

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

ESCOM

Trabajo Terminal

“Sistema Web para la Organización Personal”

2024 - A136

Presenta:
Julio Cesar Hernández Reyes

Director:
Ulises Vélez Saldaña

Estados Unidos Mexicanos
Ciudad de México
Junio 2024





**INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO
SUBDIRECCION ACADEMICA**



No de TT: 2024 - A136

10 junio 2024

Documento Técnico

“Sistema Web para la Organización Personal”

Presenta:

Julio Cesar Hernández Reyes

Director:

Ulises Vélez Saldaña

RESUMEN

El presente documento es el reporte técnico final del Trabajo Terminal con el número 2024-A136 que tiene por nombre “Sistema Web para la Organización Personal”, correspondiente al análisis, diseño y desarrollo del sistema “Tempus Victor” a lo largo del semestre 2024-1 y 2024-2, con el fin de acreditar la unidad de aprendizaje así como servir para opción de titulación de la carrera en Ingeniería en Sistemas Computacionales y poder obtener el título de Ingeniero en Sistemas Computacionales dentro de la Escuela Superior de Cómputo del Instituto Politécnico Nacional. El sistema web desarrollado tiene como objetivo proporcionar apoyo a estudiantes en su organización personal haciendo uso de herramientas personalizables enfocadas en tiempo y tareas; el sistema incluye varios módulos personalizables que les proporcionan ayuda a los estudiantes en su día a día tales como: módulo de notas, módulo de tareas, módulo de calendario, módulo de Matriz Eisenhower, módulo de Temporizador Pomodoro, módulo de métricas, módulo de configuración y módulos de ayuda.

¹ jhernandezr1923@alumno.ipn.mx

ADVERTENCIA

“Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan.”

La aplicación no convenida exime a la escuela su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.

Información adicional sobre este reporte técnico podrá obtenerse en:

La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000, extensión 52000.

ÍNDICE

1. INTRODUCCIÓN	13
2. PLANTEAMIENTO DEL PROYECTO	16
2.1. PLANTEAMIENTO DEL PROBLEMA.....	16
2.1.1 <i>Problemáticas específicas</i>	17
2.1.2 <i>Problemática General</i>	20
2.2. PROPUESTA DE SOLUCIÓN.....	21
2.2.1 <i>Tempus Victor nombre del sistema Web</i>	21
2.3. OBJETIVO GENERAL	22
2.4. OBJETIVOS ESPECÍFICOS	22
2.5. JUSTIFICACIÓN.....	24
2.5.1 <i>Combatir la falta de planificación en estudiantes</i>	24
2.5.2 <i>Superar la mala gestión del tiempo en estudiantes</i>	24
2.5.3 <i>Minimizar la pérdida de información académica</i>	25
2.5.4 <i>Facilitar la priorización de tareas académicas</i>	25
2.5.5 <i>Beneficios del sistema web para estudiantes</i>	26
2.5.6 <i>Relevancia de la tarea dentro del Trabajo Terminal</i>	27
2.5.7 <i>Relevancia de Pomodoro dentro del Trabajo Terminal</i>	28
2.5.8 <i>Relevancia de matriz Eisenhower dentro del TT</i>	28
2.6. ESTADO DEL ARTE	30
2.6.1 <i>Google Calendar:</i>	31
2.6.2 <i>Todoist:</i>	32
2.6.3 <i>Evernote:</i>	33
2.6.4 <i>TickTick:</i>	33
2.6.5 <i>Notion:</i>	34
2.6.6 <i>Tempus Victor</i>	35
3. MARCO TEÓRICO	37
3.1. TAREAS.....	37
3.1.1 <i>Definición</i>	37
3.1.2 <i>Características</i>	37
3.1.3 <i>Diferenciación de conceptos</i>	38
3.1.4 <i>Tipos</i>	38
3.1.5 <i>Ciclo de Vida de una Tarea</i>	40
3.2. ORGANIZACIÓN PERSONAL	40
3.2.1 <i>Definición</i>	40
3.2.2 <i>Características</i>	41
3.3. HERRAMIENTAS PARA LA ORGANIZACIÓN PERSONAL	41
3.4. TÉCNICAS DE ORGANIZACIÓN PERSONAL	42
3.4.1 <i>Técnica Pomodoro:</i>	42
3.4.2 <i>Matriz de Eisenhower:</i>	42

