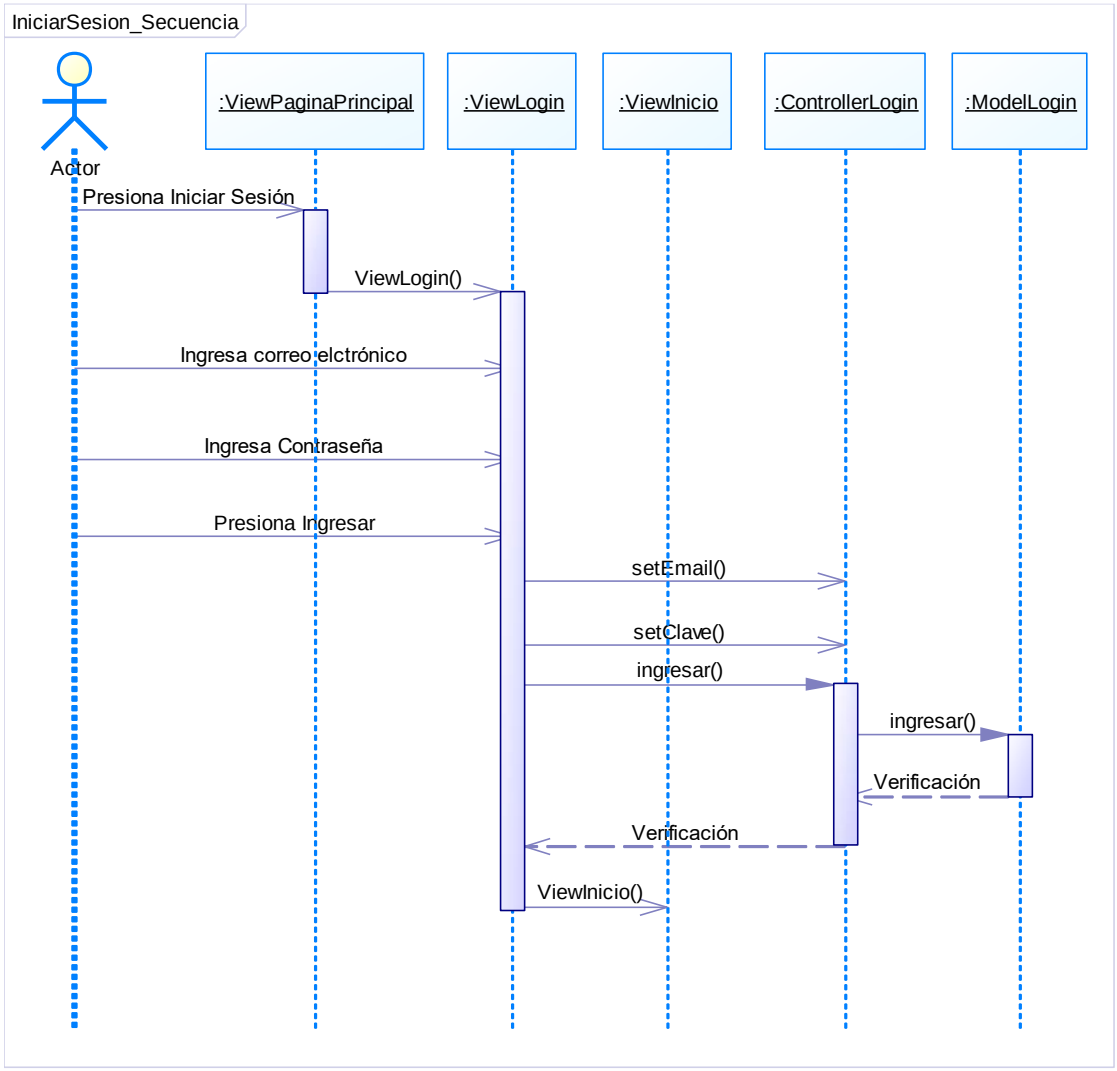


Figura 31. Diagrama de Secuencia F0.1.
Elaboración propia.

4.2.4.2. F1.1 Crear Usuario

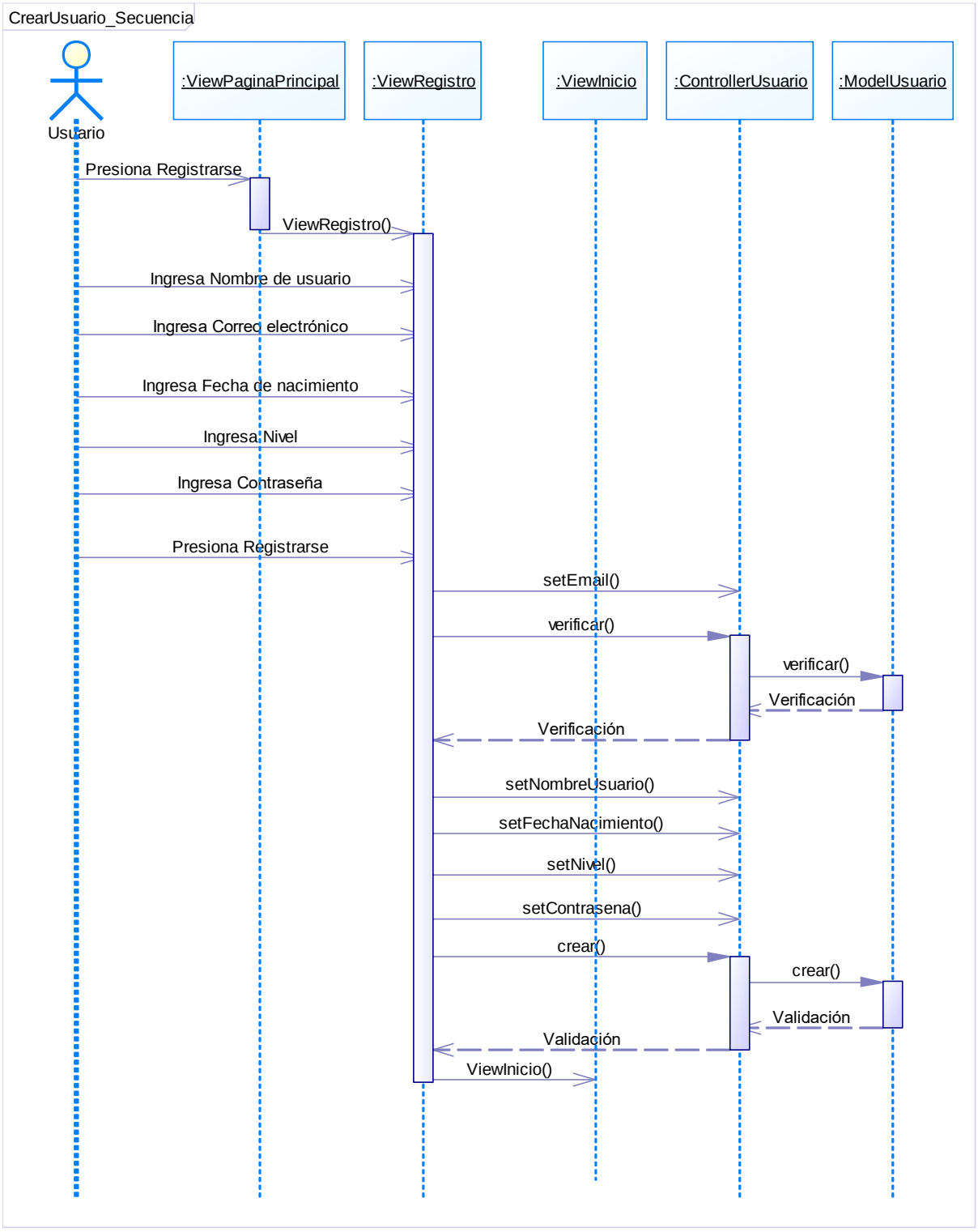


Figura 32. Diagrama de Secuencia F1.1.
Elaboración propia.

4.2.4.3. F1.2 Modificar Usuario

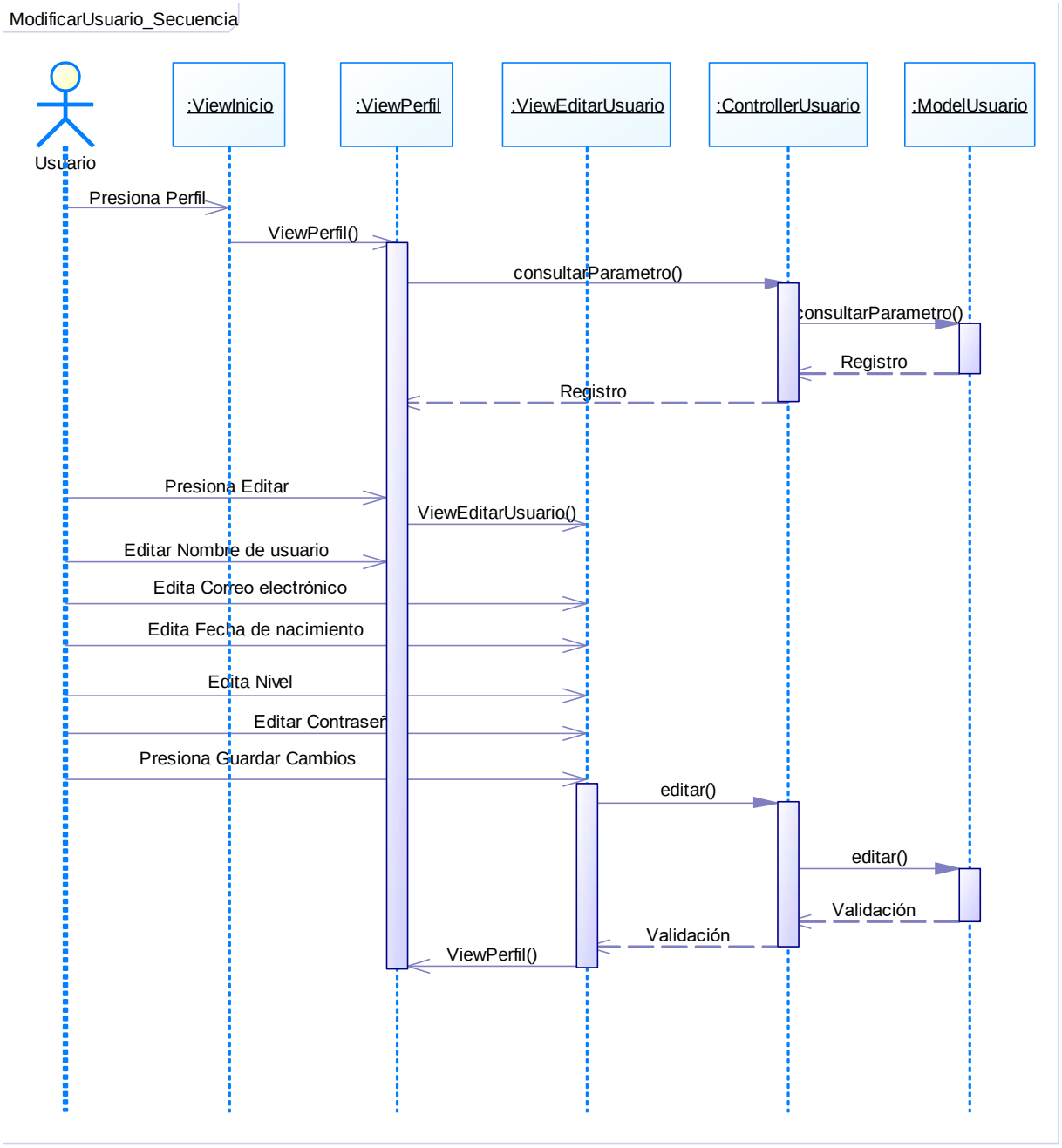


Figura 33. Diagrama de Secuencia F1.2.
Elaboración propia.

