



PROYECTO INTEGRADOR II

DOCUMENTO DE DEFINICIÓN DEL PROYECTO INTEGRADOR 2024-10

THE NEXUS BATTLES II

LENIN JAVIER SERRANO GIL. MSc. SMPC.

Facultad de Ingeniería de Sistemas e Informática

Universidad Pontificia Bolivariana Sección Bucaramanga

Bucaramanga, 25 de noviembre de 2023



REVISIÓN Y CONTROL DE CAMBIOS

Revisión y versiones

Fecha	Versión aprobada	Cargo	Nombres y apellidos
2023-11-25	4.0	Autor	Lenin Javier Serrano Gil

Control de Cambios

Fecha	Versión	Descripción del cambio	Autor
2023-11-28	4.0	Actualización de la metodología, fases de requerimientos, diseño. Cambia “Planificación del Proyecto Integrador” por “Propuesta del Proyecto Integrador”	Lenin Javier Serrano Gil
2023-11-28	4.0	Aprobado	Facultad de ingeniería de sistemas e informática



Tabla de Contenido

PROYECTO INTEGRADOR II	7
Propósitos de formación del curso.....	7
Criterios de competencia	7
Desarrollo del trabajo académico	8
Régimen de calificaciones	11
Equipos de trabajo.....	12
THE NEXUS BATTLES II.....	13
SISTEMA DE COMERCIO ELECTRÓNICO	13
SOBRE INFRAESTRUCTURA EN LA NUBE.....	13
1. Situación problema.....	13
1.1. Descripción del Juego	13
2. Requerimientos del producto	16
2.1. Módulo para Carro de compras (E-commerce)	16
2.2. Módulo de administración e inventario de cartas	19
2.3. Módulo de usuarios y comentarios	19
2.4. Módulo de Chatbot.....	19
2.5. Módulo de correo electrónico	19
2.6. Módulo para Jugar online.....	19
2.7. Módulo del Torneo.....	21
2.8. Módulo de Subasta.....	22
2.9. Infraestructura.....	23
3. Requisitos del proyecto.....	23
4. Gestión del Proyecto Integrador y desarrollo del Producto.....	24
4.1. Metodología del curso	24
4.1.1. Propuesta del Proyecto Integrador.....	24
4.1.2. Análisis de requerimientos y planificación de la solución	25
4.1.3. Diseño y desarrollo del Proyecto Integrador	25
4.1.4. Análisis de resultados del Proyecto Integrador.....	26
4.1.5. Estrategias de divulgación de resultados del Proyecto Integrador	26



4.1.6. Cierre del Proyecto Integrador.....	26
4.2. Proceso de desarrollo de software para el Proyecto Integrador.....	26
4.2.1. Ciclo de desarrollo dirigido por pruebas.....	26
4.3. Marco de referencia para el desarrollo del Proyecto Integrador.....	27
4.3.1. Iniciación.....	27
4.3.1.1. Creación de la visión del proyecto y elección del dueño del producto (Product Owner).....	27
4.3.1.2. Elección del experto en desarrollo de proyectos en el marco de Scrum (Scrum Master).....	27
4.3.1.3. Formación del equipo Scrum (Scrum Team).....	28
4.3.1.4. Desarrollo de las Épicas (Epics).....	28
4.3.1.5. Creación de la lista prioriza de pendientes del producto (Product Backlog).....	28
4.3.1.6. Planificación del lanzamiento (Project Charter).....	29
4.3.2. Planificación y estimación.....	29
4.3.2.1. Creación de historias de usuario (User Stories).....	29
4.3.2.2. Estimación de historias de usuario.....	30
4.3.2.3. Comprometer historias de usuarios.....	30
4.3.2.4. Identificación de tareas (To do list).....	30
4.3.2.5. Estimación de tareas.....	30
4.3.2.6. Creación de la lista de pendientes del Sprint (Sprint Backlog).....	31
4.3.3. Implementación.....	31
4.3.3.1. Creación de entregables.....	31
4.3.3.2. Realización de la reunión seguimiento diaria (Daily Scrum).....	31
4.3.3.3. Refinamiento del Product Backlog.....	31
4.3.4. Revisión y Retrospectiva.....	32
4.3.4.1. Demostración y validación del Sprint.....	32
4.3.4.2. Retrospectiva del Sprint.....	32
4.3.5. Lanzamiento.....	33
4.3.5.1. Liberación de Entregables.....	33
4.3.5.2. Retrospectiva del Proyecto.....	33
5. Artefactos relacionados con el Proyecto Integrador.....	33
5.1. Documento de definición del Proyecto Integrador.....	33
5.2. Presentación de Propuesta del Proyecto Integrador.....	33



5.3. Propuesta del Proyecto Integrador	33
5.3.1. Acta de constitución del proyecto Integrador (Project Charter)	34
5.4. Lista de pendientes del producto (Product Backlog).....	34
5.5. Listas de pendientes del Sprint (Sprint Backlogs)	34
5.6. Acta de seguimiento del Proyecto Integrador.....	35
5.7. Acta de entrega y cierre del Proyecto Integrador.....	35
6. Objetivos de Aprendizaje.....	36
7. Actividades de aprendizaje.....	36
8. Fuentes de información.....	37



Lista de Tablas

Tabla 1. Desarrollo del trabajo académico.....	10
Tabla 2. Régimen de calificaciones.....	11
Tabla 3. Equipos de trabajo.....	12
Tabla 4. Estadísticas para los héroes o comandantes.....	14
Tabla 5. Set de armas para cada héroe.....	14
Tabla 6. Armadura para cada héroe.....	15
Tabla 7. Ítems para cada héroe.....	15
Tabla 8. Habilidades especiales o Épicas.....	15
Tabla 9. Creación de la visión del proyecto y elección del dueño del producto (Product Owner)	27
Tabla 10. Elección del experto en desarrollo de proyectos en el marco de Scrum (Scrum Master)	28
Tabla 11. Formación del equipo Scrum (Scrum Team)	28
Tabla 12. Desarrollo de las Épicas (Epics).....	28
Tabla 13. Creación de la lista prioriza de pendientes del producto (Product Backlog)	29
Tabla 14. Planificación del lanzamiento (Project Charter)	29
Tabla 15. Creación de historias de usuario (User Stories).....	30
Tabla 16. Estimación de historias de usuario	30
Tabla 17. Comprometer historias de usuarios.....	30
Tabla 18. Identificación de tareas (To do list).....	30
Tabla 19. Estimación de tareas	30
Tabla 20. Creación de la lista de pendientes del Sprint (Sprint Backlog).....	31
Tabla 21. Creación de entregables.....	31
Tabla 22. Realización de la reunión seguimiento diaria (Daily Scrum)	31
Tabla 23. Refinamiento del Product Backlog.....	32
Tabla 24. Demostración y validación del Sprint.....	32
Tabla 25. Retrospectiva del Sprint.....	32
Tabla 26. Liberación de Entregables.....	33
Tabla 27. Retrospectiva del Proyecto	33
Tabla 28. Acta de seguimiento.....	35



PROYECTO INTEGRADOR II

Este curso forma parte de la propuesta para el desarrollo de una estrategia metodológica y pedagógica dirigida a promover competencias en habilidades blandas, así como habilidades disciplinares para la resolución de problemas que requieren apropiación e innovación en tecnologías de la información. En Proyecto Integrador el estudiante hace uso de los conocimientos y prácticas asociados al plan de estudios, que le permiten el fortalecimiento de competencias para generar, gestionar y desarrollar proyectos de ingeniería.

Propósitos de formación del curso

Desarrollar proyectos de innovación para resolver problemas basados en situaciones reales, sociales u organizacionales que conlleven a integrar competencias en Ingeniería de Sistemas, Informática y Tecnologías de la Información. El curso se fundamenta en la metodología del aprendizaje basado en problemas y proyectos. Además, La asignatura se desarrolla a través de ejercicios de indagación, pensamiento crítico y aprendizaje continuo. Asimismo, el curso abarca la gestión eficiente del tiempo y los recursos, así como el mejoramiento de las habilidades de expresión oral y escrita para abordar problemas de investigación aplicada en entornos tecnológicos multidisciplinarios y multiculturales.

Criterios de competencia

- Materializo el conocimiento en nuevas aplicaciones (teorías, modelos, metodologías, prototipos, patentes, modelos de utilidad, diseños industriales, registros de software, procesos técnicos y tecnológicos, empresas) con recursividad y concordancia con las capacidades del contexto.
- Planteo soluciones a problemas organizacionales, científico y sociales, cumpliendo las etapas del ciclo de vida del software.
- Diseño soluciones computacionales con criterios de viabilidad, calidad, seguridad y buenas prácticas, que respondan a requerimientos funcionales y no funcionales.
- Desarrollo proyectos de software, aplicando metodologías, enfoques y herramientas de productividad.
- Diseño y desarrollo, de forma detallada, el sistema con base en las plataformas técnicas y tecnológicas disponibles y los avances del conocimiento.
- Empleo técnicas relativas al aseguramiento de la información que cumplan los criterios básicos de la seguridad informática para la protección de los activos de información.
- Utilizo los modelos y estándares para la gestión de incidentes en seguridad informática que conlleven a la mitigación de los riesgos asociados a los activos de información.

Desarrollo del trabajo académico

	Actividades de aprendizaje	Entregables y/o evaluación
1	<p>Definición del Proyecto Integrador: Presentación de la definición, el propósito de formación del curso, los criterios de competencia del curso, la definición de la situación problema del proyecto, requisitos del proyecto, requisitos del producto, gestión del proyecto, desarrollo del producto y lista de entregables.</p> <p>Herramientas: Editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias y software para el apoyo en el desarrollo de los proyectos.</p>	<ul style="list-style-type: none"> • Presentación (5 minutos): Introducción a la definición del problema, pregunta de investigación, objetivos, y metodología. • Documento: Análisis y Modelo UML del Dominio.
2	<p>Propuesta del Proyecto Integrador: Descripción y delimitación de la situación problemática, la pregunta de investigación, revisión bibliográfica, objetivos, actividades, entregables, justificación, metodología y presupuesto.</p> <p>Herramientas: Editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Documento: Propuesta del Proyecto Integrador. • Documento Anexo: Project Charter (Acta de constitución del proyecto). • Formulario: Acta de seguimiento.
3	<p>Análisis de requerimientos y planificación de solución: Especificación de requerimientos, análisis de los datos, cronograma presupuesto, seguimiento y control.</p> <p>Herramientas: Editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Product Backlog: Épicas e Historias de usuario. • Story Mapping. • Formulario: Acta de seguimiento.
4	<p>Análisis de requerimientos y planificación de solución: Especificación de requerimientos, análisis de los datos, cronograma presupuesto, seguimiento y control.</p> <p>Herramientas: Editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Sprint Backlog: Tareas. • Scrum Board. • Actualización: Story Mapping. • Refinamiento y actualización: Product Backlog. • Formulario: Acta de seguimiento.
5	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Ejecución: Sprint Backlog. • Ejecución: Daily Scrum. • Diseño: Modelos de infraestructura, estructura, comportamiento, datos y casos de prueba. • Entregables del proyecto: Todo artefacto que se utiliza para gestionar el proyecto. • Entregables del producto: Todo artefacto que se utiliza en el ciclo de desarrollo de software para obtener el producto.
6	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Ejecución: Test cases. • Actualización: Story Mapping. • Actualización: Scrum Board. • Refinamiento y actualización: Product Backlog. • Formulario: Acta de seguimiento.