3.5.	METODOLOGÍA	44
3.5.1	<i>Introducción al modelo en espiral</i>	44
3.5.2	<i>Fases del modelo en espiral</i>	44
3.5.3	<i>Ventajas del modelo en espiral</i>	46
3.5.4	<i>Aplicación al desarrollo del sistema</i>	47
3.6.	CONCEPTOS TÉCNICOS	50
3.6.1	<i>Modelo de tres capas</i>	50
3.6.2	<i>API (Interfaz de Programación de Aplicaciones)</i>	51
3.6.3	<i>Servicio RESTful</i>	52
3.6.4	<i>Frontend</i>	53
3.6.5	<i>Backend</i>	53
3.6.6	<i>Base de datos</i>	54
3.6.7	<i>MySQL</i>	54
3.6.8	<i>ORM</i>	54
3.6.9	<i>Prisma</i>	55
3.6.10	<i>Visual Studio Code</i>	55
3.6.11	<i>Angular</i>	56
3.6.12	<i>Express</i>	56
3.6.13	<i>Node.js</i>	57
3.6.14	<i>TypeScript</i>	57
3.6.15	<i>JWT</i>	57
3.6.16	<i>Node Package Manager (npm)</i>	58
4.	TRABAJO REALIZADO	59
4.1.	CRONOGRAMA DE ACTIVIDADES	61
4.2.	PRIMERA ITERACIÓN	62
4.2.1	<i>Establecimiento de objetivos para la iteración actual</i>	63
4.2.2	<i>Planteamiento de la problemática y su análisis</i>	63
4.2.3	<i>Planteamiento de la propuesta de solución</i>	63
4.2.4	<i>Investigación del Estado del Arte</i>	64
4.2.5	<i>Identificación de actores del sistema</i>	65
4.2.6	<i>Investigación y desarrollo del Marco Teórico</i>	66
4.2.7	<i>Identificación de requerimientos funcionales y no funcionales</i>	66
4.2.8	<i>Análisis y diseño del diagrama relacional</i>	67
4.2.9	<i>Identificación y análisis de riesgo</i>	68
4.2.10	<i>Corrección y actualización de la documentación</i>	68
4.2.11	<i>Resumen Iteración</i>	68
4.3.	SEGUNDA ITERACIÓN	69
4.3.1	<i>Establecimiento de objetivos para la iteración actual</i>	69
4.3.2	<i>Ánalysis de factibilidad económica</i>	69
4.3.3	<i>Ánalysis de factibilidad técnica</i>	74
4.3.4	<i>Tecnologías seleccionadas</i>	76
4.3.5	<i>Arquitectura del sistema</i>	77

4.3.6	<i>Diseño de interfaces gráficas del sistema</i>	79
4.3.7	<i>Planteamiento de casos de uso</i>	84
4.3.8	<i>Identificación y descripción de reglas del negocio</i>	86
4.3.9	<i>Desarrollo y corrección de interfaces gráficas del sistema</i>	86
4.3.10	<i>Pruebas y corrección de errores del sistema</i>	97
4.3.11	<i>Corrección y actualización de la documentación</i>	97
4.3.12	<i>Planificación de la siguiente iteración</i>	98
4.3.13	<i>Resumen Iteración</i>	98
4.4.	TERCERA ITERACIÓN	98
4.4.1	<i>Establecimiento de objetivos para la iteración actual</i>	98
4.4.2	<i>Identificación y evaluación de riesgos</i>	99
4.4.3	<i>Desarrollo y corrección de interfaces gráficas del sistema</i>	99
4.4.4	<i>Desarrollo y corrección de la base de datos del sistema</i>	101
4.4.5	<i>Implementación y corrección de casos de uso</i>	102
4.4.6	<i>Implementación y corrección de reglas del negocio</i>	102
4.4.7	<i>Pruebas y corrección de errores del sistema</i>	103
4.4.8	<i>Redacción y corrección de la documentación</i>	103
4.4.9	<i>Planificación de la siguiente iteración</i>	103
4.4.10	<i>Resumen Iteración</i>	104
4.5.	CUARTA ITERACIÓN	104
4.5.1	<i>Establecimiento de objetivos para la iteración actual</i>	104
4.5.2	<i>Identificación y evaluación de riesgos</i>	105
4.5.3	<i>Desarrollo y corrección de interfaces gráficas del sistema</i>	105
4.5.4	<i>Desarrollo y corrección de la base de datos del sistema</i>	107
4.5.5	<i>Implementación y corrección de casos de uso</i>	108
4.5.6	<i>Implementación y corrección de reglas del negocio</i>	109
4.5.7	<i>Pruebas y corrección de errores del sistema</i>	110
4.5.8	<i>Redacción y corrección de la documentación</i>	110
4.5.9	<i>Planificación de la siguiente iteración</i>	110
4.5.10	<i>Resumen Iteración</i>	110
4.6.	QUINTA ITERACIÓN	111
4.6.1	<i>Establecimiento de objetivos para la iteración actual</i>	111
4.6.2	<i>Identificación y evaluación de riesgos</i>	111
4.6.3	<i>Desarrollo y corrección de interfaces graficas del sistema</i>	112
4.6.4	<i>Desarrollo y corrección de la base de datos del sistema</i>	113
4.6.5	<i>Implementación y corrección de casos de uso</i>	114
4.6.6	<i>Implementación y corrección de reglas del negocio</i>	114
4.6.7	<i>Pruebas y corrección de errores del sistema</i>	115
4.6.8	<i>Redacción y corrección de la documentación</i>	115
4.6.9	<i>Resumen Iteración</i>	115
5.	RESULTADOS	116
6.	CONCLUSIONES	125