4.2.4.4. F1.3 Eliminar Usuario

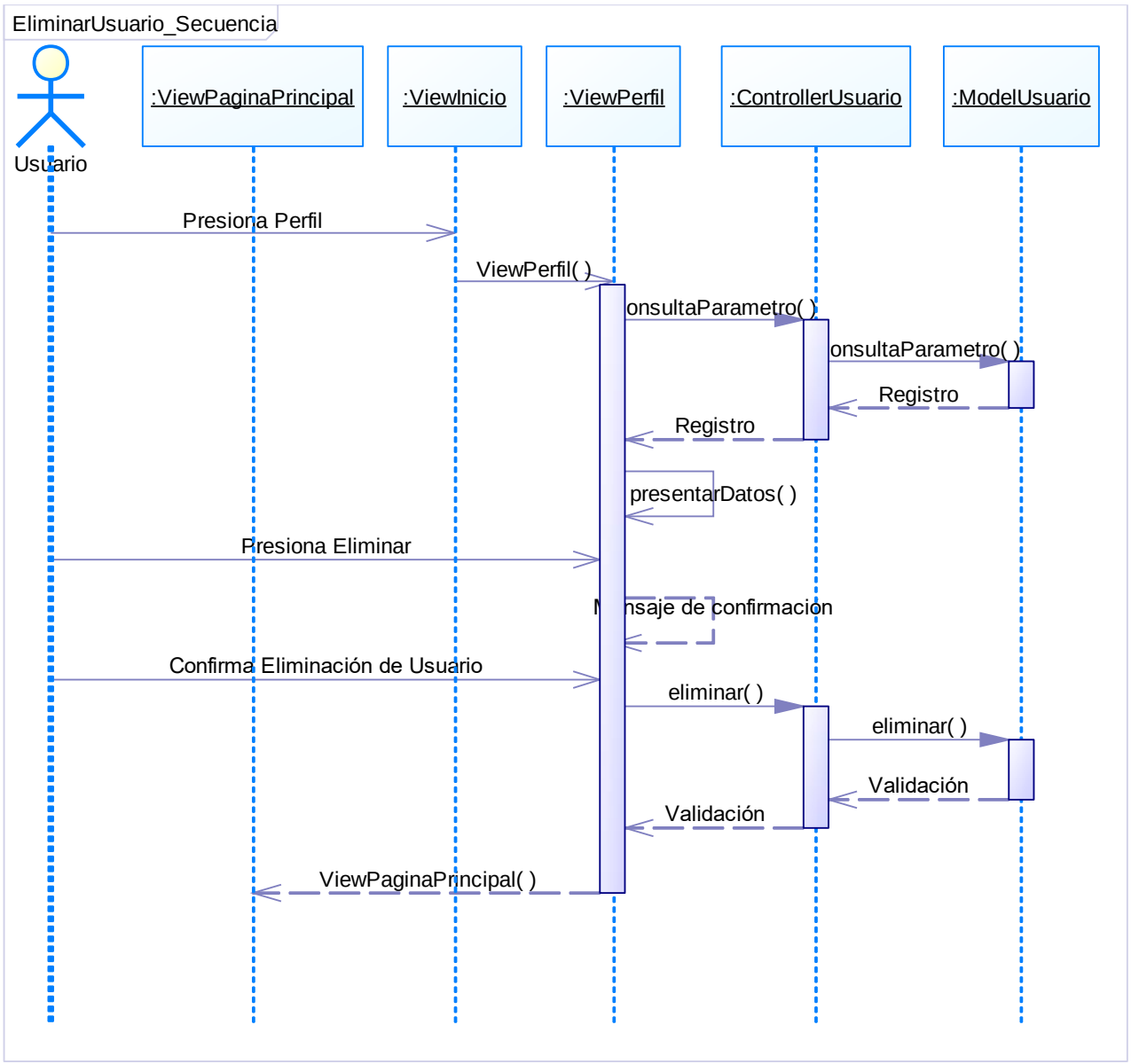


Figura 34. Diagrama de Secuencia F1.3.
Elaboración propia.

4.2.4.5. F1.4.1. Consulta General de Usuarios

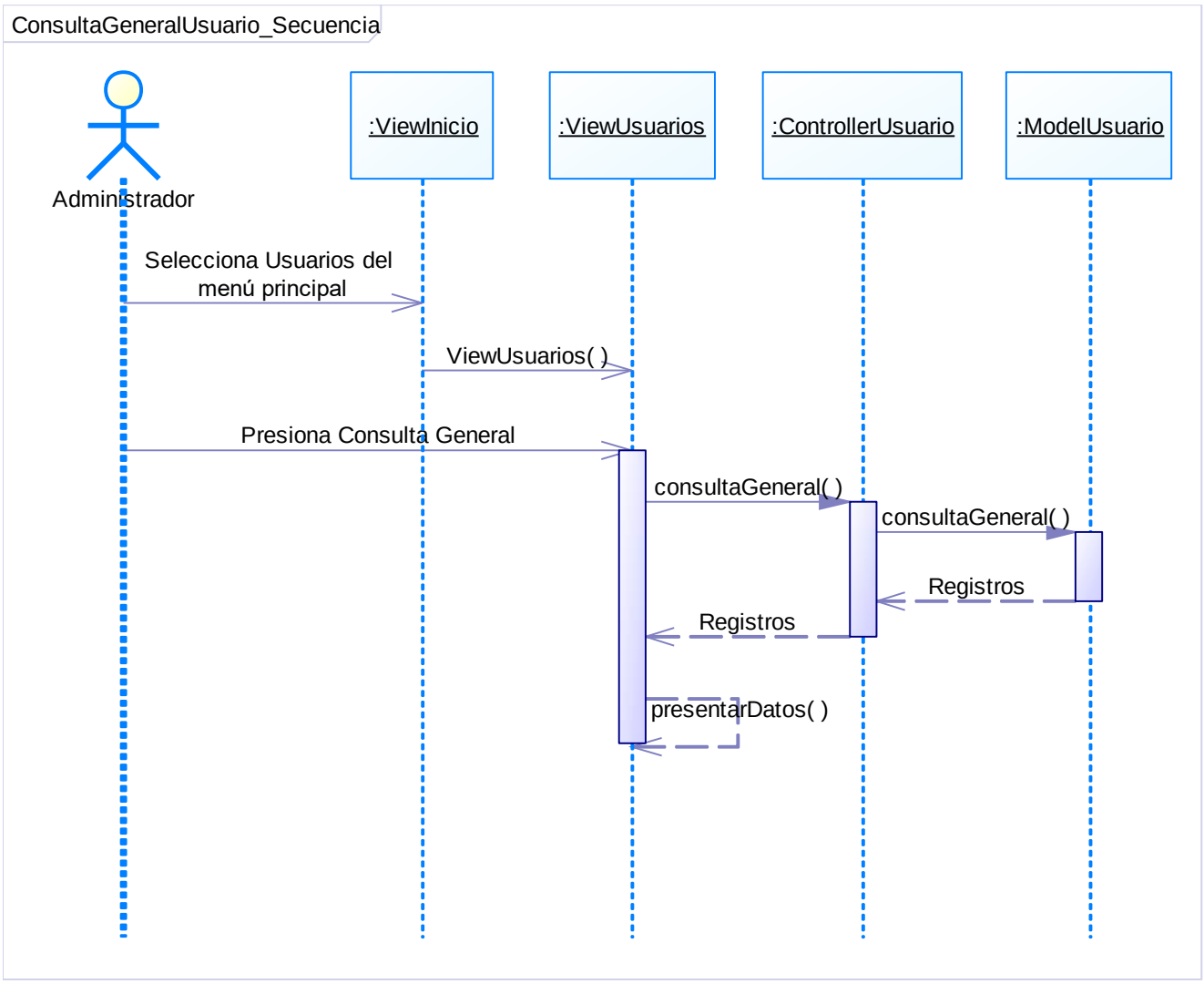


Figura 35. Diagrama de Secuencia F1.4.1.
Elaboración propia.

4.2.4.6. F1.4.2 Consulta por parámetros de Usuarios

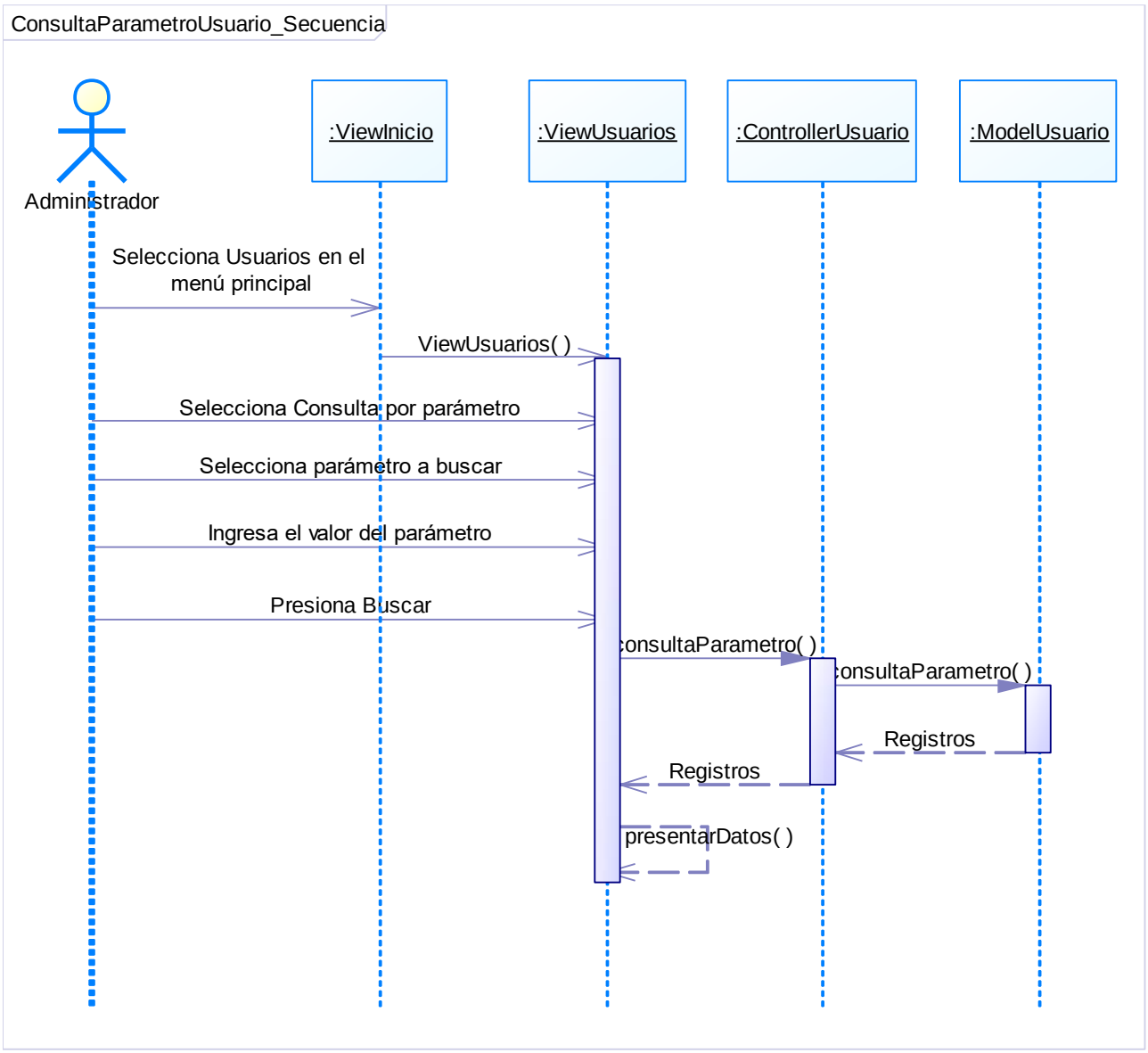


Figura 36. Diagrama de Secuencia F1.4.2.
Elaboración propia.