7	<p>Análisis de resultados del Proyecto Integrador: validación y pruebas, verificación, solución de problemas encontrados, revisión del cumplimiento en tiempo y recursos presupuestales y estimación del impacto.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Avance: Informe. • Sprint Review: Lanzamiento del producto. • Retrospectiva del Sprint. • Sprint Burndown Chart. • Nuevo: Sprint Backlog. • Actualización: Story Mapping. • Actualización: Scrum Board. • Diseño: Modelos de arquitectura, estructura, comportamiento y datos. • Test cases. • Refinamiento y actualización: Product Backlog. • Formulario: Acta de seguimiento.
8	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Ejecución: Sprint Backlog. • Ejecución: Daily Scrum. • Diseño: Modelos de infraestructura, estructura, comportamiento, datos y casos de prueba. • Entregables del proyecto: Todo artefacto que se utiliza para gestionar el proyecto. • Entregables del producto: Todo artefacto que se utiliza en el ciclo de desarrollo de software para obtener el producto.
9	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Ejecución: Test cases. • Actualización: Story Mapping. • Actualización: Scrum Board. • Refinamiento y actualización: Product Backlog. • Formulario: Acta de seguimiento.
10	<p>Análisis de resultados del Proyecto Integrador: validación y pruebas, verificación, solución de problemas encontrados, revisión del cumplimiento en tiempo y recursos presupuestales y estimación del impacto.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Sprint Review: Lanzamiento del producto. • Retrospectiva del Sprint. • Sprint Burndown Chart. • Nuevo: Sprint Backlog. • Actualización: Story Mapping. • Actualización: Scrum Board. • Diseño: Modelos de arquitectura, estructura, comportamiento y datos. • Test cases. • Refinamiento y actualización: Product Backlog. • Acta de seguimiento.
11	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Ejecución: Sprint Backlog. • Ejecución: Daily Scrum. • Diseño: Modelos de infraestructura, estructura, comportamiento, datos y casos de prueba. • Entregables del proyecto: Todo artefacto que se utiliza para gestionar el proyecto.

12	<p>Diseño y desarrollo del Proyecto Integrador: Diseño de infraestructura, estructura, comportamiento, datos y casos de prueba. Desarrollo e implementación del sistema.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Entregables del producto: Todo artefacto que se utiliza en el ciclo de desarrollo de software para obtener el producto. • Ejecución: Test cases. • Actualización: Story Mapping. • Actualización: Scrum Board. • Refinamiento y actualización: Product Backlog. • Formulario: Acta de seguimiento.
13	<p>Análisis de resultados del Proyecto Integrador: validación y pruebas, verificación, solución de problemas encontrados, revisión del cumplimiento en tiempo y recursos presupuestales y estimación del impacto.</p> <p>Herramientas: Editores de código, entornos integrados de desarrollo, editores de modelos, editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y control de versiones.</p>	<ul style="list-style-type: none"> • Sprint Review: Lanzamiento final del producto al 100%. • Retrospectiva del Sprint. • Sprint Burndown Chart. • Sprint Backlog. • Story Mapping. • Product Backlog. • Entregables del proyecto: Todo artefacto que se utiliza para gestionar el proyecto. • Entregables del producto: Todo artefacto que se utiliza en el ciclo de desarrollo de software para obtener el producto. • Formulario: Acta de seguimiento. <p>Importante: En esta semana, los entregables deben alcanzar el 100% de desarrollo del producto y/o ejecución del proyecto.</p>
14	<p>Estrategias de divulgación de resultados del Proyecto Integrador: Desarrollo del acta de entrega del proyecto, presentación oral, póster, informes, bitácoras, ensayos, artículos, videos y recursos multimedia.</p> <p>Herramientas: Editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y editores de video.</p>	<ul style="list-style-type: none"> • Validación técnica del producto. • Pre - sustentación: Socialización de la presentación en el formato para la Jornada Socialización de proyectos integradores y/o aula. • Acta de entrega y cierre del Proyecto Integrador.
15	<p>Estrategias de divulgación de resultados del Proyecto Integrador: Presentación oral, póster, informes, bitácoras, ensayos, artículos, videos y recursos multimedia.</p> <p>Herramientas: Editores de texto, hojas de cálculo, diapositivas, plantillas, bases de datos bibliográficas, aplicaciones para manejo de referencias, software para el apoyo en el desarrollo de los proyectos y editores de video.</p>	<ul style="list-style-type: none"> • Presentación Jornada de Socialización de proyectos integradores y/o aula.
16	<p>Cierre del Proyecto Integrador: Discusión, lesiones aprendidas y conclusiones.</p>	<ul style="list-style-type: none"> • Retrospectiva del Proyecto Integrador. • Nuevo cuerpo del conocimiento. • Informe final.

Tabla 1. Desarrollo del trabajo académico



Régimen de calificaciones

A continuación, se define el régimen de calificaciones. Algunos entregables tienen diferentes versiones y se evaluarán en los dos cortes de seguimiento.

Entregables	1° corte (%)	2° corte (%)
Definición y propuesta del Proyecto Integrador		
Análisis y Modelo UML del Dominio.	2	
Presentación (5 minutos): Introducción a la definición del problema.	1	
Propuesta del Proyecto Integrador.	1	
Project Charter.	2	
Análisis de requerimientos y planificación de solución		
Story Mapping (iterativo).	2	1
Product Backlog (iterativo).	4	2
Sprint Backlog (iterativo).	2	4
Diseño y desarrollo del Proyecto Integrador		
Daily Scrum y Scrum Board. (Mínimo tres reuniones a la semana de máximo 15 minutos - iterativo).	1	2
Diseño: Modelos de infraestructura, estructura, comportamiento, datos y casos de prueba (iterativo).	2	4
Test cases (iterativo)	1	2
Análisis de resultados		
Sprint Review y retrospectiva (iterativo)	2	4
Acta de seguimiento (iterativo) + Acta de entrega y cierre del Proyecto Integrador	1	1
Actividades de aprendizaje (evaluaciones, talleres, trabajos, ejercicios, exposiciones, bonos, etc.)	2	2
Estrategias de divulgación de resultados		
Pre - sustentación	2	3
Presentación Jornada de Socialización de Proyectos Integradores y/o aula		5
Seguimiento	25	30
Primer parcial = Producto 5% + Informe de avance 15%	20	
Segundo parcial = Producto 20% + Informe final 5%		25
Total	45	55

Tabla 2. Régimen de calificaciones

La asistencia a clase es obligatoria, y será verificada en cada clase. Un índice de fallas superior al 20% conduce a la pérdida de la materia, conllevando a una calificación de 0,0.



Equipos de trabajo

	Estudiantes		Programa
	Id	Nombre	Equipo
18	000472655	LIZARAZO ORTIZ KAREN NICOL DAYANNA	Serpiente
26	000429178	VASQUEZ BALLESTEROS IVAN SANTIAGO	
6	000472029	LONDOÑO GOMEZ SAMIR JOSEPH	
4	000462032	BECERRA HERNANDEZ JUAN SEBASTIAN	
3	000432240	ATUESTA TORO JOSE MANUEL	
14	000467829	HERNANDEZ ARMENTA JUAN SEBASTIAN	
2	000452246	ARANGO OLAYA ANDRES FELIPE	Tigre
28	000467654	DIAZ CARDENAS FERNEY ANTONIO	
19	000474286	BERNAL ESCALANTE ALEXANDRA	
5	000462163	BERARDINELLI ACOSTA GIOSUE	
23	000488584	REYES SOLANO JOSE GABRIEL	
31	000476838	NAJM MILGEM GEORGIO	
25	000476297	SALAS AQUINO SOFIA ALEJANDRA	Conejo
21	000453006	MORENO CABALLERO OSCAR LUIS	
15	000448378	JAGUA CARVAJAL SANTIAGO	
17	000464957	LEON SANMIGUEL ANDRES FELIPE	
9	000474833	CARVAJAL MOGOLLON DIDIER SANTIAGO	
24	000349755	RODRIGUEZ SANTANDER BRAYAN ANTONIO	
11	000450441	GAMEZ FLOREZ ANDRES FELIPE	Dragón
13	000460746	GRANADOS MORENO JAVIER MAURICIO	
8	000426440	CARDENAS RIVERA DAVID SANTIAGO	
1	000450420	ADAMES SOLARTE JUAN PABLO	
27	000467411	VILLABONA APONTE SANTIAGO ENRIQUE	
32	000374600	BELEÑO CARRASCAL NATALIA CAROLINA	
33	000476102	VALENCIA ROJAS JULIAN ANDRES	
16	000428515	JAIMES MUÑOZ JUAN PABLO	Mono
7	000465699	CADAVID NOGUERA NICOLLE FABIANA	
12	000448376	GOMEZ RODRIGUEZ JUAN CAMILO	
14	000429178	VERA MONTAÑA JIMMY ANDRES	
20	000432059	MARIN DIAZ EDWARD DANIEL	
22	000472269	QUINTANA REYES JUAN DIEGO	

Tabla 3. Equipos de trabajo



THE NEXUS BATTLES II

SISTEMA DE COMERCIO ELECTRÓNICO

SOBRE INFRAESTRUCTURA EN LA NUBE

1. Situación problema

El grupo empresarial familiar UPB-COMPANY ha decidido poner en marcha un nuevo proyecto llamado THE NEXUS BATTLES II. Para ello, ha creado un juego de cartas coleccionables e intercambiables. El proyecto tiene como objeto desarrollar un servicio web que haga uso de un modelo de inteligencia artificial soportado por infraestructura en la nube para comerciar e intercambiar cartas virtuales, además, realizar torneos y juegos online de contiendas multijugador o jugador contra la máquina.



Nota del proyecto: *THE NEXUS BATTLES II* es un juego diseñado por el **Doctor Bonny Hamir Mantilla Pérez** y adaptado por el docente **Lenin Javier Serrano Gil** para el curso de Proyecto Integrador. Se debe respetar todos los derechos asociados y utilizar recursos libres como tipografía, imágenes, iconos, entre otros, para su implementación.

1.1. Descripción del Juego

THE NEXUS BATTLES II es un juego de cartas en que cada jugador dispone de un mazo o pila de treinta cartas (30). Además de una carta especial que corresponde a un personaje, comandante o héroe con quien vivirá la aventura. Las treinta cartas aportarán estadísticas, mejoras y habilidades con las que el héroe puede realizar sus ataques o resistir los embates en su contra.

UPB-COMPANY ha desarrollado cuatro prototipos de héroes con estadísticas propias clasificados como:

- El “Guerrero Tanque”, este personaje se caracteriza por aguantar en gran medida los ataques del contrario, pero el daño que causa es reducido.
- El “Guerrero Armas”, también es un guerrero, pero su especialidad es causar todo el daño posible sacrificando su defensa.
- El “Mago Fuego”, entrenado en el arte de la magia elemental, es un personaje que puede causar gran daño al oponente.
- El “Mago Hielo”, contra parte del Mago Fuego, se especializa en disminuir el poder del enemigo para conseguir una fría victoria.
- El “Pícaro Veneno”, entrenado en alquimia puede causar gran daño al adversario afectando sus habilidades y reduciendo la posibilidad de que puedan acertar un golpe.
- El “Pícaro Machete”, entrenado en armas filosas es un experto en infligir heridas mortales.

Cada héroe cuenta con un punto de “Poder” que le permite tomar acción en un turno y cuatro estadísticas base, entre ellas: “Vida”, representa los puntos de vitalidad del héroe. “Defensa”, representa la capacidad



de esquivar o resistir un ataque. “Ataque”, representa la probabilidad de acetar un golpe al oponente. Y “daño”, es el valor probable efectivo que resta vida a causa de un ataque.



Nota del proyecto: Para el negocio los personajes descritos junto con sus estadísticas son el punto de partida, se espera se desarrollen más cartas de personajes e ítems a medida que el negocio se desarrolle en el tiempo.

Los valores de las estadísticas por tipo de héroe se presentan en la Tabla 4. Estadísticas para los héroes o comandantes.

	<i>Guerrero</i>		<i>Mago</i>		<i>Pícaro</i>	
<i>Tipode héroe</i>	Tanque	Armas	Fuego	Hielo	Veneno	Machete
<i>Poder</i>	1	1	1	1	1	1
<i>Vida</i>	24	24	20	20	16	16
<i>Defensa</i>	11	11	10	10	8	8
<i>Ataque</i>	10 + 1d6	10 + 1d6	10 + 1d8	10 + 1d8	10 + 1d10	10 + 1d10
<i>Daño</i>	1d4	1d4	1d6	1d6	1d8	1d8

Tabla 4. Estadísticas para los héroes o comandantes.