6.1.1	<i>Conclusiones generales</i>	125
6.1.2	<i>Conclusiones personales</i>	126
7.	TRABAJO A FUTURO	127
8.	REFERENCIAS	128
	ANEXOS	131
A.	ANÁLISIS DE RIESGOS	131
A.1	<i>Tabla de análisis de riesgos</i>	131
A.2	<i>Fichas de Riesgo</i>	132
B.	REQUERIMIENTOS FUNCIONALES	137
C.	REQUERIMIENTOS NO FUNCIONALES	141
D.	MODELO DEL NEGOCIO	141
D.1	<i>Términos del negocio</i>	142
D.2	<i>Hechos del Negocio</i>	142
D.3	<i>Reglas de negocio</i>	143
E.	MODELO DE LA BASE DE DATOS	148
F.	MODELO DE CASOS DE USO.....	156
F.1	<i>Diagrama de Casos de Uso</i>	156
F.2	<i>Gestion de Cuenta</i>	157
F.3	<i>Gestion de Notas</i>	158
F.4	<i>Gestion de Tareas</i>	159
F.5	<i>Gestion de Sesiones Pomodoro</i>	160
G.	CASOS DE USO	161
G.1	<i>CU1: Registrar Usuario</i>	161
G.2	<i>CU2: Iniciar Sesión</i>	162
G.3	<i>CU3: Cerrar Sesión</i>	163
G.4	<i>CU4: Ver Cuenta</i>	164
G.5	<i>CU5: Editar Cuenta</i>	165
G.6	<i>CU6: Cambiar Contraseña</i>	166
G.7	<i>CU7: Eliminar Cuenta</i>	167
G.8	<i>CU8: Personalizar Cuenta</i>	168
G.9	<i>CU9: Ver lista de notas</i>	170
G.10	<i>CU10: Agregar Nota</i>	171
G.11	<i>CU11: Ver Nota</i>	172
G.12	<i>CU12: Editar Nota</i>	173
G.13	<i>CU13: Eliminar Nota</i>	174
G.14	<i>CU14: Archivar Nota</i>	175
G.15	<i>CU15: Ayuda Notas</i>	175
G.16	<i>CU16: Vista Matriz Eisenhower</i>	176
G.17	<i>CU17: Vista Calendario</i>	177
G.18	<i>CU18: Agregar Tarea</i>	179
G.19	<i>CU19: Agregar Categorías</i>	181
G.20	<i>CU20: Ver Tarea</i>	181