4.2.4.7. F2.1 Modificar Avance

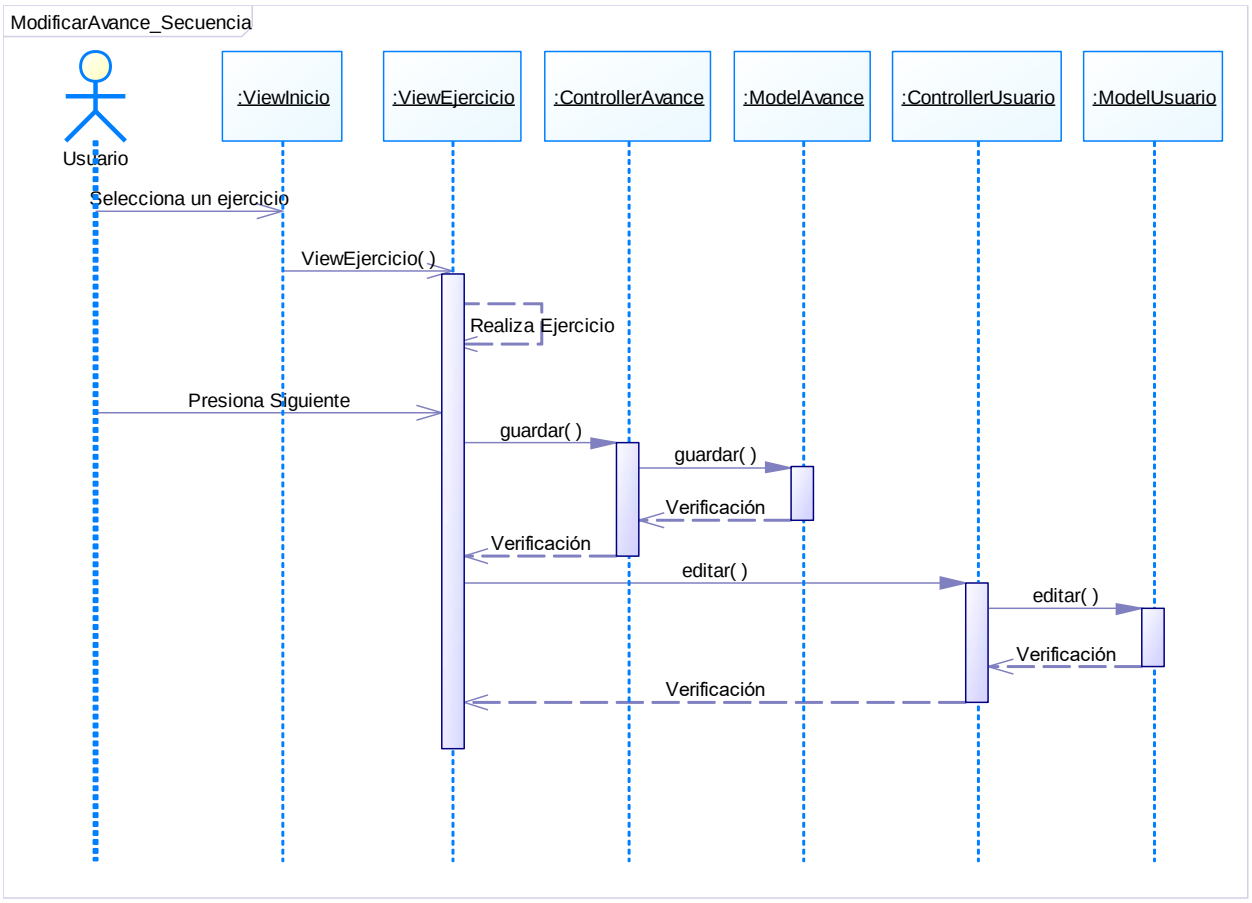


Figura 37. Diagrama de Secuencia F2.1.
Elaboración propia

4.2.4.8. F2.3.1. Consulta General de Avance

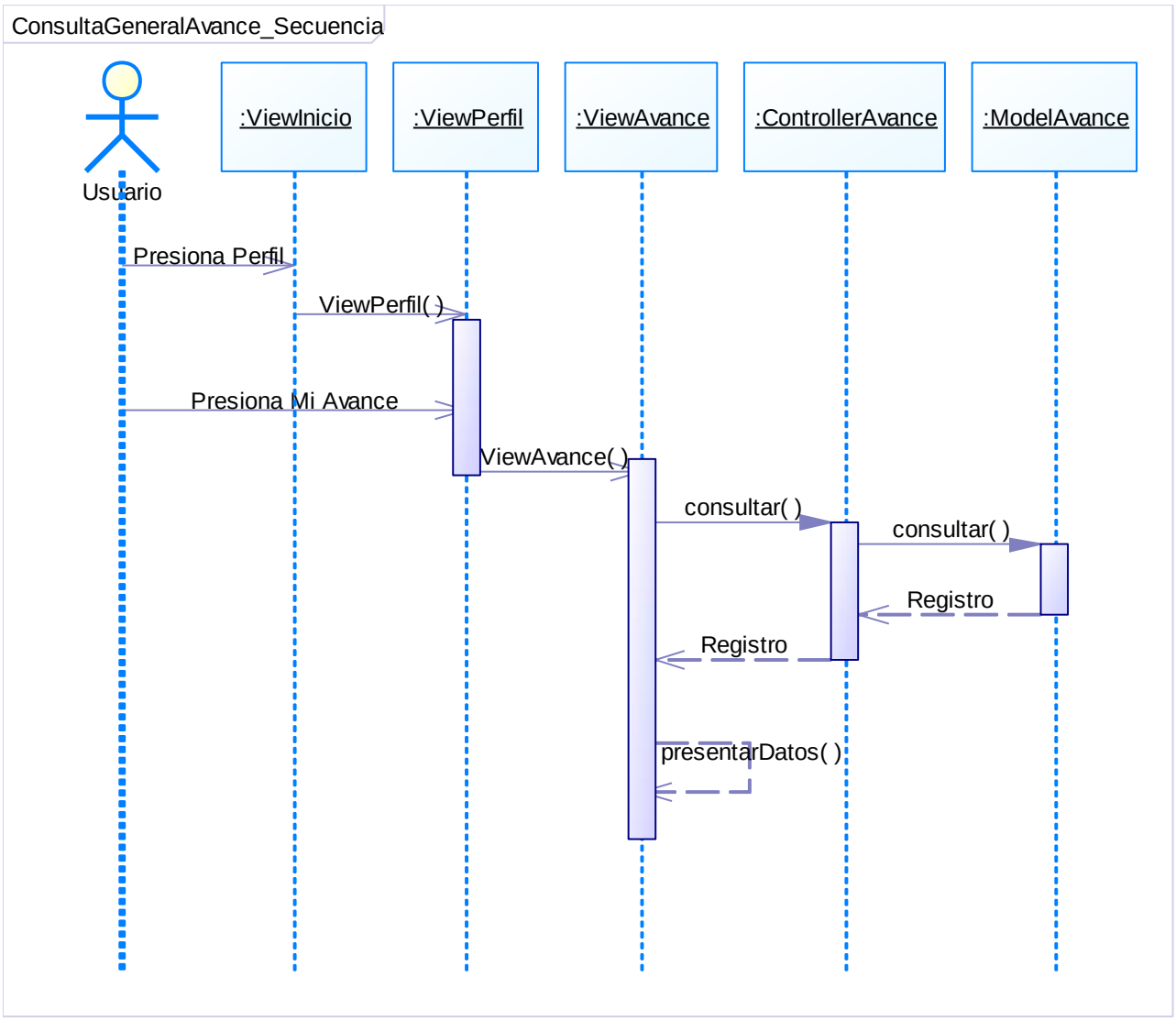


Figura 38. Diagrama de Secuencia F2.3.1.
Elaboración propia.

4.2.4.9. F2.3.2 Consulta por parámetro de Avance

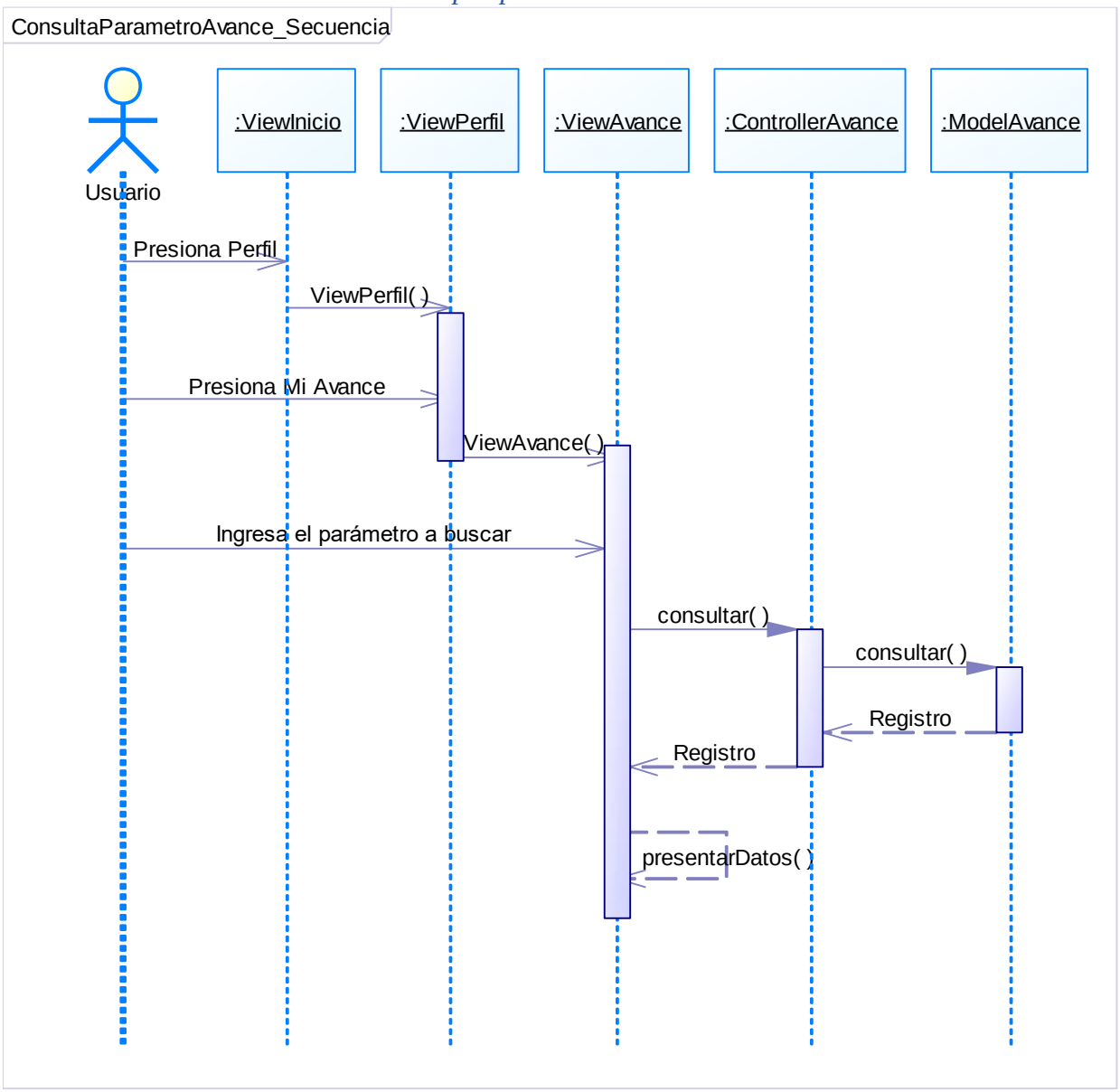


Figura 39. Diagrama de Secuencia F2.3.2.
Elaboración propia.

4.2.4.10. F3.1.1 Consulta General de Ejercicios

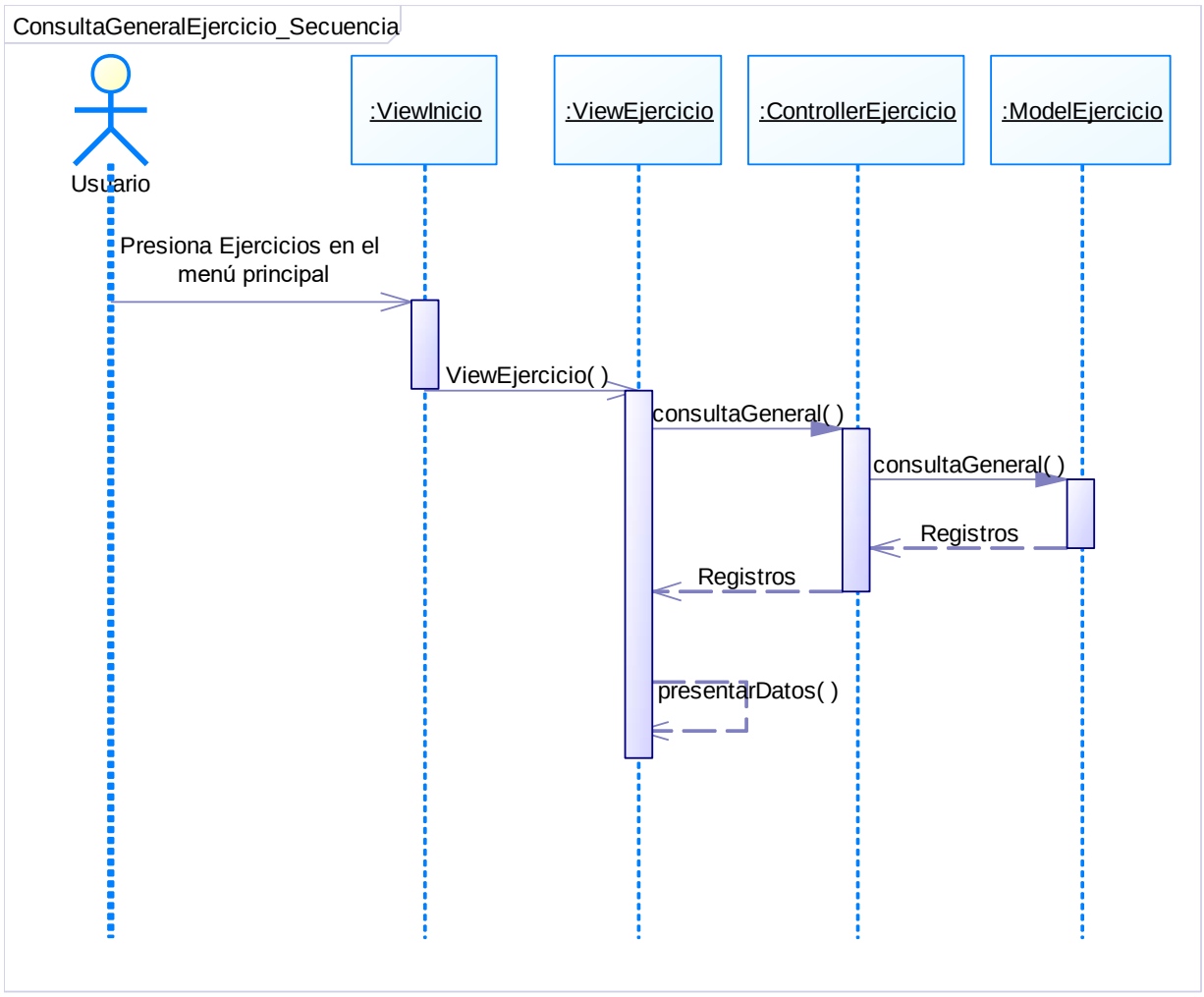


Figura 40. Diagrama de Secuencia F3.1.1.
Elaboración propia.