Por otra parte, las treinta (30) cartas adicionales otorgan la posibilidad de adquirir características y mejoras al combinarlas con los héroes. Entre las cartas hay cuatro (4) grupos “Armas”, “Armaduras”, “Ítems” y “Habilidades especiales o Épicas”.

“Armas”, equipan a los héroes con características que incrementan sus estadísticas para hacer daño o defenderse. Se observa los efectos de las armas por héroe en la Tabla 5. Set de armas para cada héroe.

	<i>Guerrero</i>		<i>Mago</i>		<i>Pícaro</i>	
<i>Tipode héroe</i>	Tanque	Armas	Fuego	Hielo	Veneno	Machete
	SET 1					
<i>Armas</i>	Espada de una mano	Espada de dos manos	Orbe de manos ardientes	Báculo de Permafrost	Daga purulenta	Machete vendido
<i>Efectos</i>	+1 al ataque	+1 al ataque	+1 al daño	-1 al daño del oponente	+1 al daño por dos turnos	+2 al daño
	SET 2					
<i>Armas</i>	Escudo de dragón	Piedra de afilar	Fuego fatuo	Venas heladas	Visión borrosa	Cierra sangrienta
<i>Efectos</i>	+1 a la defensa	+2 al daño	+1 al ataque	+1 al daño	-1 al ataque del oponente	+2 al daño por dos turnos

Tabla 5. Set de armas para cada héroe.

“Armaduras”, equipan a los personajes en defensa y pueden agregar características especiales que pueden mejorar su ataque. Las armaduras se describen catalogadas por héroe en la Tabla 6. Armadura para cada héroe.

	<i>Guerrero</i>		<i>Mago</i>		<i>Pícaro</i>	
<i>Tipode héroe</i>	Tanque	Armas	Fuego	Hielo	Veneno	Machete



	GEAR 1					
	Pesada	Media	Tela	Tela	Media liviana	Media Liviana
Armaduras						
Efectos	+4 a la defensa	+3 a la defensa	+1 a la defensa	+1 a la defensa	+2 a la defensa	+2 a la defensa

Tabla 6. Armadura para cada héroe.

“Ítems”, son objetos que mejoran o debilitan las habilidades del héroe. También afectan las características de las armas y/o armaduras y pueden extender en algunos casos efectos al oponente. Así, los ítems se describen asociados a cada héroe en la Tabla 7. Ítems para cada héroe.

	Guerrero		Mago		Pícaro	
	Tanque	Armas	Fuego	Hielo	Veneno	Machete
Tipode héroe	EQUIPMENT 1					
Ítems	Pinchos de escudo	Empuñadura de Furia	Anillo para Piro-explosión	Libro de la ventisca helada	Veneno lacerante	Mancuerna yugular
Efectos	Si el ataque del oponente es menor que la defensa del guerrero. El oponente recibe +1 de daño	+1 de daño por dos turnos. Esto causa que el guerrero pierda -1 de vida en los mismos turnos.	+3 de daño	+2 de daño	-1 al poder de oponente. Solo aplica cada dos turnos.	Explota por 2 el valor en turno causado por la cierra sangrienta en el oponente

Tabla 7. Ítems para cada héroe.

“Habilidades especiales o Épicas”, son cartas especiales que otorgan habilidades al ataque del héroe con la particularidad de incluir mejoras, si coincide con su tipo. Por ejemplo, La Épica llamada "Segundo Impulso" otorga a todos los héroes una recuperación de salud en un rango aleatorio entre 1 a 4 puntos. Además, si el héroe es un Guerrero Armas gana el efecto épico solo para el tipo de héroe, que le aumenta la vida en 3 puntos. Se puede observar las épicas correspondientes a cada tipo de héroe en la Tabla 8. Habilidades especiales o Épicas.

	Guerrero		Mago		Pícaro	
	Tanque	Armas	Fuego	Hielo	Veneno	Machete
Tipode héroe						
Épicas	Golpe de defensa	Segundo impulso	Luz cegadora	Frio concentrado	Toma y lleva	Intimidación sangrienta
Efecto para todos	+ 1 al ataque	Recupera 1d4 de vida	+1 a la vida	-1 de poder al oponente	+1 al ataque	+1 al daño
Efecto épico solo para tipo de héroe	+4 al daño	+3 a la vida	No recibe ningún daño en el siguiente turno	+2 al daño	Disminuye a la mitad del daño causado por el oponente y se lo retorna	+2 a la vida

Tabla 8. Habilidades especiales o Épicas.

Para que un jugador participe en un juego, debe comprar un mazo. Esto debería incluir dos cartas de armas, la carta de armadura, una carta de ítem, cinco cartas épicas y una carta épica de tipo héroe. El sistema se encargará de completar y mezclar las treinta cartas (30). Para ello, se incluyen en el mazo dos (2) cartas de armas multiplicadas por cuatro (4), las cartas de armadura multiplicadas por cuatro (4), cartas de ítem multiplicadas por cuatro (4), que se mezclan para sumar dieciséis cartas (16). Asimismo, cinco (5)



cartas épicas por dos (2) para diez (10). Estas últimas o "épicas" no incluyen las cartas que otorgan mejoras por tipo de héroe. Y para terminar el mazo cuatro (4) cartas épicas tipo héroe. Para treinta (30) cartas en la baraja.

THE NEXUS BATTLES II se juega por turnos, todos los turnos son iguales para los jugadores. El primer turno es el turno uno para cada miembro del grupo, el segundo turno dos para cada miembro del grupo, y así hasta el final. Para ejecutar el juego, el jugador presenta a su héroe y recibe dos cartas de su mazo. Después de recibir una tercera carta, el jugador mejorará su personaje y realizará un ataque. Puede realizar el ataque sin consumir una de las tres cartas y en la siguiente ronda podrá usarla o cambiarla. Los siguientes jugadores realizan el mismo procedimiento hasta el final de la partida. El orden en el turno se dará por sorteo y se seguirá durante todo el juego. Si el juego es cooperativo. Por ejemplo, dos jugadores en pareja contra dos jugadores no podrán hacerse daño entre los héroes del mismo equipo a menos que exista un ataque que los obligue. El juego termina cuando solo queda un héroe o equipo con vida.



Nota del proyecto: Cada estudiante representará un rol en el grupo de desarrollo con sus responsabilidades derivadas. Un estudiante puede ser expulsado de un grupo si, tras tres llamados de atención por parte del docente, con justa causa y de común acuerdo, sus compañeros lo denuncian porque no responde a sus tareas o su desempeño es muy bajo, tanto que perjudica el desempeño del grupo.

2. Requerimientos del producto

UPB-COMPANY ha nombrado a un Ingeniero Informático de total confianza para que gobierne las decisiones acerca de que debe tener, como se debe diseñar, desarrollar e implementar el servicio. El ingeniero nombrado ahora como el Cliente, ha redactado los siguientes capítulos expresando los requisitos del sistema.



Nota del proyecto: El escenario ha sido diseñado para grupos de mínimo seis (6) personas. De acuerdo con la descripción y situación del problema, cada grupo deberá diseñar e implementar los **módulos del servicio y/o sistema de información**, así como la **infraestructura en la nube** necesaria para el despliegue y control del sistema. Los sistemas operativos por utilizar son de "libre elección" junto con la base de datos y arquitectura para el despliegue. **Atención**, todo diseño, sugerencia de compra e implementación de software y/o infraestructura debe ser estudiado, justificado y presentado para **aprobación del cliente**.

2.1. Módulo para Carro de compras (E-commerce)

UPB-COMPANY solicita un módulo de comercio electrónico con carro de compras para la comercialización de las cartas. El módulo debe contar con una vista web o la vitrina de productos con un banner de presentación y una distribución responsive para seis (6) ítems. Los ítems deben distribuirse en una pantalla de resolución mínima de 1360 x 768 píxeles o página. Cada página debe mostrar los productos alineados en tres columnas y dos filas. Consultar la Ilustración 1. Prototipo para la vitrina de productos. Si la cantidad de los productos superan la vitrina deberán generarse tantas paginas como se necesiten. Las páginas deben ser navegables a través de un control de paginación. El control debe permitir que, si hay

una página, aparece la casilla 1, si hay dos, aparecen las casillas 1 y 2, y así sucesivamente hasta llegar a la casilla número 5. Si hay más páginas a la izquierda, aparece una flecha a la izquierda para mostrar las opciones ocultas y del mismo modo a la derecha.

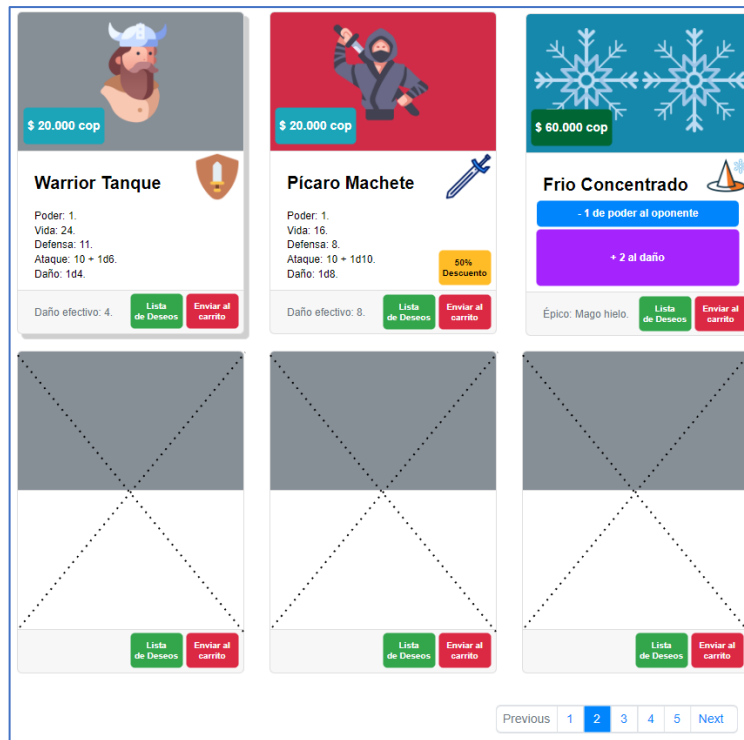


Ilustración 1. Prototipo para la vitrina de productos.

La información por producto corresponde con el diseño de la carta del juego, en ella se expone, nombre del producto, imagen, descripción, habilidades, y precio (COP o dólar o euro dependiendo de su ubicación geográfica). Los productos en promoción deben tener un marcador que identifiquen el porcentaje de descuento. Se espera que cada producto sobresalte al pasar el ratón sobre él. Además, en el área de cada producto se debe incluir un botón para añadir a la cesta de compra y añadir a la lista de deseos. Los productos que son enviados a la lista de deseos siempre tendrán una distinción para que el cliente los tenga presente. De igual manera los productos que el cliente ha adquirido estarán marcados como propios. La vitrina debe estar dotada de funcionalidades de búsqueda para cualquier información presente en la carta de juego incluido el precio. Además, debe filtrar las cartas del juego por precio y/o tipo y/o cartas en promoción. La búsqueda general debe utilizar algoritmos de indexación que permitan identificar en pocos caracteres los productos que el cliente está buscando. Si el cliente clicca sobre un producto se debe dirigir a una vista de detalle donde está la información de la vitrina junto con la calificación promedio e individual valorada en cinco estrellas, incluyendo el hilo de comentarios acerca del mismo. Los comentarios permiten incluir texto e imágenes, así como el apodo del cliente, la calificación en estrellas y la fecha en que se publicó. Los usuarios solo pueden calificar un producto una vez, pero podrán agregar o retirar tantos comentarios como sea de su agrado. Los botones de agregar un producto al carro de compras o a la lista de deseos siempre deben estar disponibles en todas las vistas.