G.21	<i>CU21: Editar Tarea</i>	182
G.22	<i>CU22: Eliminar Tarea</i>	183
G.23	<i>CU23: Ver Historial Tarea</i>	184
G.24	<i>CU24: Ver Lista Tareas Completadas</i>	185
G.25	<i>CU25: Ayuda Tareas</i>	186
G.26	<i>CU26: Métricas Tareas</i>	186
G.27	<i>CU27: Estimar Pomodoros</i>	187
G.28	<i>CU28: Temporizador Pomodoro</i>	189
G.29	<i>CU29: Sesión Pomodoro sin Iniciar</i>	191
G.30	<i>CU30: Sesión Pomodoro Abandonada</i>	193
G.31	<i>CU31: Sesión Pomodoro Pausada</i>	193
G.32	<i>CU32: Sesión Pomodoro Terminada</i>	194
G.33	<i>CU33: Historial de la Sesión Pomodoro</i>	195
G.34	<i>CU34: Métricas Sesiones Pomodoro</i>	195
G.35	<i>CU35: Ayuda Pomodoro</i>	196
H.	INTERFACES REALES DEL SISTEMA	198
H.1	<i>IU1 Pantalla De Bienvenida al Sistema</i>	198
H.2	<i>IU2 Registro de Usuario</i>	198
H.3	<i>IU3 Inicio de Sesión</i>	199
H.4	<i>IU4 Tareas Eisenhower</i>	199
H.5	<i>IU5 Eisenhower Seleccionar Tarea</i>	200
H.6	<i>IU6 Ayuda Vista Eisenhower</i>	200
H.7	<i>IU7 Modal Completar Tarea Vista Eisenhower</i>	201
H.8	<i>IU8 Tareas Calendario</i>	201
H.9	<i>IU9 Seleccionar Dia Calendario</i>	202
H.10	<i>IU10 Seleccionar Tarea Calendario</i>	202
H.11	<i>IU11 Ayuda Vista Calendario</i>	203
H.12	<i>IU12 Modal Completar Tarea Vista Calendario</i>	203
H.13	<i>IU13 Tareas Completadas</i>	204
H.14	<i>IU14 Agregar Tarea</i>	204
H.15	<i>IU15 Modal Agregar Categoría</i>	205
H.16	<i>IU16 Ver Tarea</i>	205
H.17	<i>IU17 Editar Tarea</i>	206
H.18	<i>IU18 Eliminar Tarea</i>	206
H.19	<i>IU19 Historial Tarea</i>	207
H.20	<i>IU20 Estimar Sesión Pomodoro</i>	207
H.21	<i>IU21 Temporizador Pomodoro</i>	208
H.22	<i>IU22 Temporizador Pomodoro Corriendo</i>	208
H.23	<i>IU23 Ayuda Pomodoro</i>	209
H.24	<i>IU24 Nuevo Pomodoro Temporizador</i>	209
H.25	<i>IU25 Modal Empezar Sesión Pomodoro</i>	210
H.26	<i>IU26 Modal Reiniciar Sesión Pomodoro</i>	210
H.27	<i>IU27 Modal Pausar Sesión Pomodoro</i>	211

<i>H.28 IU28 Modal Terminar Sesión Pomodoro</i>	211
<i>H.29 IU29 Modal Completar Tarea.....</i>	212
<i>H.30 IU30 Modal Alarma Tiempo Trabajo Completado</i>	212
<i>H.31 IU31 Modal Alarma Tiempo Descanso Completado</i>	213
<i>H.32 IU32 Notas</i>	213
<i>H.33 IU33 Seleccionar Nota.....</i>	214
<i>H.34 IU34 Ayuda Notas</i>	214
<i>H.35 IU35 Notas Archivadas.....</i>	215
<i>H.36 IU36 Agregar Nota</i>	215
<i>H.37 IU37 Ver Nota</i>	216
<i>H.38 IU38 Editar Nota</i>	216
<i>H.39 IU39 Eliminar Nota</i>	217
<i>H.40 IU40 Modal Archivar Nota</i>	217
<i>H.41 IU41 Cuenta.....</i>	218
<i>H.42 IU42 Cerrar Sesión.....</i>	218
<i>H.43 IU43 Editar Cuenta.....</i>	219
<i>H.44 IU44 Cambiar Contraseña</i>	219
<i>H.45 IU45 Eliminar Cuenta.....</i>	220
<i>H.46 IU46 Confirmar Eliminar Cuenta.....</i>	220
<i>H.47 IU47 Configuración</i>	221
<i>H.48 IU48 Ayuda Configuración</i>	221
<i>H.49 IU49 Resetear Valores Configuración</i>	222
<i>H.50 IU50 Guardar Valores Configuración</i>	222
<i>H.51 IU51 Métricas</i>	223
<i>H.52 IU52 Ayuda Métricas</i>	223