4.2.4.11. F3.1.2 Consulta por parámetro de Ejercicios

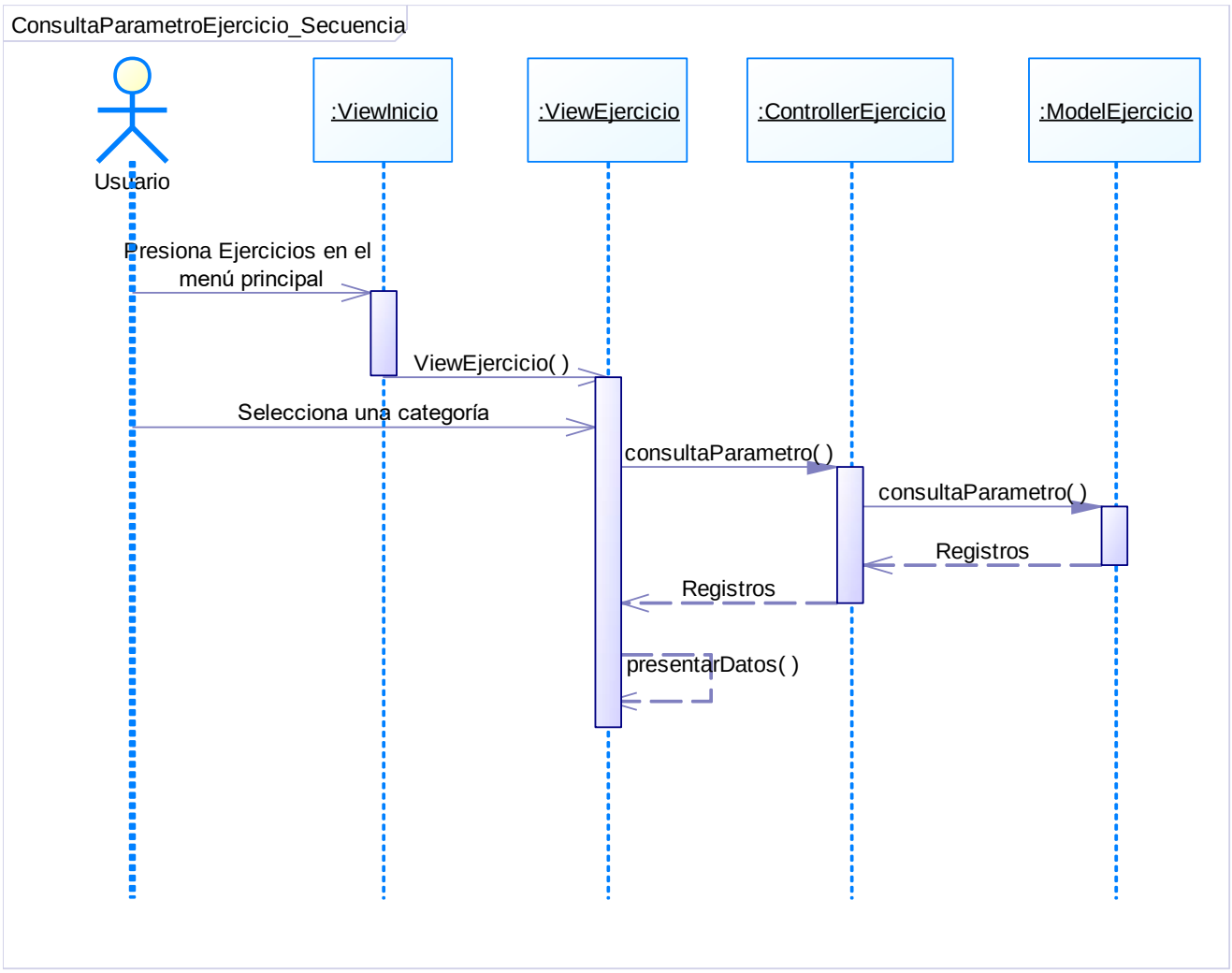


Figura 41. Diagrama de Secuencia F3.1.2.
Elaboración propia

4.2.4.12. F3.2 Realizar Ejercicio

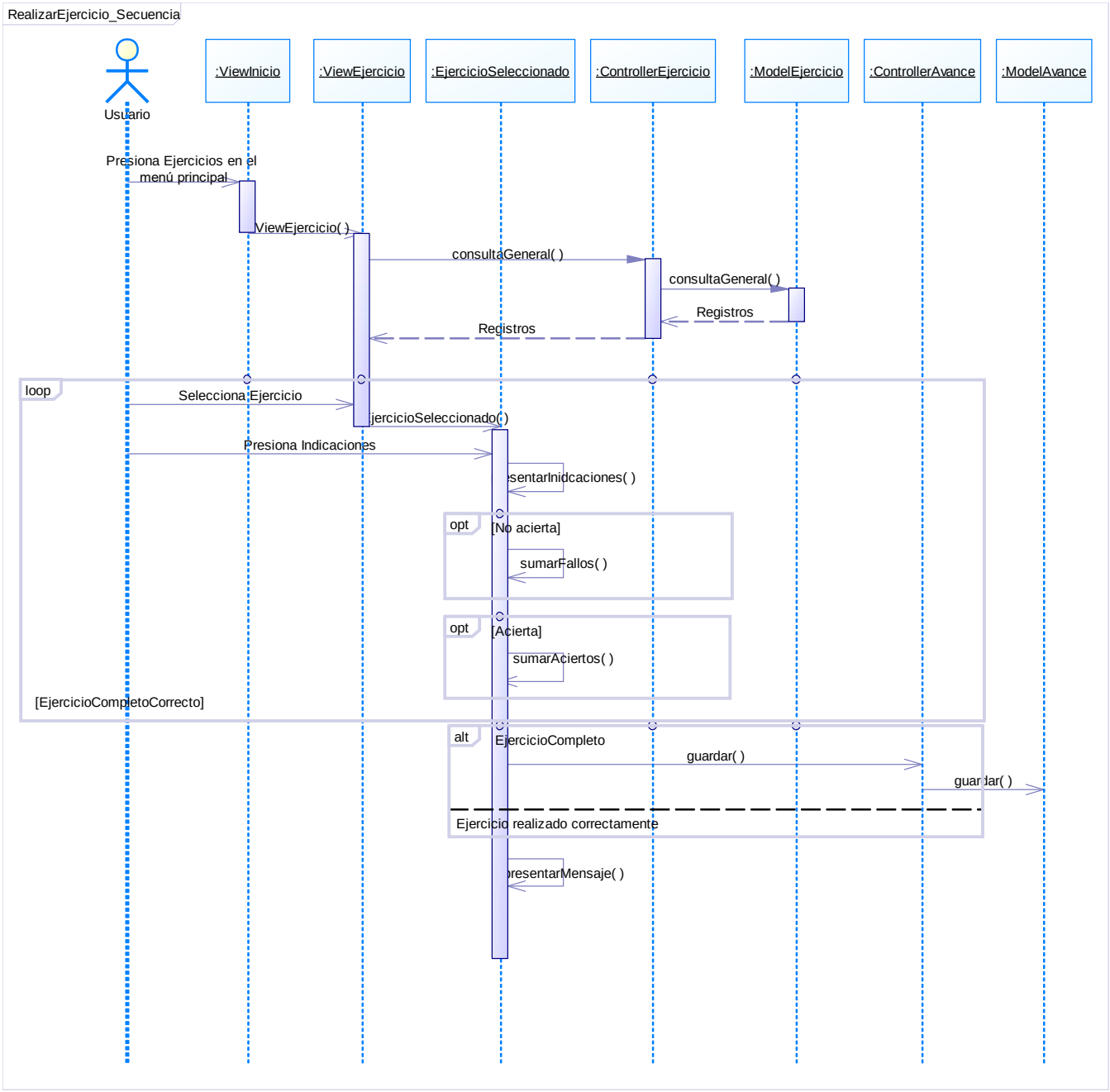


Figura 42. Diagrama de Secuencia F3.2.
Elaboración propia.

4.3. Desarrollo

4.3.1. Introducción

En esta sección se muestra el desarrollo de la aplicación de acuerdo a lo descrito en la etapa de requerimientos, análisis y diseño. Posterior a la codificación de la aplicación se realizarán pruebas del sistema, mismas que constan más adelante en este documento.

4.3.2. Estándares de programación

4.3.2.1. Manejo de paquetes

Por la utilización del modelo Vista-Controlador, los paquetes se dividirán en tres: vistas, controladores y modelos. El nombre de cada uno será:

- Vista: View
- Controlador: Controller
- Modelo: Model

4.3.2.2. Manejo de variables

Para mantener el concepto de encapsulamiento, las variables de las clases serán de tipo privado. Estas serán nombradas dependiendo de su uso y entidad a la que pertenecen. Por ejemplo, para el nombre usuario sería: Nombre_Usuario.