Se debe tener una barra de navegación en la parte superior donde se incluya la búsqueda de los productos, los accesos al menú de “Jugar online”, al menú “Torneo”, al menú “Mi banco”, al menú de “Subasta”, al menú de “Mi Cuenta” y al resumen flotante de “Mi carrito de compra”.



Nota del proyecto: El sistema debe diseñarse e implementarse utilizando el **patrón de microservicios**. Cada servicio debe ser autónomo, especializado, elástico y resiliente ante el funcionamiento del sistema. Así mismo, el ciclo de desarrollo debe ajustar **el proceso de desarrollo orientado por pruebas (Test-driven development) e integración y distribución continua (CI/CD)** para desplegar los microservicios mediante el uso de automatización.

El icono del menú resumen para el carro de compras debe mostrar la cantidad de productos que se ha agregado. El menú “Jugar online” permite, jugar una partida entre dos o más jugadores o jugar una batalla entre dos o más jugadores incluida la IA. El menú “torneo” enlaza una vista para la inscripción, participar, consulta de los equipos, entrar al “live-stream” si están jugando en este instante o a los juegos anteriores “video-stream”. El menú “Mi banco” es donde se muestran las cartas de juego que se ha adquirido. En el menú “Subasta” se despliega las opciones de comprar, vender o recoger la venta de cartas. Para la subasta se utiliza una moneada ficticia (créditos) que únicamente se puede obtener jugando. Por ejemplo, dos (2) créditos por ganar un juego uno vs uno. El menú “Mi cuenta” muestra, si el usuario se encuentra registrado en el sistema, las opciones para control del perfil del cliente y cambio de contraseña o de lo contrario una única opción al registro. El usuario puede recuperar su cuenta contestando preguntas con respuestas previamente configuradas y el código enviado a su correo electrónico. El resumen flotante del carro de compras debe contener los productos que se han agregado en cualquier periodo de tiempo y los lista con imagen, nombre, precio, así como el total bruto, impuestos y total neto. Desde el resumen flotante se podrá ir al carro de compras o directamente a realizar la compra. Un nuevo usuario o cliente se registra con la información personal nombres, apellidos, email, contraseña, apodo, avatar y medio de pago. Las contraseñas deben ser mayores a ocho (8) caracteres y contener entre caracteres alfabéticos, mayúsculas, minúsculas, números y símbolos. El apodo no podrá contener palabras ofensivas o nombres respetados o reconocidos políticos, celebridades, dirigentes, o marcas registradas entre otros. El carro de compras es una vista que se recupera desde el botón “Ir al carrito”. Muestra los productos y la información de la venta. También, agrega la opción de eliminar productos. Además, incluye el botón “Realizar pedido” y el enlace para volver a la tienda “¿Quieres añadir más productos?”. Un usuario registrado puede adquirir cartas del juego pagando en línea con tarjeta de crédito o PayPal, ofrecer cartas del juego en la subasta, si posee los créditos para cubrir las comisiones, adquirir nuevas cartas en la subasta si tiene créditos para ello, pagar para jugar partidas individuales, o pagar por partidas en grupo con descuento de 6 partidas, 12 partidas, o 24 partidas, o mediante una suscripción mensual ilimitada de partidas. El sistema debe garantizar que cumple con la ley de protección de datos. Además, garantizar la protección de los bienes digitales adquiridos por los clientes, hasta que cierren sus cuentas o hasta que NEXUS BATTLES II suspenda su actividad comercial, y los bienes digitales se entregarán a los clientes en un documento XML o JSON.



2.2. Módulo de administración e inventario de cartas

El sistema proporciona un módulo de inventario y administración de cartas. En este módulo, los usuarios autorizados tendrán acceso al inventario de cartas para crear nuevas cartas, modificar o “eliminar” las existentes. En el caso de eliminar cartas, se habla de suspender temporalmente la adquisición en la tienda, pero seguirán existiendo en el inventario del cliente si se compraron antes. Todos los atributos como imagen, nombre, descripción, habilidades entre otros se pueden editar y afectarán a todo el inventario de los usuarios. Al crear cartas se debe incluir cuantas habrá en el inventario, para que el tiraje de cartas sea limitado y así ganen valor en la subasta. Las cartas con un parámetro, por ejemplo, “-1” serán infinitas, es decir siempre se podrán vender. Además, habrá cartas que solo se podrán comercializar en la subasta y/o en el e-commerce. También, el módulo debe incluir un diseñador para el aspecto de la carta que permita crear o cambiar el aspecto de los productos. Los cambios se informarán a los usuarios a través de alertas al iniciar sesión o en el banner de la vista principal con una descripción y fecha clara.

2.3. Módulo de usuarios y comentarios

En el módulo de usuarios y comentarios se permite a un usuario autorizado gestionar y dar de baja a usuarios y comentarios. Los usuarios o clientes solo son creados por ellos mismos. Usuarios de administración son creados por administradores. Los usuarios pueden ser baneados temporal o definitivamente si faltan a las políticas de la compañía o porque se ha terminado la relación legal entre las partes.

2.4. Módulo de Chatbot

El objetivo principal del Chatbot es ofrecer una experiencia de atención al cliente eficiente y rápida, brindando respuestas precisas y útiles a las preguntas de los usuarios. El Chatbot es un sistema de inteligencia artificial diseñado para brindar respuestas y asistencia a los usuarios en sus consultas relacionadas con la plataforma. Deberá programarse para comprender y responder preguntas comunes sobre productos, pagos, reglas del juego y cualquier otro aspecto relacionado con la experiencia en el sistema. Los usuarios podrán usar el Chatbot en cualquier momento y/o vista dentro del sistema.

2.5. Módulo de correo electrónico

El sistema debe proveer un módulo de correo electrónico para comunicación con el cliente, confirmar la creación de una cuenta de usuario, recuperar contraseñas, enviar mensajes publicitarios del e-commerce, así como enviar mensajes del módulo de la subasta. Todo mensaje debe utilizar una plantilla que cumpla con las políticas de la compañía.

2.6. Módulo para Jugar online

El menú "Jugar online" le permite ingresar al módulo para jugar una partida entre dos o más jugadores o jugar una batalla cooperativa o versus entre dos o más jugadores, incluida la IA. El módulo muestra una vista en la que el jugador puede crear una sala de batalla o ingresar a una batalla de una lista creada por otros jugadores. Para poder participar en un juego, el jugador en su banco de cartas debe tener las cartas mínimas para crear la baraja. Si elige crear un juego, debe seleccionar cuántos jugadores participarán en él, la posible recompensa y si se incluye un héroe aleatorio controlado por la IA. Los partidos pueden ser uno contra uno, uno contra IA o en mesas de hasta seis (6) jugadores donde cualquiera puede ser controlado por la IA. Las reglas y el método del juego ya se han descrito anteriormente y así es como debe

implementarse. Se adjunta una imagen que muestra la distribución y presentación mínima de las cartas en el modo de juego. Ver la Ilustración 2. Cartas de los héroes en modo juego. Las cartas deben tener la siguiente interacción, un botón o similar para realizar las mejoras, las mejoras se obtienen de las tres cartas en juego, para esto se despliega un menú para elegir la mejora entre las cartas disponibles. Un botón o similar para pasar en su turno, puede pasar antes o después de hacer la mejora. Un botón o similar para realizar el ataque, este siempre debe estar dirigido a uno de los oponentes en el juego. El valor de daño real para cada ataque se muestra en la parte inferior. Además, junto a la imagen, se muestra una barra de progreso que muestra el estado de vida con su valor y en tres colores, verde si está por encima del 60%, amarillo si está entre el 60% y el 40%, y rojo si cae por debajo del 40%. La bolsa que acompaña a la imagen es un botón o similar que muestra una lista flotante de las mejoras que se le han aplicado al héroe con sus valores.

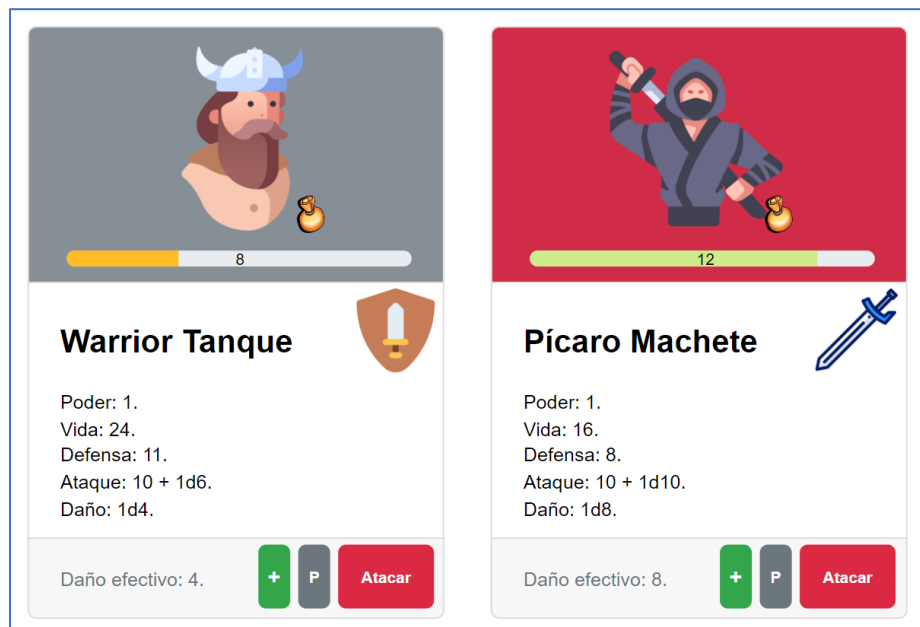


Ilustración 2. Cartas de los héroes en modo juego.

Las salas de batalla, así como la vista general, deben tener un sistema de chat donde los jugadores puedan comunicarse para organizar nuevos juegos o chatear en medio de uno. Los jugadores son recompensados en todos los partidos. Al final de cada partida, el jugador ganador recibe dos (2) créditos si la batalla es uno contra uno o cuatro (4) créditos si es una batalla grupal, los demás jugadores deben conformarse con un (1) crédito por participar. Cuando un jugador logre acumular veinte (20) créditos en juegos ganados recibirá un sobre con una tarjeta de juego aleatoria cada vez que se cumpla la cuota. Este beneficio solo podrá obtenerse dos veces por semana y el contador se reinicia cada vez que complete los créditos. También, los jugadores de manera libre podrán poner en juego sus créditos en las batallas o utilizarlos para negociar en la subasta. Con respecto a la IA, el sistema debe valerse de algoritmos de aprendizaje automático que le permitan a la IA establecer estrategias de juego a partir de los datos almacenados de todas las partidas. A continuación, se muestra el borrador del campo de batalla. Ver la Ilustración 3. Campo de batalla para cuatro jugadores. Vista desde el Guerrero Tanque. En el campo debe incluir los

marcadores que informan el turno y el mazo donde se obtiene las cartas en cada turno, en la ilustración representada por el rectángulo rayado en color verde.