ÍNDICE DE TABLAS

TABLA 1:SISTEMAS WEB DE ORGANIZACIÓN PERSONAL EXISTENTES	31
TABLA 2: ANÁLISIS DE PUNTOS POR FUNCIÓN	70
TABLA 3: ANÁLISIS DE PUNTOS DE FUNCIÓN SIN AJUSTAR.....	70
TABLA 4: TABLA DE FACTOR DE AJUSTE.....	71
TABLA 5:ESTIMACIÓN DE ESFUERZO.....	72
TABLA 6:ESPECIFICACIÓN DEL HARDWARE DISPONIBLE.....	76
TABLA 7: PANTALLAS DISEÑADAS IMPORTANTES	79
TABLA 8: TABLA ESTÁNDAR DE PROBABILIDAD DE OCURRENCIA.....	131
TABLA 9: TABLA DE RIESGOS.....	132
TABLA 10: FICHA DE RIESGO RT-1	133
TABLA 11: FICHA DE RIESGO RT-2.....	133
TABLA 12: FICHA DE RIESGO RT-3	134
TABLA 13: FICHA DE RIESGO RT-4	134
TABLA 14: FICHA DE RIESGO RT-5	135
TABLA 15: FICHA DE RIESGO RT-6	135
TABLA 16: FICHA DE RIESGO RT-7	136
TABLA 17: FICHA DE RIESGO RT-8	136
TABLA 18: FICHA DE RIESGO RT-9	137
TABLA 19: REQUERIMIENTOS FUNCIONALES DEL SISTEMA	137
TABLA 20: REQUERIMIENTOS NO FUNCIONALES	141
TABLA 21: ENTIDAD USUARIO.....	150
TABLA 22: ENTIDAD CONFIGURACIÓN.....	150
TABLA 23: ENTIDAD NOTA	151
TABLA 24: ENTIDAD TAREA.....	151
TABLA 25: ENTIDAD EVENTOS	152
TABLA 26:ENTIDAD CATEGORÍA.....	152
TABLA 27: ENTIDAD SESIONES POMODORO.....	152
TABLA 28: ENTIDAD EVENTOS POMODORO	153
TABLA 29: CASOS DE USO GESTION DE CUENTA	157
TABLA 30: CASOS DE USO GESTION DE NOTAS.....	158
TABLA 31: CASOS DE USO GESTION DE TAREAS	159
TABLA 32: CASOS DE USO SESIÓN POMODORO	160
TABLA 33: INTERFACES REALES DEL SISTEMA	198

ÍNDICE DE IMÁGENES

IMAGEN 1: MODELO EN ESPIRAL	46
IMAGEN 2: CRONOGRAMA DE ACTIVIDADES TT 2024-A136.....	61
IMAGEN 3: ARQUITECTURA DEL SISTEMA DE TRES CAPAS.....	77
IMAGEN 4: DISEÑO IU2 PANTALLA DE REGISTRO DE USUARIO	79
IMAGEN 5: DISEÑO IU3 PANTALLA DE INICIO DE SESIÓN	80
IMAGEN 6: DISEÑO IU4 VISTA DE TAREAS EISENHOWER.....	80
IMAGEN 7: DISEÑO IU6 VISTA DE TAREAS CALENDARIO.....	81
IMAGEN 8: DISEÑO IU17 PANTALLA DE NOTAS	81
IMAGEN 9: DISEÑO IU23 PANTALLA DE CUENTA.....	82
IMAGEN 10: DISEÑO IU28 PANTALLA DE CONFIGURACIÓN DEL SISTEMA.....	82
IMAGEN 11: DISEÑO IU29 PANTALLA DE MÉTRICAS DE LA CUENTA.....	83
IMAGEN 12: DISEÑO IU14 PANTALLA DE TEMPORIZADOR POMODORO	83
IMAGEN 13: DIAGRAMA DE CASOS DE USO	85
IMAGEN 14: DESCARGA DE INSTALADOR NODE.JS.....	87
IMAGEN 15: CAPTURA DE PROYECTO ANGULAR CORRIENDO EN NAVEGADOR	88
IMAGEN 16: COMPONENTE DE REGISTRO DE USUARIO	89
IMAGEN 17: COMPONENTE DE INICIO DE SESIÓN	89
IMAGEN 18: COMPONENTE DE CUENTA DE USUARIO	89
IMAGEN 19:COMPONENTE DE CONFIGURACIÓN DE USUARIO	89
IMAGEN 20: URL DE CONEXIÓN CON LA BASE DE DATOS Y PRISMA	91
IMAGEN 21:ESQUEMA PRISMA CONEXIÓN DB	92
IMAGEN 22: ENTIDAD USUARIO PRISMA	92
IMAGEN 23:ENTIDAD CONFIGURACIÓN PRISMA.....	93
IMAGEN 24: CARPETA BACKEND INICIAL	93
IMAGEN 25: CONFIGURACIÓN DE EJECUCIÓN BACKEND	95
IMAGEN 26: CONFIGURACIÓN DE EJECUCIÓN FRONTEND	96
IMAGEN 27: COMANDO PARA EL CONSTRUIR LOS ARCHIVOS ESTÁTICOS DEL BACKEND	96
IMAGEN 28: COMANDO PARA EJECUCIÓN DEL BACKEND.....	96
IMAGEN 29: COMANDO PARA EJECUCIÓN DEL FRONTEND	97
IMAGEN 30: COMPONENTES DE CUENTA	100
IMAGEN 31: COMPONENTES DE NOTAS.....	101
IMAGEN 32:ENTIDAD NOTE PRISMA.....	101
IMAGEN 33: COMPONENTES DE TAREAS	106
IMAGEN 34:COMPONENTE DE MÉTRICAS	106
IMAGEN 35:COMPONENTE DE HISTORIAL DE EVENTOS.....	107
IMAGEN 36: ENTIDAD TASK PRISMA	107
IMAGEN 37: ENTIDAD EVENT PRISMA	108
IMAGEN 38: ENTIDAD CATEGORY PRISMA.....	108
IMAGEN 39: COMPONENTES DE TEMPORIZADOR POMODORO.....	112
IMAGEN 40: ENTIDAD POMODOROSESSION PRISMA	113