4.3.3. Imágenes del sistema

4.3.3.1. Iniciar Sesión (Login)

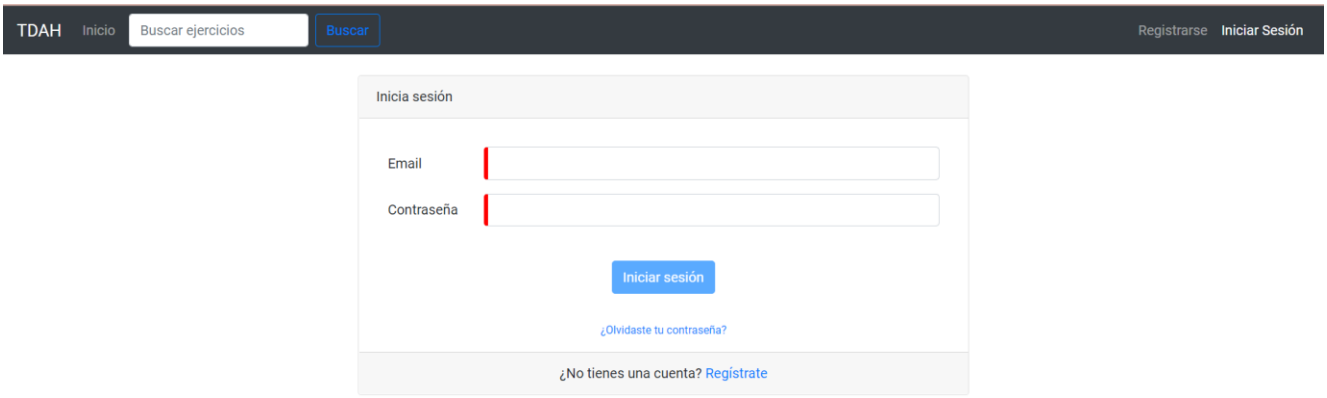


Figura 43. Login.
Elaboración propia

4.3.3.2. Registro

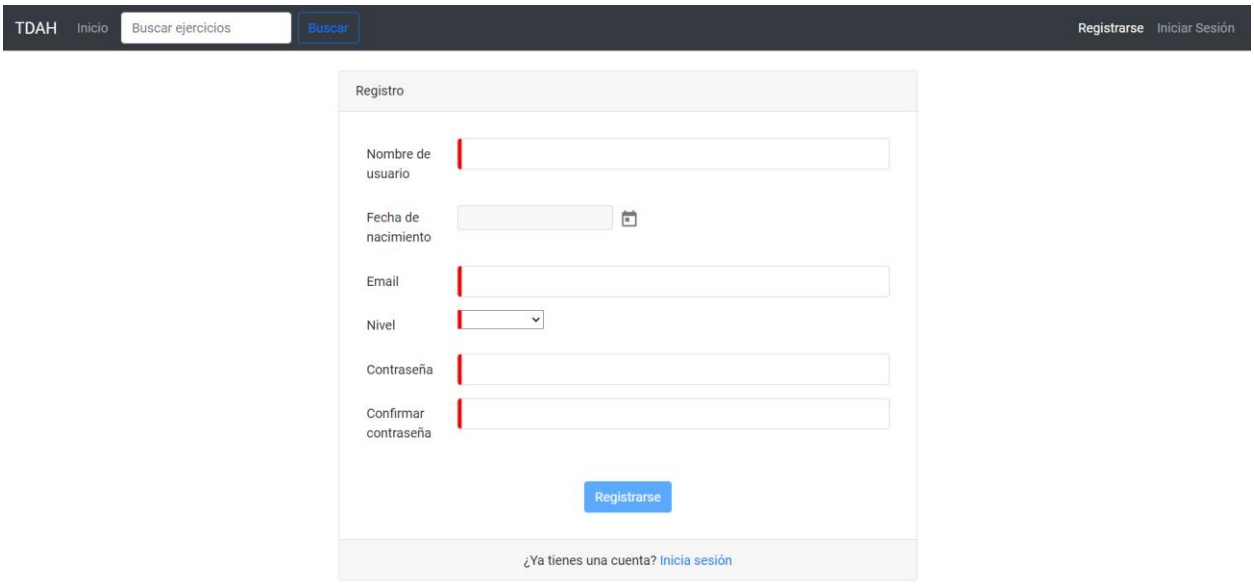


Figura 44. Registro.
Elaboración propia

4.3.3.3. Página principal



Figura 45. Página principal.
Elaboración propia

4.3.3.4. Ejercicio "El ahorcado"



Figura 46. El Ahorcado. Elaboración propia

4.3.3.5. Ejercicio “Cartas de memoria”



Figura 47. Cartas de memoria.
Elaboración propia

4.3.3.6. Ejercicio “Semejanzas y Diferencias”

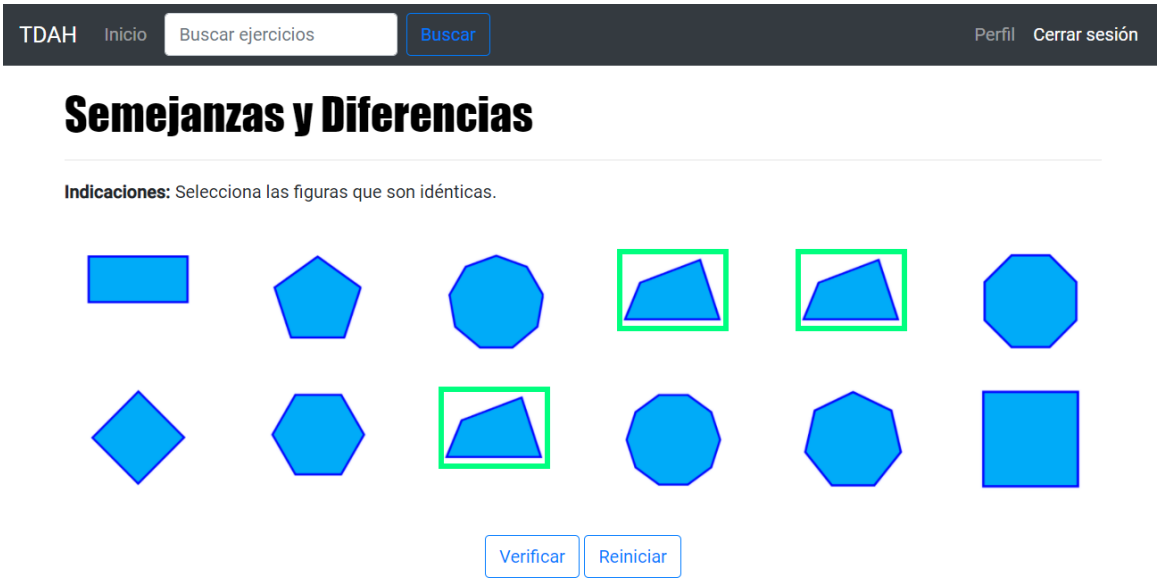


Figura 48. Semejanzas y diferencias.
Elaboración propia

4.3.3.7. Ejercicio “Descifrar mensaje”



Figura 49. Descifrar mensaje.
Elaboración propia.

4.3.3.8. Ejercicio “Detección de cantidades”



Figura 50. Detección de cantidades.
Elaboración propia

4.3.3.9. Ejercicio “Sombras de figuras”

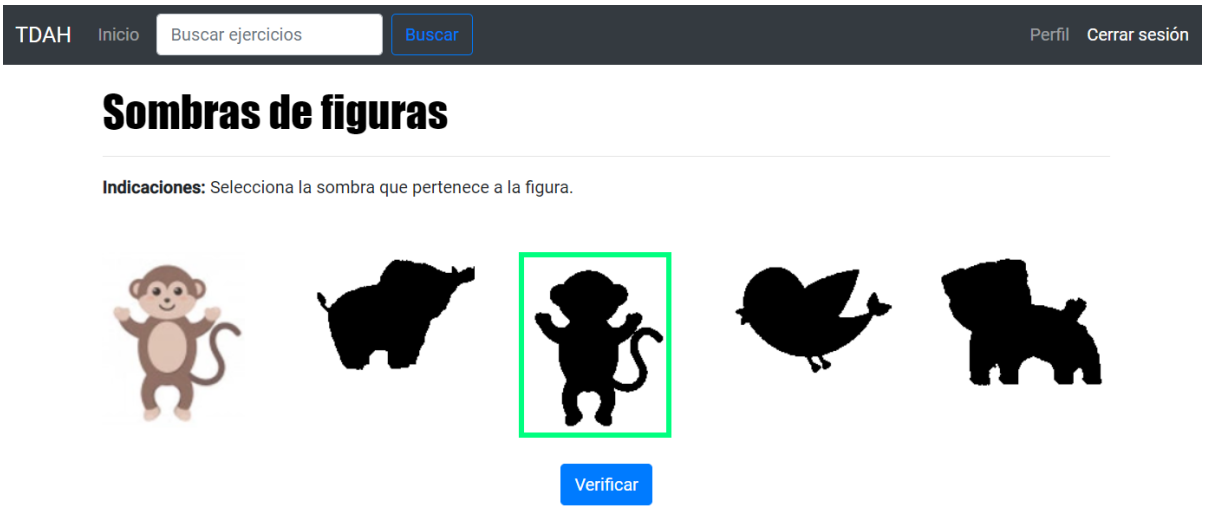


Figura 51. Sombras de figuras.
Elaboración propia.

4.3.3.10. Ejercicio “Contar colores”



Figura 52. Contar colores.
Elaboración propia.

4.3.3.11. Ejercicio “Ordenar los números”

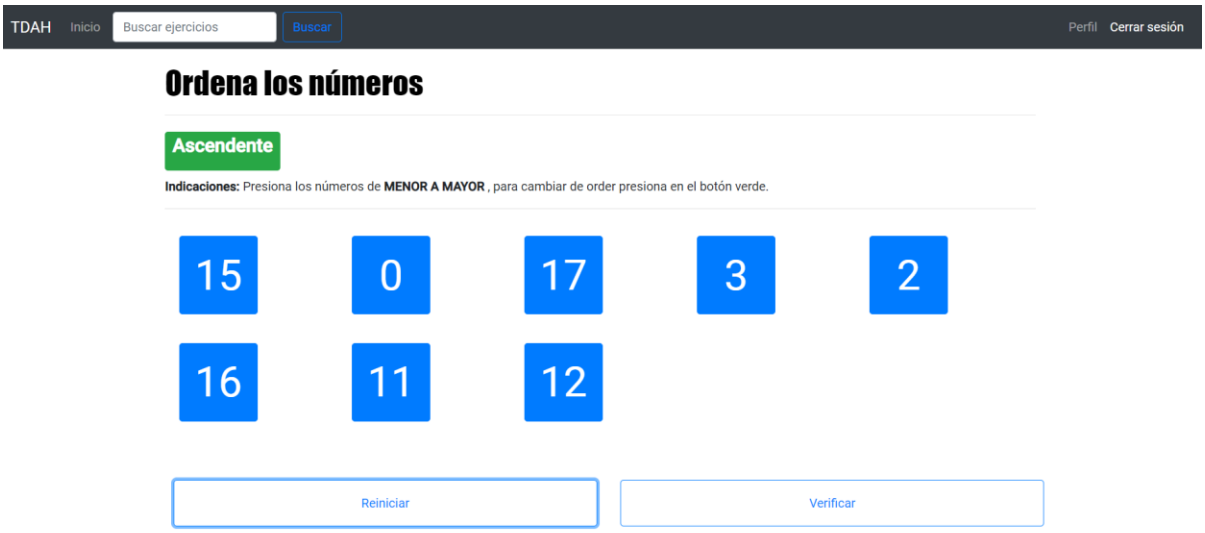


Figura 53. Ordenar los números.
Elaboración propia.

4.3.3.12. Ejercicio “Cartas de memoria (sonidos)”



Figura 54. Cartas de memoria sonidos.
Elaboración propia.

4.3.3.13. Ejercicio “Detección de animales”

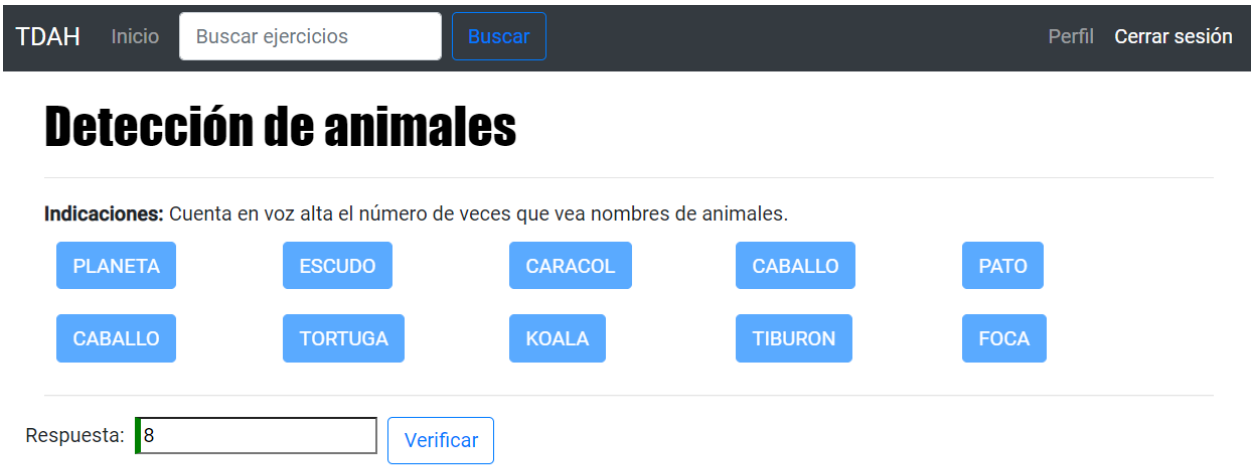


Figura 55. Detección de animales.
Elaboración propia.

4.3.3.14. Ejercicio “Historia”

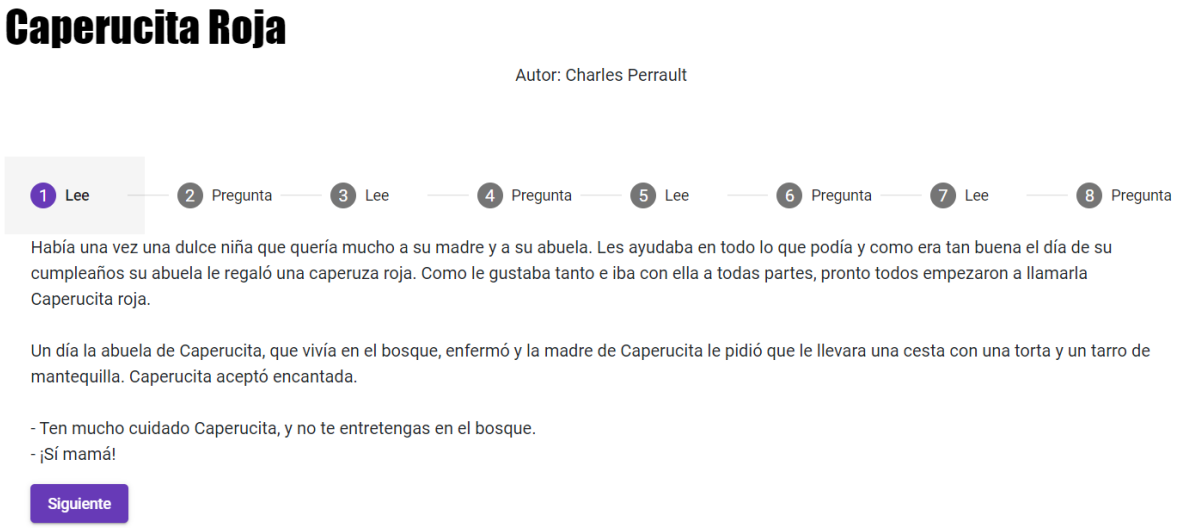


Figura 56. Historia1.
Elaboración propia.