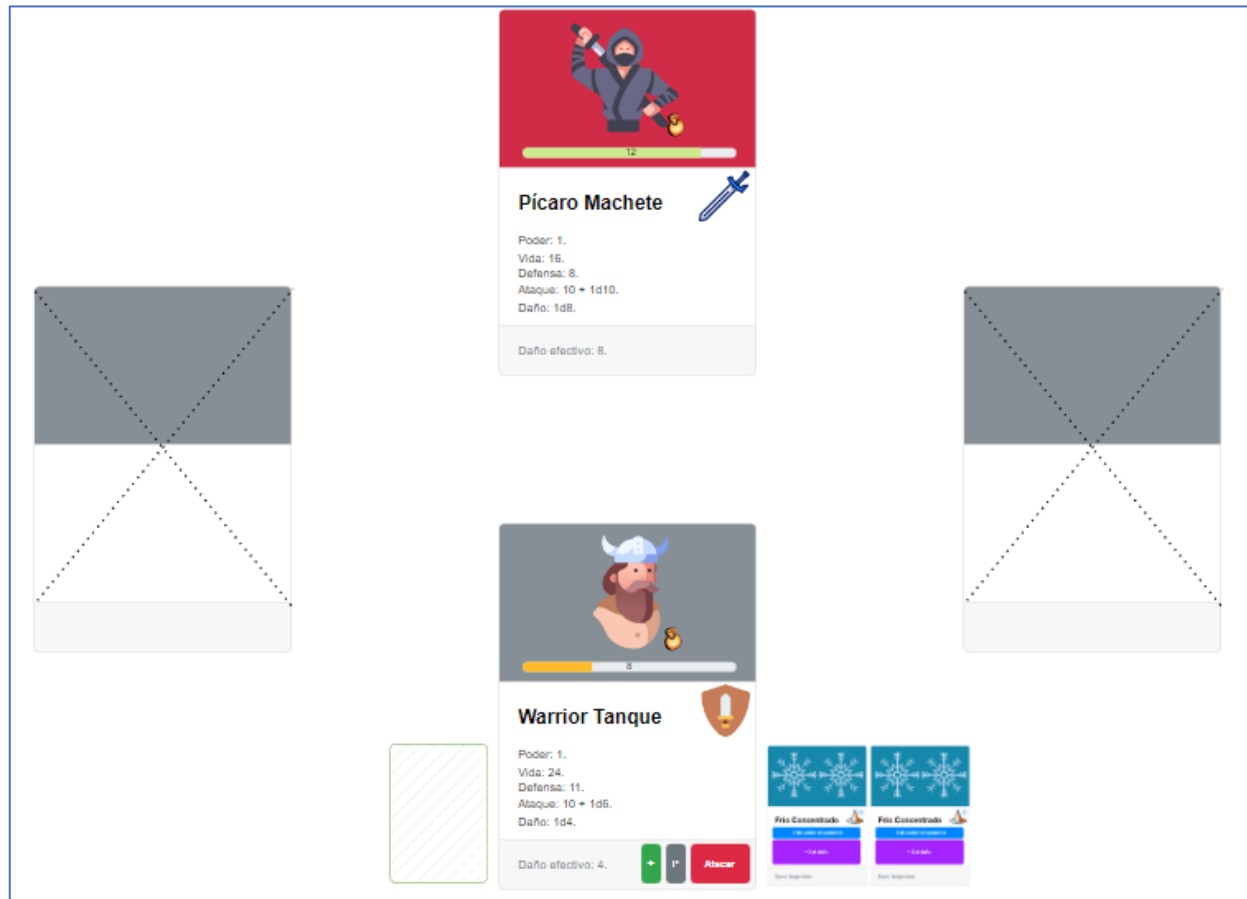


Ilustración 3. Campo de batalla para cuatro jugadores. Vista desde el Guerrero Tanque.

2.7. Módulo del Torneo

El módulo de torneo permite inscribirse a una competencia en una fecha particular. El torneo se juega en equipos de dos jugadores con un máximo de cuatro (4) jugadores por mesa. Para participar en el torneo se debe pagar una inscripción que puede ser en dinero real o créditos del juego. El árbol del torneo tiene una longitud de ocho (8) equipos de jugadores o clientes. Si no se completan los equipos de jugadores, los remplazarán equipos de la IA. Los equipos tendrán un nombre y un avatar que los identifique. El torneo se desarrolla con la siguiente mecánica: Al inicio todos los equipos estarán en el mismo árbol. Hay dos árboles, el árbol de ganadores y el árbol de secundarios como lo muestra la Ilustración 4. Árbol del Torneo. Los equipos ocuparán los lugares y recorrerán el torneo de acuerdo con sus victorias o derrotas. El árbol de ganadores tiene los encuentros (E): 1, 2, 3, 4, 5, 6, 11, y final. El árbol de secundarios tiene los encuentros (E): 7, 8, 9, 10, 12 y 13. El módulo de torneo permite la inscripción, pago, jugar, la administración de los árboles, la gestión de los equipos, el live-stream o video-stream de las justas acompañadas de los comentarios de los presentadores. La competencia es automática y el grupo ganador

recibirá créditos y una carta épica única para su héroe. Solo administradores de torneo podrán crear las justas y solo se permitirá un torneo cada noventa y un (91) días.

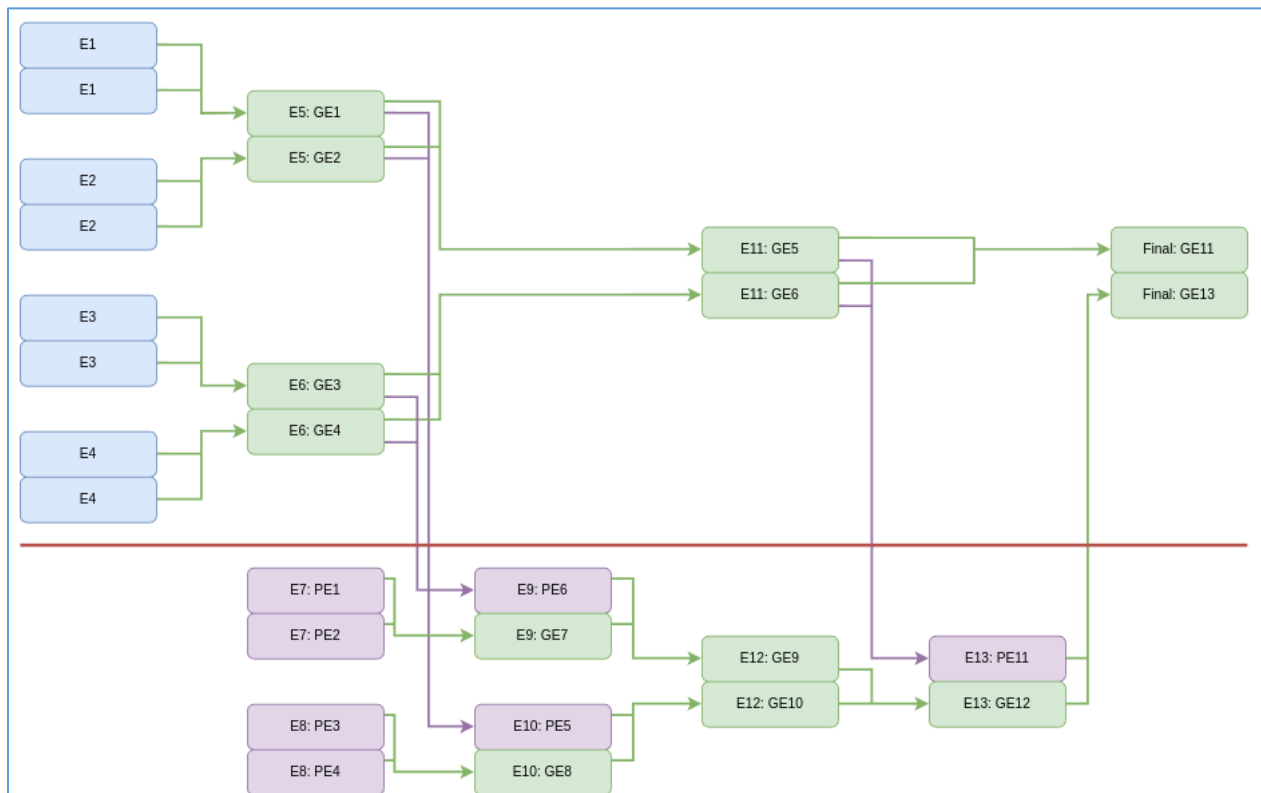


Ilustración 4. Árbol del Torneo.

2.8. Módulo de Subasta

En THE NEXUS BATTLES II la subasta es un módulo muy importante porque permite a los jugadores intercambiar cartas usando los créditos ganados en las batallas. El sistema de subasta consiste en que un jugador publica en modo subasta las cartas que desea canjear. Las cartas estarán disponibles en la subasta en periodos de veinte y cuatro (24) horas o cuarenta y ocho (48) horas. Para cada opción de horaria, la subasta cobra una comisión de un (1) crédito por las veinte y cuatro horas y tres (3) créditos por las cuarenta y ocho horas. En la puja se puede fijar un valor mínimo, es decir, en el periodo de tiempo se acepta que los participantes hagan pujas desde un valor hasta el final del tiempo. También si es urgente se puede usar la modalidad de compra inmediata con el valor fijo superior de la puja. El valor de intercambio puede pedir como pago una o varias cartas y/o créditos. UPB-COMPANY podrá participar en la subasta con ítems que no se pueden comparar en el carro de compras. Esto quiere decir que la compañía tiene un usuario maestro de juego que puede agregar novedades en cualquier momento. Solo usuarios registrados pueden utilizar la subasta siempre y cuando tengan créditos para las transacciones. El módulo de subasta debe tener una vista donde se encuentre la lista de las cartas, los filtros de búsqueda



similares al carro de compra y las vistas para configurar las pujas, recoger lo vendido o comprado. El sistema le notificara al ganador al finalizar el tiempo. Es importante mostrar el detalle de las cartas y que el módulo sea de “fácil uso” para los jugadores.

2.9. Infraestructura

UPB-COMPANY apuesta por una infraestructura en la nube, con un respaldo periódico on-premise para sus datos. Se solicita presentar el diseño de infraestructura antes del primer mes junto con el presupuesto para una operación mensual del sistema. El diseño debe incluir microservicios, APIs, definición y características de bases de datos, definición y características de protocolos de comunicación, balanceo de carga para los servicios, diseño de red, diseño de escalabilidad del sistema y recuperación ante desastres. El sistema debe cumplir con una resiliencia del 99,95%. El módulo de e-commerce debe utilizar una base de datos relacional. El módulo de inventario deberá usar bases de datos no relacionales. El módulo del juego debe utilizar una base de datos en memoria. El análisis de IA debe usar sistemas apropiados para almacenar toda la información de los partidos. La gestión de perfiles de usuario y credenciales debe contratarse con el directorio activo. Se espera un flujo de cien mil (100.000) usuarios simultáneos. Además, el sistema debe adaptarse si su capacidad de procesamiento supera el 70%. La meta de la compañía es manejar un volumen de un millón (1.000.000) de usuarios en la primera mitad del año, con latencias inferiores a quinientos (500) milisegundos. Se deben entregar diseños, informes de pruebas de latencia y carga.

3. Requisitos del proyecto

El proyecto se considera completo cuando se hace entrega de: **PRIMERO.** La totalidad de la implementación de los requisitos funcionales y no funcionales descritos en el acta de constitución del proyecto (Project Charter), backlog o similar. **SEGUNDO.** El diseño, la implementación de la infraestructura y escenario de comunicaciones con todos los servicios y protocolos solicitados. **TERCERO.** El informe final en formato IEEE [<https://www.ieee.org/conferences/publishing/templates.html>] acompañado de la totalidad de la documentación derivada del proyecto y el proceso de desarrollo del software. El documento debe contener como mínimo: Título. Introducción. Estado del arte y marco conceptual. Objetivos. Metodología. Resultados del producto y del proyecto según cada componente del integrador. Reporte de avance del cronograma (coherente con la bitácora o seguimiento). Análisis de desempeño del equipo. Análisis de la retrospectiva del proyecto. Discusión. Conclusiones. Trabajo futuro. Agradecimientos. Todas las referencias bibliográficas actualizadas, relevantes, académicas o de fuentes notables. Además, se debe adjuntar como anexos, Diseños, Tablas, Documentos, Evidencias, Documento de especificación de requerimientos, Documentos de diseño y ejecución de pruebas, Código fuente debidamente documentado, Product Backlog, Sprint Backlogs, Story Mapping, Scrum Board, Actas de seguimiento, Retrospectivas del Sprint, Sprint Burndown Charts, entre otros. Así también el manual de usuario, el documento de política de seguridad y el procedimiento de recuperación del sistema. **CUARTO.** La realización de la presentación en la Jornada de Integración y Proyectos de Aula. **QUINTO.** La firma del acta de entrega del producto o similar de acuerdo con el cumplimiento de los criterios de aceptación y a conformidad de las partes.