IMAGEN 41: ENTIDAD POMODOROEVENT PRISMA	113
IMAGEN 42: REPOSITORIO DE TEMPUS VICTOR	116
IMAGEN 43: INSTRUCCIONES PARA INSTALACIÓN DEL SISTEMA TEMPUS VICTOR.....	117
IMAGEN 44: PREGUNTA 1.....	118
IMAGEN 45: PREGUNTA 2.....	118
IMAGEN 46: PREGUNTA 3.....	119
IMAGEN 47: PREGUNTA 4.....	119
IMAGEN 48: PREGUNTA 5.....	119
IMAGEN 49: PREGUNTA 6.....	120
IMAGEN 50: PREGUNTA 7.....	120
IMAGEN 51: PREGUNTA 8.....	120
IMAGEN 52: PREGUNTA 9.....	121
IMAGEN 53: PREGUNTA 10.....	121
IMAGEN 54: PREGUNTA 11.....	121
IMAGEN 55: PREGUNTA 12.....	122
IMAGEN 56: PREGUNTA 13.....	122
IMAGEN 57: PREGUNTA 14.....	122
IMAGEN 58: DISEÑO DE LA BASE DE DATOS	148
IMAGEN 59: BASE DE DATOS IMPLEMENTADA.....	149
IMAGEN 60: DIAGRAMA DE CASOS DE USO.....	156
IMAGEN 61: CASOS DE USO GESTIÓN DE CUENTA.....	157
IMAGEN 62: CASOS DE USO GESTION DE NOTAS	158
IMAGEN 63: CASOS DE USO GESTION DE TAREAS.....	159
IMAGEN 64: CASOS DE USO GESTION DE SESIONES POMODORO	160

1. INTRODUCCIÓN

El presente documento es el reporte técnico final del Trabajo Terminal con el número 2024-A136 que tiene por nombre “Sistema Web para la Organización Personal”, correspondiente al análisis diseño y desarrollo del sistema “Tempus Victor” a lo largo del semestre 2024-1 y 2024-2 con el fin de acreditar la unidad de aprendizaje así como servir para opción de titulación de la carrera en Ingeniería en Sistemas Computacionales y poder obtener el título de Ingeniero en Sistemas Computacionales dentro de la Escuela Superior de Computo del Instituto Politécnico Nacional.

Este documento, ofrece una guía para sumergirse en el mundo de la organización personal y descubrir cómo este innovador sistema web puede convertirse en un aliado indispensable. En primera instancia, es muy común que dentro del ajetreado mundo académico la desorganización personal se convierte en el enemigo número uno del éxito estudiantil. Esto es porque una gran cantidad de estudiantes desorganizados se enfrentan a una batalla constante contra la falta de planificación, la mala gestión del tiempo, la sobrecarga de tareas, la pérdida de información y la dificultad para priorizar. Estas problemáticas se traducen en retrasos en la entrega de trabajos, bajas calificaciones, estrés, ansiedad y una sensación de agobio que afecta negativamente el rendimiento académico y el bienestar general.