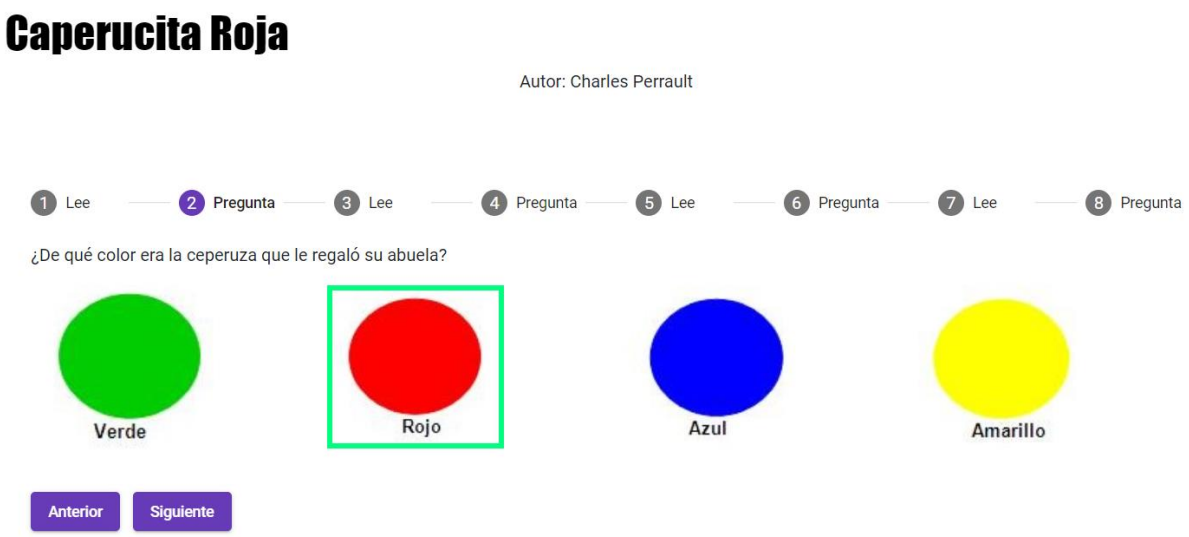


Figura 57. Historia 2.
Elaboración propia.

4.3.3.15. Perfil de usuario

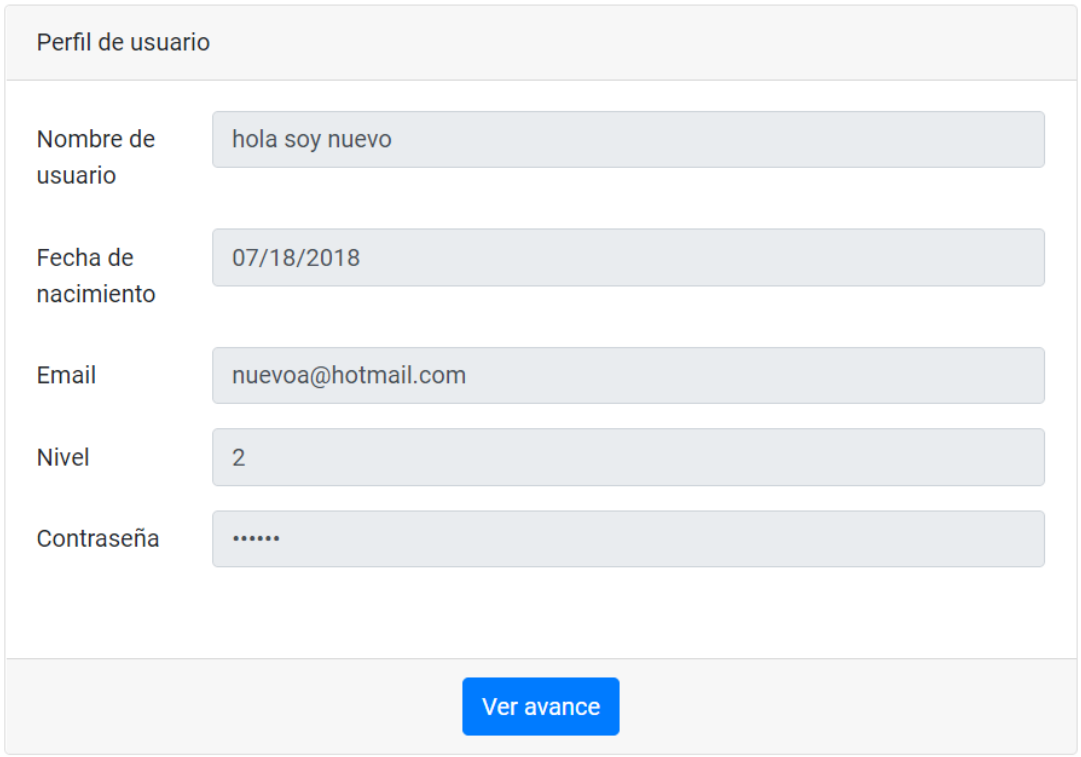


Figura 58. Perfil de usuario.
Elaboración propia.

4.3.3.16. Avance del usuario

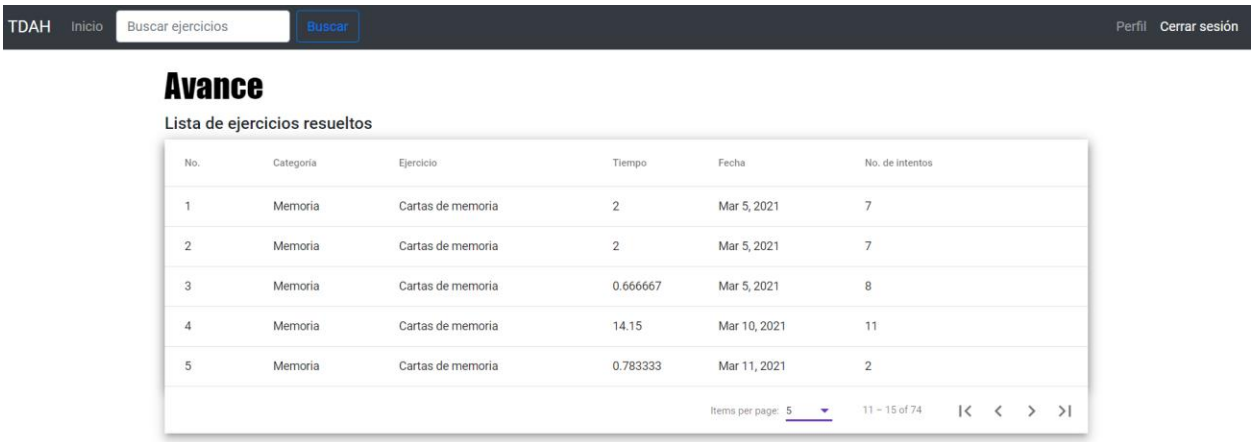


Figura 59. Avance general.
Elaboración propia.

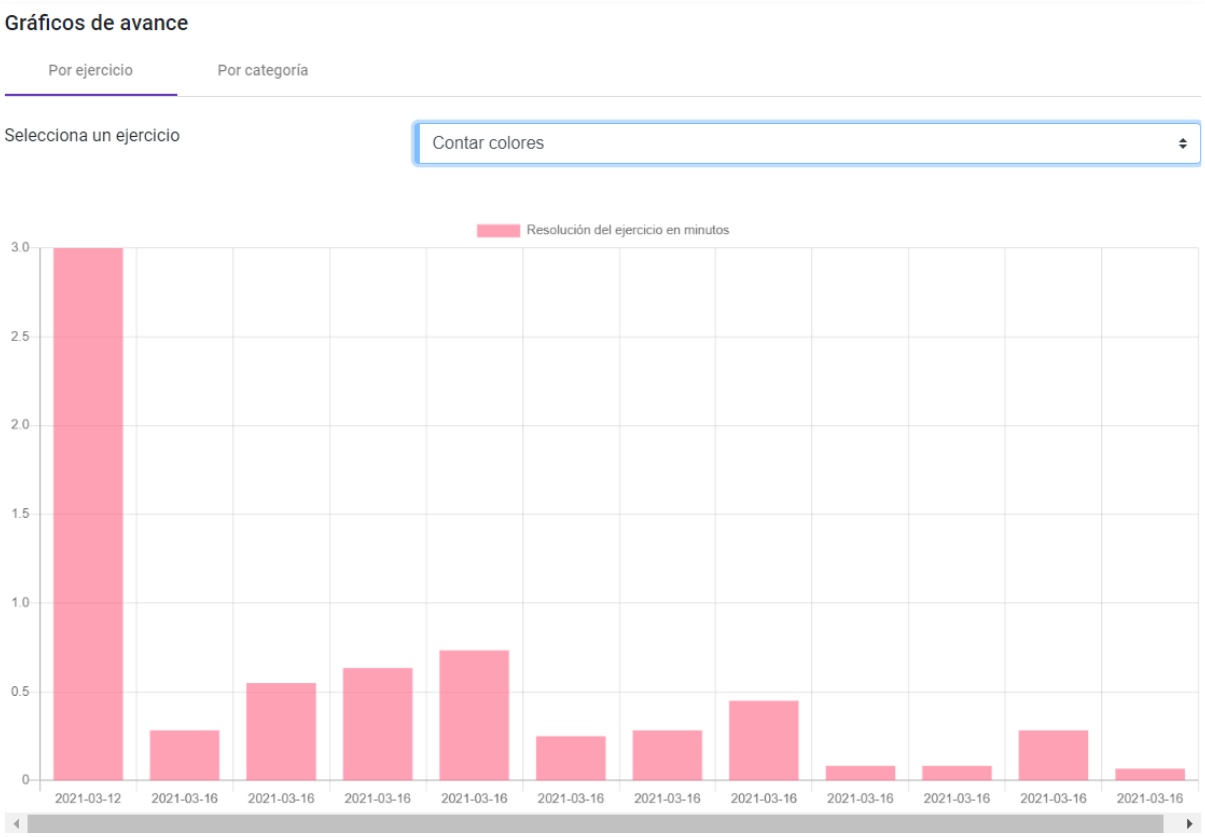


Figura 60. Avance por ejercicio.
Elaboración propia.

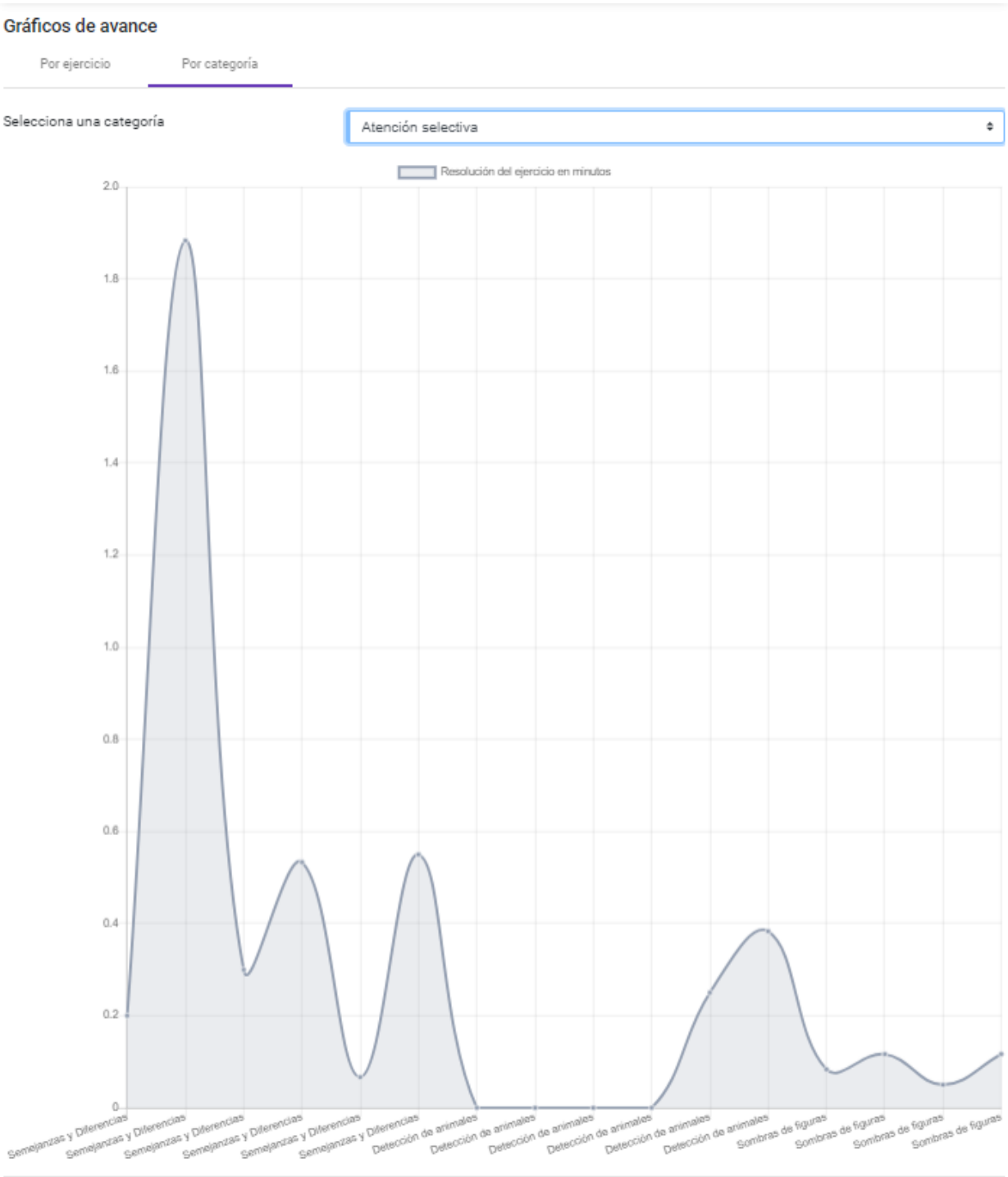


Figura 61. Avance por categoría.
Elaboración propia.