Nota del proyecto: En el proceso de análisis, diseño, desarrollo e implementación, se deben contemplar aspectos de **desarrollo de software seguro**, así como un **proceso basado en pruebas (TDD)** para la verificación del funcionamiento del sistema.

Se debe como mínimo presentar los documentos de gestión del proyecto, un diagrama de estructura, un diagrama de comportamiento, un diagrama de componentes, un diagrama de despliegue o arquitectura del sistema, el modelo relacional o no relacional de las bases de datos con su diccionario de datos, el diagrama de la infraestructura de red y los archivos de configuración de los servicios y dispositivos.

Se debe tener en cuenta el uso de herramientas de apoyo para la gestión y seguimiento de proyectos de desarrollo de software, tales como: GitHub, GitLab, One-drive, Drive, Trello, Jira o similares.

El uso de **Jira Software** [<https://www.atlassian.com/software/jira>] para la gestión del proyecto es obligatorio. Todas las entregas deben agruparse en un **zip** de acuerdo con las fechas de entrega. Los documentos siempre deben ser impresos en **PDF** y deben adjuntar un **acta** con la **evaluación porcentual** acerca de la **completitud de las tareas de los miembros del equipo**.

Los informes finales se publicarán en un sitio web proporcionado por la Facultad de Ingeniería de Sistemas e Informática. Los mejores tendrán la opción de ser asesorados por el profesorado para una posible presentación en congresos y / o revistas científicas.

4. Gestión del Proyecto Integrador y desarrollo del Producto

Para la gestión del proyecto y desarrollo del producto se seguirá la metodología de Proyectos Integradores de la Facultad de Ingeniería de Sistemas e Informática. Usando un proceso de desarrollo basado en pruebas y el marco de trabajo de SCRUM [26].

4.1. Metodología del curso

El curso implementa la metodología de aprendizaje basado en problemas y aprendizaje basado en proyectos. Se plantea situaciones problemáticas que deben ser resueltas mediante ejercicios de indagación, definición de la propuesta de proyecto, definición de objetivo, implementación, análisis de resultados esperados, seguimiento, control y la presentación oral de los resultados del Proyecto Integrador. Para tal fin, se plantean cinco (5) etapas que resguardan el desarrollo del curso y se exponen a continuación.

4.1.1. Propuesta del Proyecto Integrador

En esta etapa, el estudiante concentra sus esfuerzos en desarrollar un artefacto que incluye la descripción y delimitación de la situación problemática. Esto identifica y describe el problema que el proyecto busca resolver, proporcionando un contexto y una definición clara del problema planteado. Además, debe incluir la pregunta de investigación, que está orientada al enfoque de la investigación aplicada para la creación de un producto.

El estudiante también realiza un proceso de búsqueda y revisión de la literatura relacionada con el problema. Esta revisión debería incluir investigaciones previas, teorías relevantes y cualquier otro trabajo que pueda informar o influir en el proyecto.



El artefacto debe incluir el objetivo general y los objetivos específicos, que deben ser claros, medibles y alcanzables. Estos objetivos deben responder a las preguntas: ¿Qué? ¿Cómo? y ¿Para qué? Aunque la justificación deriva de la definición del problema, es importante incluir la relevancia del problema, los beneficios potenciales de resolverlo y la contribución del proyecto. Además, el artefacto debe describir la metodología para el desarrollo del producto. Esto puede incluir las técnicas de recopilación de datos, los métodos de análisis y cualquier otra estrategia que se utilizará en el desarrollo del producto.

Las actividades que se enumeran son las tareas que se llevarán a cabo para alcanzar los objetivos del proyecto. Es importante que cada actividad esté vinculada a un objetivo específico. De lo anterior se definen los entregables, o productos tangibles, que resultarán del proyecto. Por ejemplo, un informe de investigación, un prototipo de software, modelos, código, pruebas, etc. Los entregables deben contemplar el producto y el proyecto. Por último, se debe agregar un cronograma detallado y el presupuesto inicial. Este último debería describir los costos asociados con la realización del proyecto e incluir costos de personal, costos de recursos, los costos de infraestructura para el desarrollo, etc.

4.1.2. Análisis de requerimientos y planificación de la solución

El análisis de requerimientos y la planificación de la solución son etapas cruciales en el ciclo de vida del software para el proyecto. Esta fase incluye el análisis de requerimientos, que implica la identificación y documentación de las necesidades y expectativas del cliente. Los requerimientos pueden ser funcionales (lo que el sistema debe hacer) o no funcionales (cómo debe funcionar el sistema). Este análisis debería incluir entrevistas con los interesados, la especificación del problema basada en un análisis del modelo del dominio, discusiones con los interesados y la revisión de la documentación existente.

Una vez que se han identificado los requerimientos, se formalizan en un formato estándar para que todos los interesados puedan entenderlos. La especificación de requerimientos debe incluir al menos una especificación en diagramas de casos de uso, descripciones textuales y otros artefactos que ayuden a comunicar los requerimientos de manera efectiva. Además, en esta etapa se realiza el análisis de los datos con un examen detallado que puede incluir la identificación de las entidades de datos, sus atributos y las relaciones entre ellas. Derivado del análisis y especificación de requerimientos, se actualiza el cronograma y presupuesto para asegurar que el proyecto se complete a tiempo y dentro del alcance.

Por último, se diseña una estrategia de seguimiento y control. Durante la ejecución del proyecto, se realiza un seguimiento continuo del progreso del proyecto. Si se identifican cambios, problemas o desviaciones, se establece cómo se manejará el control de cambios para minimizar el impacto en el proyecto.

4.1.3. Diseño y desarrollo del Proyecto Integrador

La etapa de diseño y desarrollo del proyecto integrador es donde se toman las decisiones técnicas y se construye el producto de software.

Por parte del diseño, se toman decisiones basadas en los requerimientos y análisis previos. Se definen las estructuras de datos, la arquitectura del software, las interfaces de usuario y las interacciones entre los diferentes componentes del sistema. Se suele utilizar diagramas UML, y otras especificaciones de modelado para visualizar y documentar el diseño. Se realiza el diseño de infraestructura, estructura, comportamiento, datos y casos de prueba.



En cuanto a el desarrollo de Software, se codifica, configura y despliega el sistema basándose en el diseño, seguido de pruebas unitarias e integración entre otras, a medida que se van desarrollando los diferentes componentes del software. Si se encuentran errores durante las pruebas, se depuran y se corrigen. Es importante mencionar que este es un proceso iterativo e incremental, donde cada etapa puede revisarse y refinarse a medida que se obtiene más información y se comprende mejor el problema.

4.1.4. Análisis de resultados del Proyecto Integrador

En esta etapa, se evalúa el éxito del proyecto en función de las metas establecidas. Además, se revisan los objetivos del proyecto en relación con el producto final. Esto puede implicar la revisión de los entregables, la evaluación de la funcionalidad del producto, la medición del progreso de las actividades, así como la validación de las pruebas, la verificación, la resolución de problemas encontrados, la revisión del cumplimiento del tiempo y los recursos presupuestados, además, la estimación del impacto. Si los resultados no coinciden con los objetivos, se identifican los problemas y se analizan las posibles causas. Finalmente, se reflexiona sobre el proceso del proyecto y se identifican las lecciones aprendidas.

4.1.5. Estrategias de divulgación de resultados del Proyecto Integrador

Presentación oral, informes, bitácoras, ensayos, artículos, videos y recursos multimedia.

4.1.6. Cierre del Proyecto Integrador

Discusión, lecciones aprendidas y conclusiones.

4.2. Proceso de desarrollo de software para el Proyecto Integrador

Para el desarrollo del Proyecto Integrador se recomienda seguir las prácticas de desarrollo de software orientada por pruebas o Test-Driven Development (TDD) [27]. TDD tiene por objeto diseñar y escribir pruebas automatizadas antes de escribir el código. Las pruebas guían el diseño, así como la implementación del software desde la definición de requerimientos y sus criterios de aceptación [28]. Además, TDD promueve un enfoque incremental e iterativo en el desarrollo de software, que facilita la inclusión de cambios, adaptación y mejora en el proceso de desarrollo, así como compatibilidad con los principios y modelos ágiles [29].

4.2.1. Ciclo de desarrollo dirigido por pruebas

4.2.1.1. *Diseñar y escribir la prueba.* Se escriben pruebas unitarias automatizadas que definen las funcionalidades o mejoras de los casos de uso. Estas pruebas inicialmente fallan porque el código aún no se ha implementado [27]. Este proceso exige que se debe comprender los requisitos antes de escribir el código. Los requerimientos van avanzando a medida que se cumple con las pruebas [30].

4.2.1.2. *Ejecutar la prueba.* Las pruebas se ejecutan para validar que el código cumple con los requerimientos. Las pruebas que fallan lo hacen porque son funcionalidades aún no implementadas o porque se ha reescrito una prueba existente para cumplir con el requerimiento [31].

4.2.1.3. *Diseñar y desarrollar el código.* Se implementa la mínima cantidad de código necesario para que pase la prueba. Se debe mantener el código lo más simple posible. No se implementa funcionalidad más de la que la prueba requiere [32].

4.2.1.4. *Ejecutar las pruebas nuevamente.* Las pruebas se ejecutan nuevamente. Este proceso corrobora que no se han introducido errores. Si las pruebas no pasan, los casos de uso deberán recodificarse hasta que lo hagan. Se debe reescribir cambios mínimos que no introduzcan nuevos errores [33].

4.2.1.5. *Refactorizar el código.* Refactorizar es importante para ajustar el código, mejorar su estructura, legibilidad o rendimiento. Así como eliminar duplicados y establecer patrones. Refactorizar no implica crear código nuevo orientado a cubrir nuevas funcionalidades. Al finalizar la refactorización, se deben ejecutar todas pruebas para asegurarse que no se ha introducido nuevos errores [34].

4.2.1.6. El ciclo se repite para cada nueva funcionalidad o mejora en el código.

4.3. Marco de referencia para el desarrollo del Proyecto Integrador

El desarrollo del presente Proyecto Integrador se adapta al marco de trabajo de Scrum. Scrum, “es un Framework adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto” [26]. Este marco de trabajo cuenta con 19 procesos distribuidos en cinco (5) fases o etapas de ciclo de vida. Entre ellas la fase de Iniciación (6 procesos), la fase de Planificación y Estimación (6 procesos), la fase de Implementación (3 procesos), la fase de Revisión y Retrospectiva (2 procesos) y la fase de Lanzamiento (2 procesos) [26]. Scrum coordina un proceso iterativo entre las fases de Revisión y Retrospectiva con la fase de Planificación y Estimación. Para la aplicación en el Proyecto Integrador las etapas han sido adaptadas y se describen mediante tablas en el presente documento. En cada tabla hay tres columnas que refieren las entradas al proceso, las herramientas recomendadas y la salida de cada uno. Para más información acerca de aspectos generales de Scrum, consulte el SBOK [26].

4.3.1. Iniciación

4.3.1.1. Creación de la visión del proyecto y elección del dueño del producto (Product Owner)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Caso de negocio del proyecto. ✓ Misión y visión de la empresa. ✓ Documento de definición del Proyecto Integrador. ✓ Documento: Análisis y Modelo UML del Dominio. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reuniones de la visión del proyecto. ✓ Análisis de Fortalezas, Oportunidades, Debilidades y Amenazas. 	<ul style="list-style-type: none"> ✓ Elección del dueño del producto (Product Owner). ✓ Propuesta del Proyecto Integrador. ✓ Declaración de la visión del proyecto. ✓ Presupuesto (primera aproximación). ✓ Acta de constitución del proyecto integrador (Project Chapter).

Tabla 9. Creación de la visión del proyecto y elección del dueño del producto (Product Owner)

4.3.1.2. Elección del experto en desarrollo de proyectos en el marco de Scrum (Scrum Master)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Product Owner. ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del Proyecto Integrador. ✓ Project Chapter: Declaración de la visión del proyecto. 	<ul style="list-style-type: none"> ✓ Criterios de selección (experiencia, horas en proyectos, conoce el tipo de negocio, este certificado en el marco de trabajo, etc.). 	<ul style="list-style-type: none"> ✓ Elección del Scrum Master. ✓ Socio(s) identificados (personas u organizaciones externas). ✓ Actualización: Project Chapter.