Por lo cual, se presenta “**Tempus Victor**” el **Sistema web para la organización personal** enfocado a estudiantes, un proyecto desarrollado a lo largo de dos semestres, 2024-1 y 2024-2, con el fin de **brindar a estudiantes las herramientas necesarias para optimizar su rendimiento académico y alcanzar sus metas educativas**. Este sistema web no solo combate la desorganización, sino que también provee a los estudiantes de dos metodologías muy útiles para el seguimiento de tareas: **la Matriz Eisenhower y la Técnica Pomodoro**.

- **La Matriz Eisenhower** brinda la capacidad de clasificar tareas según su urgencia e importancia, permitiendo a los estudiantes enfocarse en las actividades que realmente impactan en su éxito académico.
- **La Técnica Pomodoro** introduce intervalos de trabajo concentrado de 25 minutos cada uno, seguidos de breves descansos, fomentando la concentración y la productividad.

La integración de estas dos metodologías en Tempus Victor lo convierte en una herramienta invaluable para estudiantes que buscan optimizar su tiempo, reducir el estrés y alcanzar sus metas académicas.

Una vez planteado el Sistema Web a desarrollar, se concretó la definición del objetivo general del Trabajo Terminal, el cual se consideró puntualmente para los cambios que se fueron realizando a lo largo del desarrollo.

Después se presentan los objetivos específicos que nacieron con la finalidad de ayudar y guiar al estudiante a solventar o minimizar las problemáticas identificadas, todo esto por medio del sistema Tempus Victor.

Posterior a ello se muestra la comparativa de diversos sistemas web de organización personal como aliados en la lucha contra la desorganización estudiantil. Entre ellos, encontramos Google Calendar, Todoist, Evernote, TickTick y Notion; cada uno con sus propias fortalezas y debilidades.

Después de que se fundamenta el Trabajo Terminal se presentan varios conceptos importantes necesarios para entender mejor el contexto en el que se desarrolló Tempus Victor, desde lo que es la organización personal, qué es una tarea, la diferencia entre tarea y otras actividades, hasta conceptos técnicos que ayudarán en la comprensión de los capítulos posteriores.

Continuando con el documento se presenta la metodología con la que se desarrolló el sistema web Tempus Victor, la cual tiene por nombre “Modelo en espiral”, una metodología iterativa que permite ajustes y mejoras continuas a lo largo del proceso. El modelo en espiral se aplicó en seis fases: Planificación, Análisis, Diseño, Implementación, Pruebas y Evaluación.

Lo que sigue es la explicación de actividades sobre el trabajo realizado a lo largo de los dos semestres del Trabajo Terminal, dividiendo el trabajo en un total de 5 iteraciones.

- La primera iteración contiene el planteamiento de la problemática y su análisis, la propuesta de solución, la investigación del estado del arte, la identificación de actores, desarrollo del marco teórico, identificación de requerimientos funcionales, análisis y diseño de diagrama relacional, análisis de riesgo, análisis de factibilidad económica, análisis de factibilidad técnica.
- La segunda iteración contiene el diseño de interfaces gráficas, planteamiento de casos de uso, identificación y descripción de reglas del negocio, desarrollo de diagrama de

clases, desarrollo de diagrama de procesos, fase de implementación, fase de pruebas y fase de corrección.

- Las tres iteraciones restantes conforman el desarrollo del sistema web Tempus Victor; estas iteraciones contienen el desarrollo de interfaces gráficas, desarrollo de casos de uso, desarrollo de diagramas de clases, ajustes y valoración de reglas de negocio, desarrollo de Frontend, desarrollo de Backend, fase de implementación, fase de pruebas y fase de corrección.

Casi finalizando el documento se encuentran los resultados a los que se llegaron con el Trabajo Terminal, en esta parte se explican lo que, si se logró alcanzar y lo que faltó por completar, también las razones y justificaciones para dichos resultados y conclusiones. De la misma manera, se presentan datos recolectados en la parte final y pruebas con usuarios reales, permitiendo construir conclusiones propias sobre el desarrollo de este sistema web Tempus Victor.