4.3.3.17. Mensaje de resolución correcta



Figura 62. Ejercicio correcto.
Elaboración propia

4.3.3.18. Mensaje de resolución incorrecta



Figura 63. Ejercicio incorrecto.
Elaboración propia.

4.3.4. Sprints

El presente trabajo se divide en cuatro Sprints, en los cuales se desarrollan las diferentes funcionalidades del sistema. Estos fueron divididos por la complejidad y tiempo que iba a requerir cada uno de ellos.

4.3.4.1. Primer Sprint

En este Sprint se desarrolla la funcionalidad de ingreso al sistema que comprende el backend y el frontend de registro e inicio de sesión, adicional se realiza la implementación de algunos ejercicios como: Cartas de memoria, el Ahorcado y Semejanzas y Diferencias; las cuales pertenecen a la funcionalidad de Gestión de Ejercicios.

SPRINT				
ID	TAREA	PRIORIDAD	ESTADO	RESPONSABLE
1	Desarrollo de registro al sistema	Alta	Completa	Valeria De La Torre
2	Desarrollo de inicio de sesión	Alta	Completa	Valeria De La Torre
3	Desarrollo de pantalla principal	Media	Completa	Valeria De La Torre
4	Desarrollo del ejercicio: El Ahorcado	Alta	Completa	Valeria De La Torre
5	Desarrollo del ejercicio: Cartas de memoria	Alta	Completa	Valeria De La Torre
6	Desarrollo del ejercicio: Semejanzas y Diferencias	Alta	Completa	Valeria De La Torre

4.3.4.2. Segundo Sprint

En este Sprint se desarrollan los siguientes ejercicios: Descifrar Mensaje, Detección de cantidades, Sombra de figuras, Contar Colores. Adicional se implementa la búsqueda de los ejercicios y la gestión de Usuarios para el perfil del administrador.

SPRINT				
ID	TAREA	PRIORIDAD	ESTADO	RESPONSABLE
1	Desarrollo del ejercicio: Descifrar Mensaje	Alta	Completa	Valeria De La Torre
2	Desarrollo del ejercicio: Detección de cantidades	Alta	Completa	Valeria De La Torre
3	Desarrollo del ejercicio: Sombras de figuras	Alta	Completa	Valeria De La Torre
4	Desarrollo del ejercicio: Contar Colores	Alta	Completa	Valeria De La Torre
5	Desarrollo de la sección de búsqueda de ejercicios	Media	Completa	Valeria De La Torre
6	Gestión de usuarios	Alta	Completa	Valeria De La Torre

4.3.4.3. Tercer Sprint

En este sprint se desarrollan los ejercicios: Ordenar Números, Cartas de memoria (sonidos), Detección de animales e Historia.

SPRINT				
ID	TAREA	PRIORIDAD	ESTADO	RESPONSABLE
1	Desarrollo del ejercicio: Ordenar números	Alta	Completa	Valeria De La Torre
2	Desarrollo del ejercicio: Cartas de memoria (sonidos)	Alta	Completa	Valeria De La Torre
3	Desarrollo del ejercicio: Detección de Historia	Alta	Completa	Valeria De La Torre
4	Desarrollo del ejercicio: Historia	Alta	Completa	Valeria De La Torre

4.3.4.4. Cuarto Sprint

En el último sprint se desarrolla el perfil de usuario y la gestión de avance, donde se puede visualizar la información de cada una de las realizaciones de los ejercicios, así como el informe gráfico que representa el avance por ejercicios y por categorías.

SPRINT				
ID	TAREA	PRIORIDAD	ESTADO	RESPONSABLE
1	Desarrollo de Perfil de usuario (editar o eliminar)	Alta	Completa	Valeria De La Torre
2	Desarrollo de gestión de avance (tabla de avances)	Alta	Completa	Valeria De La Torre
3	Desarrollo de gráfica de avance por ejercicio	Alta	Completa	Valeria De La Torre
4	Desarrollo de gráfico de avance por categoría	Alta	Completa	Valeria De La Torre

4.3.5. Pruebas del sistema

F0: Ingreso al sistema

Descripción: En esta funcionalidad el actor puede ingresar al sistema

Precondiciones: Existencia del usuario en el sistema

Funcionalidad	Datos Entrada/Salida	Resultados esperados	Resultados obtenidos
F0.1 Iniciar sesión	- Credenciales válidas	- Acceso al sistema	- Satisfactorio
	- Credenciales no válidas	- Mensaje de usuario y/o contraseña incorrectos - Usuario no registrado	- Satisfactorio

Tabla 21. Ingreso al sistema.
Elaboración propia.

Postcondiciones: Gestión de Usuario, Gestión de Ejercicios, Gestión de Avance

F1: Gestión de Usuarios

Descripción: En esta funcionalidad el actor puede crear un nuevo usuario; eliminar y modificar su propio usuario; y, en el caso de ser Administrador puede consultar los usuarios existentes en el sistema.

Precondiciones: -

Funcionalidad	Datos Entrada/Salida	Resultados esperados	Resultados obtenidos
F1.1 Crear usuario	- Tipo de datos validos	- Mensaje de datos incorrectos.	- Satisfactorio
	- Tipo de datos no válidos	- Creación del usuario en la base de datos.	- Satisfactorio
F1.2 Modificar usuario	- Tipo de datos no validos	- Mensaje de datos incorrectos.	- Satisfactorio
	- Tipo de datos válidos	- Actualización de datos en la base de datos.	- Satisfactorio
F1.3 Eliminar usuario	- Confirmación de eliminación.	- Eliminar los datos en la base de datos.	- Satisfactorio
F1.4.1 Consulta general	- Mostrar usuarios existentes en la base de datos.	- Mostrar en pantalla una lista de los usuarios.	- Satisfactorio
	- Mensaje de que no existen usuarios.	- Mensaje en el caso de que no existan usuarios.	- Satisfactorio
F1.4.2 Consulta por parámetro	- Tipo de dato válido	- Mostrar en pantalla una lista de los usuarios que coincidan con el parámetro a buscar.	- Satisfactorio
	- Tipo de dato no válido	- Mensaje de dato incorrecto.	- Satisfactorio

	- Mensaje que indica que no existen usuarios que coincidan con el parámetro a buscar.	- Mensaje en el caso que no existan usuarios que coincidan con el parámetro a buscar.	- Satisfactorio

Tabla 22. Gestión de Usuarios.
Elaboración propia.

Postcondiciones: Gestión de Ejercicios y Gestión de Avance.

F2: Gestión de Avance

Descripción: En esta funcionalidad el actor podrá modificar, eliminar o consultar el avance de los ejercicios que ha resuelto.

Precondiciones: Acceder al sistema y realizar ejercicios

Funcionalidad	Datos Entrada/Salida	Resultados esperados	Resultados obtenidos
F2.1 Modificar Avance	- Realizar ejercicio	- Actualización de información en la base de datos.	- Satisfactorio
F2.3.1 Consultar General	- Datos del usuario	- Mostrar en pantalla en avance de los ejercicios realizados	- Satisfactorio
F2.3.2 Consulta por parámetro	- Ingreso de dato del parámetro	- Mostrar en pantalla el avance que coincida con el parámetro a buscar.	- Satisfactorio

Tabla 23. Pruebas Gestión de Avance.
Elaboración propia.

Postcondiciones: -

F3: Gestión de Ejercicios

Descripción: En esta funcionalidad el actor podrá buscar o realizar cualquier ejercicio que exista en el sistema.

Precondiciones: crear Usuario, Ingreso al sistema

Funcionalidad	Datos Entrada/Salida	Resultados esperados	Resultados obtenidos
F3.1.1 Consulta general	- Ejercicios existentes en el sistema	- Mostrar en pantalla los ejercicios con los datos correspondientes.	- Satisfactorio
F3.1.2 Consulta por parámetro	- Valor del parámetro	- Mostrar en pantalla los ejercicios que coincidan con el parámetro a buscar.	- Satisfactorio
F3.2. Realizar ejercicio	- Realización del ejercicio	- Obtener el resultado del ejercicio resuelto	- Satisfactorio

Tabla 24. Pruebas Gestión de Ejercicios.
Elaboración propia.

5. Conclusiones y Recomendaciones

5.1. Conclusiones

1. La importancia de la tecnología web cada día es mayor, ya que permite que las personas puedan conectarse a las aplicaciones desde cualquier dispositivo. El aplicativo web desarrollado para el tratamiento del TDAH ayuda a que los niños puedan realizar los ejercicios junto con su terapeuta o en su hogar, brindando disponibilidad y facilidad a la hora de analizar el avance en el desarrollo de sus capacidades cognitivas.
2. Los objetivos de este trabajo fueron la selección y desarrollo de ejercicios que sirvan para el tratamiento de TDAH, los cuales se cumplieron con el acompañamiento de una especialista en el área. La experiencia de la terapeuta ayudó en la selección de los ejercicios que mejor se adapten a las necesidades de los pacientes sean implementados en un ambiente web.
3. Los ejercicios seleccionados fueron adaptados para el desarrollo en una plataforma web. El desarrollo de un aplicativo web trae varios beneficios como facilidad de uso y

accesibilidad para los usuarios, ya que se pueden conectar desde cualquier dispositivo con acceso a internet manteniendo su información y avance en cada sesión, así como también, facilita la escalabilidad y rápida actualización del sistema.

4. Las pruebas funcionales que se realizaron al sistema permitieron comprobar que el mismo realiza correctamente las funciones que fueron especificadas en los requerimientos, como la administración de usuarios, avance y ejercicios.
5. Se realizaron pruebas de facilidad de uso, las cuales arrojaron resultados positivos en cuanto a la adaptabilidad de la aplicación a las necesidades de los usuarios, ya que estos son niños, se debe tomar en cuenta ciertos aspectos como colores, tamaño de fuente y sonidos utilizados para una mejor interacción con el sistema.
6. La utilización de la metodología SCRUM para el desarrollo del sistema permitió corregir y modificar diferentes aspectos de la aplicación en cada Sprint según el criterio de la especialista, esto conllevó a la creación de un programa que cumpla con los requerimientos establecidos anteriormente.
7. Utilizar el patrón Modelo-Vista-Controlador para el desarrollo de la aplicación trae varias ventajas como la escalabilidad y mantenimiento de esta. Con este modelo se pudo dividir en diferentes capas las lógicas utilizadas y facilita la integración de los diferentes módulos existentes en el sistema.

5.2. Recomendaciones

1. Cada ejercicio debe ser ejecutado según el criterio del terapeuta, es decir, que el aplicativo web debe ser un apoyo al tratamiento propuesto por el especialista y este debe establecer el nivel en el cual se encuentra el paciente para la resolución de los ejercicios.
2. El sistema proporciona la información del resultado de cada ejercicio por usuario que se resume en su avance, sin embargo, se debe tomar en cuenta que esta información es útil por el terapeuta para analizar la situación del paciente y bajo su criterio poder considerar algún cambio en su tratamiento o aumento de dificultad para la resolución de las actividades.
3. Es recomendable informarse sobre el TDAH para conocer los efectos secundarios que provoca no tratarse del mismo y la importancia del correcto tratamiento. De tal manera que

se intenta generar conciencia sobre el problema que enfrentan las personas que padecen del trastorno por la falta de información de las personas que los rodean.