<ul style="list-style-type: none"> ✓ Requerimientos de las personas o perfiles del equipo. ✓ Disponibilidad y compromiso de los prototipos. ✓ Matriz de recursos organizacionales. ✓ Matriz de destrezas requeridas. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Asesoramiento de expertos en RRHH. ✓ Costos de recursos relacionados. 	
---	--	--

Tabla 10. Elección del experto en desarrollo de proyectos en el marco de Scrum (Scrum Master)

4.3.1.3. Formación del equipo Scrum (Scrum Team)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Product Owner. ✓ Scrum Master. ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del proyecto integrador. ✓ Project Chapter: Declaración de la visión del proyecto. ✓ Reunión de la visión del proyecto (Product Owner y Scrum Master). ✓ Requerimientos de las personas o perfiles del equipo. ✓ Disponibilidad y compromiso de los prototipos. ✓ Matriz de recursos organizacionales. ✓ Matriz de destrezas requeridas. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Selección del equipo Scrum. ✓ Asesoramiento de expertos en RRHH. ✓ Costos de recursos relacionados. ✓ Capacitación y costos de capacitación. 	<ul style="list-style-type: none"> ✓ Elección del Scrum Team. ✓ Suplentes. ✓ Plan de colaboración. ✓ Plan para la formación del equipo (empatía, actividades de integración, capacitación, entre otros). ✓ Actualización: Project Chapter.

Tabla 11. Formación del equipo Scrum (Scrum Team)

4.3.1.4. Desarrollo de las Épicas (Epics)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del Proyecto Integrador. ✓ Project Chapter: Declaración de la visión del proyecto. ✓ Socio(s). ✓ Identificación de riesgos. ✓ Leyes y regulaciones. ✓ Información de proyectos previos. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reuniones con grupos de usuarios. ✓ Reuniones del grupo de enfoque. ✓ Entrevistas usuario y/o cliente. ✓ Cuestionarios. ✓ Técnicas de identificación de riesgos. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Épicas (Epics). ✓ Prototipo(s). ✓ Cambios aprobados. ✓ Riesgos identificados. ✓ Actualización: Project Chapter.

Tabla 12. Desarrollo de las Épicas (Epics)

4.3.1.5. Creación de la lista prioriza de pendientes del producto (Product Backlog)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). 	<ul style="list-style-type: none"> ✓ Método de priorización de historias de usuario (Por 	<ul style="list-style-type: none"> ✓ Lista priorizada de pendientes del producto (Product Backlog). ✓ Criterios de terminado (DONE).

<ul style="list-style-type: none"> ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del Proyecto Integrador. ✓ Project Chapter: Declaración de la visión del proyecto. ✓ Project Chapter: Epics. ✓ Project Chapter: Riesgos identificados. ✓ Prototipo(s). ✓ Socio(s). ✓ Documento de especificación de requerimientos (Funcionales y no funcionales). ✓ Solicitudes de cambios aprobadas. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ejemplo, priorización basada en la serie de Fibonacci). ✓ Talleres de historias de usuario. ✓ Planificación de valor de las historias de usuario. ✓ Técnicas de identificación de riesgos. ✓ Estimación de valor del proyecto. ✓ Métodos de estimación de historias de usuario. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Actualización: Project Chapter.
---	---	--

Tabla 13. Creación de la lista prioriza de pendientes del producto (Product Backlog)

4.3.1.6. Planificación del lanzamiento (Project Charter)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del Proyecto Integrador. ✓ Project Chapter. ✓ Product Backlog. ✓ Product Backlog: Criterios de terminado. ✓ Documento de especificación de requerimientos (Funcionales y no funcionales). ✓ Calendario. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Sesiones de planificación del lanzamiento. ✓ Métodos de priorización del lanzamiento. ✓ Cronograma del proyecto integrador. 	<ul style="list-style-type: none"> ✓ Cronograma de planificación del lanzamiento. ✓ Duración del Sprint. ✓ Metas para el lanzamiento. ✓ Actualización: Product Backlog. ✓ Actualización: Project Chapter.

Tabla 14. Planificación del lanzamiento (Project Charter)

4.3.2. Planificación y estimación

4.3.2.1. Creación de historias de usuario (User Stories)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Documento de definición del Proyecto Integrador. ✓ Propuesta del Proyecto Integrador. ✓ Project Chapter: Declaración de la visión del proyecto. ✓ Project Chapter: Epics. ✓ Product Backlog. ✓ Prototipo(s). ✓ Stakeholder(s). ✓ Documento de especificación de requerimientos (Funcionales y no funcionales). ✓ Leyes y regulaciones. 	<ul style="list-style-type: none"> ✓ Experiencia en la redacción de historias de usuario. ✓ Talleres de historias de usuario. ✓ Reuniones del grupo de usuarios. ✓ Reuniones del grupo de enfoque. ✓ Entrevistas cliente o usuario. ✓ Cuestionarios y entrevistas. ✓ Métodos de estimación de historias de usuario. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Historias de usuario (User Stories). ✓ Criterios de aceptación de las historias de usuario. ✓ Actualización: prototipos. ✓ Actualización: Product Backlog.

✓ Recomendaciones del cuerpo de asesoramiento de Scrum.		
---	--	--

Tabla 15. Creación de historias de usuario (User Stories)

4.3.2.2. Estimación de historias de usuario

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Product Backlog. ✓ Project Chapter: Epics. ✓ User Stories. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de planificación del Sprint. ✓ Reuniones de revisión del Backlog. ✓ Técnicas de estimación. 	<ul style="list-style-type: none"> ✓ Actualización: Historias de usuario (User Stories) estimadas. ✓ Actualización: Criterios de aceptación de las historias de usuario. ✓ Actualización: Product Backlog Priorizado. ✓ Sprint Goals.

Tabla 16. Estimación de historias de usuario

4.3.2.3. Comprometer historias de usuarios

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ User Stories. ✓ Sprint Goals. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de planificación del Sprint. ✓ Técnicas de estimación. 	<ul style="list-style-type: none"> ✓ Actualización: Historias de usuario (User Stories) estimadas, comprometidas y asignadas.

Tabla 17. Comprometer historias de usuarios

4.3.2.4. Identificación de tareas (To do list)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ User Stories. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de planificación del Sprint. ✓ Tareas de descomposición. ✓ Determinación de dependencias. ✓ Técnicas de estimación. 	<ul style="list-style-type: none"> ✓ Lista de tareas (To do list). ✓ Actualización: Historias de usuario (User Stories).

Tabla 18. Identificación de tareas (To do list)

4.3.2.5. Estimación de tareas

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ To do list. ✓ User Stories: Criterios de aceptación de las historias de usuario. ✓ Dependencias. ✓ Riesgos identificados. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de planificación del Sprint. ✓ Criterios de estimación. ✓ Técnicas de estimación. 	<ul style="list-style-type: none"> ✓ Lista del esfuerzo estimado por tareas (si es primera vez se recomienda estimación holgada). ✓ Actualización: To do list.

Tabla 19. Estimación de tareas

4.3.2.6. Creación de la lista de pendientes del Sprint (Sprint Backlog)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ To do list. ✓ Project Charter: Duración del Sprint. ✓ Dependencias. ✓ Calendario. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de planificación del Sprint. ✓ Herramientas de seguimiento del Sprint. ✓ Parámetros de seguimiento del Sprint. 	<ul style="list-style-type: none"> ✓ Lista de pendientes del Sprint (Sprint Backlog). ✓ Grafica de trabajo pendiente del Sprint (Sprint Burndown Chart).

Tabla 20. Creación de la lista de pendientes del Sprint (Sprint Backlog)

4.3.3. Implementación

4.3.3.1. Creación de entregables

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Sprint Backlog. ✓ Tablero Scrum. ✓ Registro de impedimentos. ✓ Project Charter: Cronograma de planificación del lanzamiento. ✓ Dependencias. 	<ul style="list-style-type: none"> ✓ Experiencia del equipo. ✓ Herramientas de seguimiento del Sprint ✓ Herramientas de desarrollo. ✓ Herramientas de Software. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Entregables del Sprint. ✓ Actualización: Tablero Scrum. ✓ Actualización: Registro de impedimentos. ✓ Solicitud de cambios no aprobadas. ✓ Actualización: Registro de riesgos identificados y mitigados. ✓ Actualización: registro de dependencias.

Tabla 21. Creación de entregables

4.3.3.2. Realización de la reunión seguimiento diaria (Daily Scrum)

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master (optional) y Development Team). ✓ Sprint Backlog. ✓ Tablero Scrum. ✓ Sprint Burndown Chart. ✓ Registro de impedimentos. ✓ Experiencia del día anterior de trabajo. ✓ Dependencias. 	<ul style="list-style-type: none"> ✓ Reunión diaria de pie del equipo (máximo 15 min). ✓ Tres preguntas: ¿Qué hice ayer? ¿Qué es lo que voy a hacer hoy? ¿He tenido algún tipo de problema? ✓ Salón de guerra. ✓ Videoconferencia. ✓ Herramientas de seguimiento del Sprint. 	<ul style="list-style-type: none"> ✓ Actualización: Sprint Backlog. ✓ Actualización: Tablero Scrum. ✓ Actualización: Registro de impedimentos. ✓ Equipo Scrum motivado. ✓ Solicitud de cambios no aprobadas. ✓ Actualización: Registro de riesgos identificados y mitigados. ✓ Actualización: registro de dependencias.

Tabla 22. Realización de la reunión seguimiento diaria (Daily Scrum)

4.3.3.3. Refinamiento del Product Backlog

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Product Backlog. ✓ Project Charter: Cronograma de planificación del lanzamiento. ✓ Dependencias. 	<ul style="list-style-type: none"> ✓ Reunión de revisión del Product Backlog. ✓ Técnicas de comunicación. ✓ Técnicas de mantenimiento del Product Backlog. 	<ul style="list-style-type: none"> ✓ Actualización: Product Backlog. ✓ Actualización: Cronograma de planificación del lanzamiento.

<ul style="list-style-type: none"> ✓ Entregables rechazados. ✓ Solicitud de cambios aprobadas. ✓ Solicitud de cambios rechazados. ✓ Registro de riesgos identificados y mitigados. ✓ Dependencias. ✓ Calendario. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 		
---	--	--

Tabla 23. Refinamiento del Product Backlog

4.3.4. Revisión y Retrospectiva

4.3.4.1. Demostración y validación del Sprint

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Entregables del Sprint ✓ Product Backlog. ✓ Criterios de Terminado ✓ Criterios de Aceptación de las historias de usuario. ✓ Socio(s). ✓ Cronograma de planificación del lanzamiento ✓ Riesgos identificados. ✓ Dependencias. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reuniones de revisión del Sprint. ✓ Análisis del valor ganado. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Entregables aceptados. ✓ Entregables rechazados. ✓ Riesgos actualizados. ✓ Resultados del análisis del valor ganado. ✓ Cronograma de planificación del lanzamiento actualizado. ✓ Dependencias actualizadas.

Tabla 24. Demostración y validación del Sprint

4.3.4.2. Retrospectiva del Sprint

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Salidas de la demostración y validación del Sprint. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de retrospectiva del Sprint. ✓ Técnica: Explorador, Comprador, Vacacionista, y Prisionero. ✓ Parámetros y técnicas de medición. ✓ Evaluación y coevaluación del equipo Scrum. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Mejoras accionables aceptadas. ✓ Elementos de acción asignados y fechas de entrega. ✓ Elementos no funcionales propuestos para la lista priorizada de pendientes del producto. ✓ Registro de la retrospectiva del sprint. ✓ Lecciones aprendidas del equipo Scrum. ✓ Recomendaciones actualizadas del cuerpo de asesoramiento de Scrum.