4. Para un trabajo futuro, se podría implementar más ejercicios para el tratamiento del TDAH, con lo cual se lograría tener más diversidad al momento que los pacientes realicen tu terapia.
5. Se recomienda llevar la documentación adecuada dependiendo de la metodología seleccionada. Es importante tener bien claro los requerimientos del sistema antes de diseñar y desarrollar, con el objetivo de evitar demasiados cambios a lo largo del desarrollo.
6. Se debe tener un respaldo del código realizado y de la base de datos, ya que puede surgir un error informático. De esta manera se asegura la accesibilidad a la información en cualquier momento durante el desarrollo y posterior a este.
7. Realizar un mantenimiento a la aplicación para la actualización de ejercicios y métricas de estos, siempre con el acompañamiento de un profesional.

Referencias

Alda, J. A., Serrano, E., Ortiz, J. J., & San, L. (Mayo de 2015). *El TDAH y su tratamiento.* Obtenido de Servicio de Psiquiatría y Psicología infanto-juvenil : https://faros.hsjdbcn.org/adjuntos/1830.1-pildora_tdah.pdf

Alonso, A. (10 de Diciembre de 2015). *Esta es la razón por la que el TDAH ha aumentado entre los hispanos*. Obtenido de PSYCIENCIA: https://www.psyciencia.com/esta-es-la-razon-por-la-que-el-tdah-ha-aumentado-entre-los-hispanos/

Angular. (2020). *FEATURES & BENEFITS*. Obtenido de angular.io: https://angular.io/features

Ansón Artero, L., Bayés Marín, I., Gavara Beltran , F., Giné Rabadán, A., Nuez Hernández, C., & Torrea Araiz, I. (Junio de 2015). *Cuaderno de ejercicios de estimulación cognitiva para reforzar la memoria.* Obtenido de Consorci Sanitari Integral: www.csi.cat

Arjonilla, R. (2020). *BackEnd*. Obtenido de https://rafarjonilla.com/que-es/backend/

Bean, M. (2015). *Laravel 5 Essentials.* Birmingham: Packt Publishing.

Berenguel Gómez, J. L. (2015). *Desarrollo de aplicaciones web en el entorno servidor.* Madrid: Ediciones Paraninfo.

Bitbrain. (16 de Agosto de 2018). *12 tipos de actividades y ejercicios de estimulación cognitiva*. Obtenido de Bitbrain.com: https://www.bitbrain.com/es/blog/ejercicios-estimulacion-cognitiva

Bitbrain. (20 de Julio de 2018). *Qué es el TDAH, síntomas, tratamiento y estimulación cognitiva*. Obtenido de bitbrain: https://www.bitbrain.com/es/blog/rehabilitacion-cognitiva-tdah#:~:text=Estimulaci%C3%B3n%20cognitiva%20para%20personas%20con%20TDAH&text=Una%20forma%20cl%C3%A1sica%20de%20rehabilitaci%C3%B3n,%2C%20atenci%C3%B3n%2C%20orientaci%C3%B3n%2C%20etc.

Brad Green, S. S. (2013). *AngularJS.* O'Reilly Media .

Caballero Legarda, L. Á., Moya Alonso, G., Romero Seijas, L., & Hernández Millán, A. (2019). Nivel de conocimiento sobre el trastorno por déficit de atención e hiperactividad en estomatólogos. *Revista Científica Estudiantil de Cienfuegos INMEDSUR*, 12-19.

Cobo, Á. (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web.* Ediciones Díaz de Santos.

CogniFit. (Julio de 2020). *Cognición Y Ciencia Cognitiva*. Obtenido de cognifitcom: https://www.cognifit.com/es

Cruz Velarde. (08 de Mayo de 2019). *TDAH: Atención selectiva, sostenida y dividida*. Obtenido de Neuronae: https://www.neuronae.net/post/2016/03/01/tdah-atenci%C3%B3n-selectiva-sostenida-y-dividida

Díaz González, Y., & Fernández Romero, Y. (2012). PATRÓN MODELO-VISTA-CONTROLADOR. *Revista Telemática*, 47-57.

eclass. (22 de Abril de 2020). *¿Qué es un Sprint? Conoce el corazón de Scrum*. Obtenido de eclass: https://blog.eclass.com/que-es-un-sprint-conoce-el-corazon-de-scrum

Eng, R. (15 de Enero de 2020). *Cómo aprender Smalltalk puede convertirlo en un mejor desarrollador*. Obtenido de TechBeacon: https://techbeacon.com/app-dev-testing/how-learning-smalltalk-can-make-you-better-developer

Fundación CADAH. (s.f.). *Dificultad en el diagnóstico del TDAH en la edad preescolar*. Obtenido de fundacioncadah.org: https://www.fundacioncadah.org/web/articulo/dificultad-en-el-diagnostico-del-tdah-en-la-edad-preescolar.html#:~:text=Los%20manuales%20de%20diagn%C3%B3stico%20fijan,en%20sus%20respuestas%20al%20entorno.

Galán, D. (11 de Marzo de 2020). *¿Qué es AngularJS y por qué deberías usarlo?* Obtenido de ifgeekthen: https://ifgeekthen.everis.com/es/que-es-angularjs-y-por-que-deberias-usarlo

Gauchat, J. (2012). *El gran libro de HTML5, CSS3 y JavaScript.* Barcelona: Marcombo.

González Collantes, R., Rodríguez Sacristán, A., & Sánchez García, J. (2015). Epidemiología del TDAh. *REVISTA ESPAÑOLA DE PEDIATRÍA*, 71(2): 58-61.

Gutiérrez, J. J. (12 de Mayo de 2014). *¿Qué es un framework web?* Obtenido de http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

James, M., & Walter, L. (2010). *Scrum Reference Card.* Obtenido de CollabNet, Inc.: https://www.collab.net/sites/default/files/uploads/CollabNet_scrumreferencecard.pdf

Klimenko, O. (2009). La enseñanza de las estrategias cognitivas y metacognitivas como una vía de apoyo para el aprendizaje autónomo en los niños con déficit de atención sostenida. *Revista Virtual Universidad Católica del Norte*, 27.

Loro López, M., Quintero, J., García Campos, N., Jiménez Gómez, B., Pando, F., Varela Casal, P., . . . Correas Lauffer, J. (2009). Actualización en el tratamiento del trastorno por déficit de atención/hiperactividad. *Revista de Neurología*, 257-264.

Luján Mora, S. (2002). *Programación de aplicaciones web: historia, principios básicos y clientes web.* san Vicente (Alicate): Editorial Club Universitario.

Luna, F., Peña Millahual, C., & Iacono, M. (2018). *PROGRAMACION WEB Full Stack 13 - PHP: Desarrollo frontend y backend - Curso visual y práctico.* Buenos Aires: RedUsers.

Martínez Sibón, A. M. (Marzo de 2010). ¿Qué es el TDAH? *Innovación y Experiencia*(28), 1-10.

Meca Martínez, R. (2013). *Juegos para entrenar y mejorar dificultades en el TDAH*. Obtenido de Fundación CADAH: https://www.fundacioncadah.org/web/articulo/juegos-para-entrenar-y-mejorar-dificultades-en-el-tdah-.html

Mina, M. A. (2018). Análisis comparativo entre ASP. NETy PHP. *INNOVA Research Journal*, 3(4), 25-43.

NeuronUP. (08 de Agosto de 2018). *Funciones cognitivas*. Obtenido de neuronup.com: https://www.scribbr.es/detector-de-plagio/generador-apa/new/webpage/

Organización Panamericana de la Salud. (20 de Agosto de 2013). *Amamantar puede reducir el riesgo de hiperactividad infantil.* Obtenido de OPS: https://www.paho.org/ecu/index.php?option=com_content&view=article&id=998:agosto-20-2013&Itemid=972

Ostrocky-Solís, F., Gómez Pérez, E., Chayo-Dichy, R., & Flores Lázaro, J. C. (2004). *¿Problemas de atención? Un programa para su estimulación y rehabilitación.* México, D.F., México: American Book Store.

Parekh, R. (Julio de 2017). *AMERICAN PSYCHIATRIC ASSOCIATION*. Obtenido de What Is ADHD?: https://www.psychiatry.org/patients-families/adhd/what-is-adhd

Pérez, J. E. (2019). *Introducción a JavaScript.* Obtenido de http://190.57.147.202:90/xmlui/bitstream/handle/123456789/430/introduccion_javascript%20(1).pdf?sequence=1

Ponguta, E. (12 de Febrero de 2019). *Front-Ed y Back-End*. Obtenido de https://edwinponguta.blogspot.com/2019/02/caracteristicas-y-complementariedad.html

Ramos Quiroga, J. A., Bosch, R., & Casas , M. (03 de Marzo de 2009). *Comprender el TDAH en adultos.* AMAT.

Ramos Usuga, D. (20 de Febrero de 2018). *Rehabilitación de la atención alternante*. Obtenido de neuroup: https://blog.neuronup.com/rehabilitacio-atencion-alternante/

Ramos, C. A., Bolaños, M. F., & Ramos, D. A. (Diciembre de 2015). PREVALENCIA DEL TRASTORNO POR DÉFICIT DE ATENCIÓN CON HIPERACTIVIDAD EN ESTUDIANTES ECUATORIANOS. *Revista Científica y Tecnológica UPSE, III*(1), 13-19.

Regader, B. (2019). *Dopamina: 7 funciones esenciales de este neurotransmisor*. Obtenido de Psicología y Mente: https://psicologiaymente.com/neurociencias/dopamina-neurotransmisor

Rehkopf, M. (2020). *Sprints*. Obtenido de ATLASSIAN: https://www.atlassian.com/agile/scrum/sprints#:~:text=What%20are%20sprints%3F-,A%20sprint%20is%20a%20short%2C%20time%2Dboxed%20period%20when%20a,better%20software%20with%20fewer%20headaches.

Reza Suárez, L., Solórzano Almazán, M., Erazo Ruiz, E., Oñate Amaya, F. X., & Lamingo Soriano, G. (Noviembre de 2018). EL TDAH Y SU REPERCUSIÓN EN EL RENDIMIENTO ACADÉMICO. *Revista ATLANTE*. Obtenido de https://www.eumed.net/rev/atlante/2018/11/tdah-rendimiento-academico.html

Rodriguez Puerta, A. (18 de Diciembre de 2018). *Atención focalizada: características, teorías, tests y actividades*. Obtenido de lifeder.com: https://www.lifeder.com/atencion-focalizada/

Rosero, M. (01 de Mayo de 2017). *El déficit de atención se detecta cuando el niño va a la escuela*. Obtenido de El Comercio: https://www.elcomercio.com/tendencias/deteccion-deficit-atencion-ninos-escuela.html

Schwaber, K., & Sutherland, J. (2016). *La Guía de Scrum TM* . Obtenido de La Guía Definitiva de Scrum: Las Reglas del Juego: https://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Spanish.pdf#zoom=100

Scrum.org. (2020). *What is a Scrum Master?* Obtenido de The Home of Scrum: https://www.scrum.org/resources/what-is-a-scrum-master

Soutullo Esperón, C., & Mardomingo Sanz, M. J. (2010). *Manual de PSIQUIATRÍA DEL NIÑO Y DEL ADOLESCENTE*. Buenos Aires : Madrid: Editorial Médica Panamericana.

Sulkes, S. B. (Abril de 2018). *Trastorno por déficit de atención/hiperactividad (TDA, TDAH)*. Obtenido de MANUAL MSD: https://www.msdmanuals.com/es/professional/pediatr%C3%ADa/trastornos-del-aprendizaje-y-del-desarrollo/trastorno-por-d%C3%A9ficit-de-atenci%C3%B3n-hiperactividad-tda-tdah#

Trigas Gallego, M. (2012). *Metodología Scrum.* Obtenido de GESTIÓN DE PROYECTOS INFORMÁTICOS: http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf

Universidad de Alicante. (2020). *Modelo vista controlador (MVC)*. Obtenido de https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html#:~:text=Modelo%20Vista%20Controlador%20(MVC)%20es,control%20en%20tres%20componentes%20distintos.&text=La%20Vista%2C%20o%20interfaz%20de,los%20mecanismos%20interacci%C3

Valetanga, J. (11 de Julio de 2016). *Trastorno por déficit de atención afecta más a niños que a niñas*. Obtenido de ediciónmédica: https://www.edicionmedica.ec/secciones/salud-

publica/trastorno-por-d-ficit-de-atenci-n-e-hiperactividad-afecta-m-s-a-ni-os-que-a-ni-as-88201

Velarde, J. A. (08 de Mayo de 2019). *TDAH: Atención selectiva, sostenida y dividida*. Obtenido de Blog Neuronae: https://www.neuronae.net/post/2016/03/01/tdah-atenci%C3%B3n-selectiva-sostenida-y-dividida

Vilajosana Guilén, X., & Navarro Moldes, L. (2019). *Arquitectura de aplicaciones web.*