Tabla 25. Retrospectiva del Sprint

4.3.5. Lanzamiento

4.3.5.1. Liberación de Entregables

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Socio(s). ✓ Entregables aceptados. ✓ Criterios de Aceptación de las historias de usuario. ✓ Cronograma de planificación del lanzamiento ✓ Pla de pilotaje. ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Métodos de desplazamiento organizacional. ✓ Plan de comunicación. 	<ul style="list-style-type: none"> ✓ Acuerdo de entregables funcionales. ✓ Entregables Funcionales. ✓ Lanzamiento del producto.

Tabla 26. Liberación de Entregables

4.3.5.2. Retrospectiva del Proyecto

Entradas	Herramientas	Salidas
<ul style="list-style-type: none"> ✓ Scrum Team (Product Owner, Scrum Master y Development Team). ✓ Socio(s). ✓ Recomendaciones del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Reunión de retrospectiva del Proyecto. ✓ Herramientas para retrospectiva del proyecto. ✓ Experiencia del cuerpo de asesoramiento de Scrum. 	<ul style="list-style-type: none"> ✓ Mejoras accionables aceptadas. ✓ Elementos de acción asignados y fechas límite. ✓ Elementos no funcionales propuestos para la lista priorizada de pendientes del proyecto y producto. ✓ Recomendaciones actualizadas del cuerpo de asesoramiento de Scrum.

Tabla 27. Retrospectiva del Proyecto

5. Artefactos relacionados con el Proyecto Integrador

Los artefactos mínimos y contenido mínimo relacionados con la gestión del proyecto se enumeran a continuación.

5.1. Documento de definición del Proyecto Integrador

Documento actual.

5.2. Presentación de Propuesta del Proyecto Integrador

- Especificación de la situación problemática.
- Objetivos.
- Metodología.

5.3. Propuesta del Proyecto Integrador

- Especificación de la situación problemática.
- Marco tecnológico y conceptual.
- Objetivos.
- Justificación.



- Metodología.
- Cronograma de actividades.
- Referencias.

5.3.1. Acta de constitución del proyecto Integrador (Project Charter)

- Información general del Proyecto Integrador.
- Nombre del Proyecto Integrador.
- Promotor del proyecto Integrador.
- Patrocinador del Proyecto Integrador.
- Definición del Product Owner.
- Definición del Scrum Master.
- Definición del Scrum Team.
- Interesados (Stakeholders).
- Socio(s).
- Fecha de inicio esperada.
- Fecha de finalización.
- Detalles del Proyecto Integrador.
- Declaración de la visión del Proyecto Integrador.
- Planteamiento del problema.
- Misión.
- Visión.
- Alcance.
- Objetivo general.
- Objetivos específicos.
- Beneficios esperados.
- Especificación de requerimientos.
- Identificación de riesgos.
- Criterios de aceptación del Proyecto Integrador.
- Duración del Sprint.
- Criterios de terminado (Done).
- Cronograma de lanzamientos.
- Presupuesto inicial.
- Documentación de soporte.

5.4. Lista de pendientes del producto (Product Backlog)

- Épicas (Epics).
- Historias de usuario (User stories).
- Criterios de aceptación de las historias de usuario.

5.5. Listas de pendientes del Sprint (Sprint Backlogs)

- Tareas.

5.6. Acta de seguimiento del Proyecto Integrador

El acta de seguimiento del Proyecto Integrador es un documento formal donde se describe los eventos, acciones, procesos y desempeño del equipo a lo largo de una semana. El acta es muy importante porque permite la personalización de la nota y debe entregarse todas las semanas. El acta debe ser asignada a un miembro del equipo a la vez, pero el contenido debe generarse por todo el equipo. Debe publicarse en el espacio de trabajo del equipo (Por ejemplo, Microsoft Teams) Y contener como mínimo la siguiente información:

- Fecha año, mes, día, y hora.
- Descripción.



Nota del proyecto: Si no se entrega el acta, toda nota asociada a esa semana que se encuentre relacionada con el proyecto o producto se multiplicará por cero.

- Evaluación del desempeño individual: Descripción de las tareas asignadas a cada miembro del equipo, que incluye, el estado actual de cada tarea, como completada, en progreso, o deuda técnica, entre otros. Si se encuentra en progreso, que tanto se ha completado en porcentaje. Si se encuentra retrasada informar acerca del motivo. Ejemplo:

ID	Nombre	Completado	Estado	Observaciones
123123	Marty McFly Klein	80%	En progreso	Marty no ha completado las pruebas del requerimiento n, además, no participó en la entrega del Story mapping como se había comprometido.
987987	Sasha Zo El	100%	completado	Todas las tareas se han completado antes de tiempo y ha ayudado a contribuir en disminuir la deuda técnica de los compañeros Bender y Lila.

Tabla 28. Acta de seguimiento.

- Evaluación del desempeño del equipo: Burndown Chart, identificación de problemas que afectan al equipo, compromisos y propuestas de soluciones o acciones para mejorar.
- Firma del editor.

5.7. Acta de entrega y cierre del Proyecto Integrador

- Título del proyecto.
- Fecha de inicio del proyecto.
- Fecha de fin del proyecto.
- Presupuesto y costo final.
- Objetivos: criterios de aceptación, verificación y porcentaje de cumplimiento.
- Productos y/o servicios generados.
- Resultados del proyecto.

- Impacto futuro.
- Firmas y sellos.



Nota del proyecto: Este resumen de algunos artefactos relacionados con el proyecto se presenta con el fin de que los grupos se orienten y operen una lista de verificación de cuáles son los elementos mínimos en la gestión de proyectos que deben desarrollar para las entregas.

6. Objetivos de Aprendizaje

- Desarrollar habilidades en el análisis, diseño, desarrollo e implementación de software, contemplando buenos niveles de seguridad, confiabilidad e integridad de los datos.
- Desarrollar habilidades de expresión oral.
- Desarrollar habilidades para la documentación técnica y para la elaboración de informes.
- Desarrollar habilidades para el liderazgo, planeación y ejecución de proyectos.
- Analizar el funcionamiento de los protocolos y servicios de red.
- Diseñar e implementar la infraestructura tecnológica requerida para el desarrollo e implementación de la solución.
- Realizar operaciones complejas con los datos, usando el lenguaje procedimental de los gestores.
- Aplicar las normas de seguridad y redundancia en las diferentes bases de datos.
- Aplicar herramientas y procesos de ingeniería de software en la construcción o adaptación de software seguro.

7. Actividades de aprendizaje

- Habilidades lectoras, de exploración sobre los temas requeridos, y habilidades para escribir documentos y citar fuentes.
- Conceptos de definición de arquitecturas tecnológicas.
- Capacidad de diseñar e implementar aplicaciones basadas en servicios.
- Indaga en diversas fuentes sobre las temáticas involucradas para el desarrollo del proyecto.
- Capacidad de diseñar e implementar un modelo de base de datos.
- Definición de protocolos de pruebas.
- Elaboración de documentos.
- Trabajo en equipo.
- Aprendizaje autónomo.
- Capacidad para implementar software.
- Implementa infraestructuras de red y sincroniza servicios.
- Implementa procedimientos almacenados que aumentan el rendimiento de la base de datos.
- Crea estructuras que mejoran el tiempo de respuesta en consultas.
- Implementa niveles de seguridad del sistema y la infraestructura.
- Elaboración de documentos.



8. Fuentes de información

- [1] Hernández Sampieri, R., Fernández Collado, C., Baptista Lucio, M.P. (2014) Metodología de la investigación -- 6a ed. -- México: McGraw-Hill.
- [2] Niño Rojas, V.M. (2011) Metodología de la investigación: diseño y ejecución -- Bogotá: Ediciones de la U.
- [3] Hernández Sampieri, R. (2010) Metodología de la investigación -- 5a ed. -- México: McGraw-Hill, 614 p.
- [4] Kurose, J F., Ross, K W. (2010) Redes de Computadoras: un enfoque descendente. 5ª. Edición. Pearson Education.
- [5] Forouzan, B A. (2006) Transmisión de datos y redes de comunicaciones. Cuarta Edición. McGrawHill.
- [6] Stallings, W. (2004) Comunicaciones y Redes de Computadores. 7ª. Edición. Prentice Hall.
- [7] Tanenbaum S., A. (2013) Redes de Computadoras. Quinta Edición. Prentice Hall.
- [8] Mannino, M. V. (2007) Administración de Bases de Datos: Diseño y Desarrollo de Aplicaciones. 3a. edición. México: McGraw-Hill.
- [9] Rob, P., coronel, C. (2004) Sistemas de Bases de Datos: Diseño, Implementación y Administración. 5a. edición. México: International Thomson Editores.
- [10] Project Management Institute. (2017). A guide to the project management body of knowledge (6th ed.). Newtown Square, Pennsylvania, USA.
- [11] Graziani, R. (2017) IPv6 fundamentals: A straightforward approach to understanding IPv6. Indianapolis: Cisco Press.
- [12] Kurose, J F., Ross, K W. (2013) Computer Networking: A top-down approach. 6th edition. Addison Wesley.
- [13] Stallings, W. (2013) Data and Computer Communications. Tenth edition. Pearson.
- [14] Wallace, K., Hucaby, D. (2014) CCNP Routing and Switching v2.0. Cisco Press.
- [15] Stevens, W. R. (2011) TCP/IP Illustrated: The Protocols. Addison Wesley.
- [16] Alam, M. (2014) Oracle NoSQL database: real-time big data management for the enterprise -- New York: McGraw-Hill.
- [17] Cabral, S., Murphy, K. (2009) MySQL Administrator's Bible. 1a Edición. Wiley.
- [18] Celki, J. (2015) Joe Celko's SQL for smarties advanced SQL programming -- 5th ed. -- Waltham: El Sevier.
- [19] Gertz, M., Jajodia, S. (2008) Handbook of Database Security: Applications and Trends -- New York: Springer.



- [20] M. A. Gallo and W. M. Hancock, Comunicación entre computadores y tecnologías de redes. Thomson, 2002.
- [21] D. Mauro and K. Schmidt, Essential SNMP: Help for System and Network Administrators." O'Reilly Media, Inc.", 2005.
- [22] A. Clemm et al., Network management fundamentals. Cisco Press Indianapolis, IN, USA: 2007, vol. 800.
- [23] D. Hucaby, CCNP Routing and Switching SWITCH 300-115 Official Cert Guide: Exam 38 Cert Guide. Cisco Press, 2014.
- [24] W. Lewis, Cisco Networking Academy Program: CCNP 3: Multilayer Switching Lab Companion. Cisco Press, 2005.
- [25] K. R. Fall and W. R. Stevens, TCP/IP illustrated, volume 1: The protocols. addison-Wesley, 2011.
- [26] SCRUMstudy, A guide to the Scrum Body of knowledge (SBOK Guide). <https://www.scrumstudy.com/account/registration-success>. 2016.
- [27] D. Astels, Test-driven Development: A Practical Guide. Prentice Hall PTR, 2003.
- [28] R. C. Martin, Agile software development: principles, patterns, and practices. Prentice Hall PTR, 2003.
- [29] K. Beck, Test Driven Development. By Example (Addison-Wesley Signature). Addison-Wesley Longman, Amsterdam, 2002.
- [30] L. Koskela, Test driven practical tdd and acceptance tdd for java developers. Simon and Schuster, 2007.
- [31] H. Percival, Test-driven development with Python: obey the testing goat: using Django, Selenium, and JavaScript. O'Reilly Media, Inc.", 2014.
- [32] J. H. Hayes, A. Dekhtyar, and D. S. Janzen, "Towards traceable test-driven development," in 2009 ICSE Workshop on Traceability in Emerging Forms of Software Engineering, pp. 26–30, IEEE, 2009.
- [33] B. Turhan, L. Layman, M. Diep, H. Erdogmus, and F. Shull, "How effective is test-driven development," Making Software: What Really Works, and Why We Believe It, pp. 207–217, 2010.
- [34] M. Fowler, Refactoring. Addison-Wesley Professional, 2018.