Para finalizar el documento se agregan la parte de anexos, la cual incluye en forma extendida lo que se explica en capítulos anteriores: los análisis, los modelos, los diagramas, los casos de uso y sobre todo las pantallas del sistema.

2. PLANTEAMIENTO DEL PROYECTO

En este capítulo se presenta en detalle el planteamiento del proyecto Tempus Victor, un sistema web de organización personal diseñado específicamente para estudiantes. Se aborda la problemática de la desorganización estudiantil, una realidad que afecta negativamente el rendimiento académico y genera estrés en los estudiantes.

Tempus Victor surge como una solución innovadora a esta problemática, ofreciendo un sistema web que integra metodologías como la Técnica Pomodoro y la Matriz Eisenhower para optimizar la gestión del tiempo y las tareas. Se definen con claridad los objetivos del proyecto, tanto el objetivo general como los objetivos específicos que guiarán el desarrollo del sistema.

Se describen las características principales de Tempus Victor, resaltando las ventajas que ofrece a los estudiantes, la integración de metodologías útiles, el seguimiento en tiempo real, la personalización, el análisis de productividad y la interfaz intuitiva.

A lo largo de este capítulo, se construye una sólida base para el desarrollo de Tempus Victor, sentando las bases para un sistema web que realmente impacte positivamente en la vida académica de los estudiantes.

2.1. Planteamiento del Problema

En el mundo actual, marcado por un ritmo de vida acelerado y múltiples demandas, la organización personal se ha convertido en una habilidad fundamental para el éxito en diversos ámbitos, incluyendo el académico.

Los estudiantes, en particular, enfrentan una infinidad de desafíos para gestionar su tiempo, tareas y responsabilidades, lo que puede impactar negativamente en su rendimiento, bienestar y motivación. En este contexto, surge la necesidad de abordar las problemáticas relacionadas con la falta de organización personal en los estudiantes.

Esta sección tiene como objetivo analizar las principales dificultades que enfrentan los estudiantes actualmente, describiendo sus diferentes manifestaciones y consecuencias. A partir de este análisis, se establecerá una problemática general que sintetiza las necesidades y desafíos que requieren atención para promover hábitos en la población estudiantil.

2.1.1 Problemáticas específicas

La falta de planificación efectiva:

La falta de planificación a menudo surge de la falta de claridad sobre los objetivos y pasos necesarios para alcanzarlos. (Saurer, 2021).

Los estudiantes a menudo se encuentran luchando contra la dificultad para establecer planes claros y estructurados que les permitan alcanzar sus metas académicas. Esta falta de planificación efectiva se manifiesta en la ausencia de objetivos definidos, el desconocimiento de técnicas adecuadas de organización y la carencia de habilidades para gestionar el tiempo de manera eficiente.

Es muy común el que un estudiante no planifique su estudio para un examen importante, al acercarse la fecha límite el estrés aumenta; no sabe por dónde empezar, qué contenidos revisar o cuánto tiempo dedicar a cada tema. La falta de un plan lo lleva a estudiar a última hora, sintiéndose abrumado y con dificultad para comprender los contenidos; esto, sin duda, afecta negativamente su desempeño en el examen.

Las consecuencias de la falta de planificación efectiva son evidentes:

- **Procrastinación:** La postergación de tareas importantes se convierte en un hábito común, generando retrasos, estrés y ansiedad.
- **Desorganización:** La falta de un plan estructurado dificulta la organización de las tareas, el tiempo y los recursos, lo que lleva a la pérdida de tiempo y la sensación de caos.
- **Agotamiento:** La sobrecarga de tareas sin un plan definido para abordarlas genera agobio, estrés y afecta la salud mental y el rendimiento académico.

Mala gestión del tiempo:

Cuando no se sabe cómo administrar el tiempo de manera eficiente, es más probable que se pierdan oportunidades y que se acumulen las tareas pendientes (Moreno, 2020).

Muchos estudiantes experimentan la dificultad para distribuir el tiempo de manera eficiente entre las diferentes tareas y responsabilidades académicas. Esta mala gestión del tiempo se debe a la falta de habilidades para priorizar, la ausencia de herramientas adecuadas para la gestión del tiempo o la dificultad para controlar las distracciones.

Un estudiante que no sabe priorizar sus tareas dedica más tiempo a actividades menos importantes, dejando las tareas más urgentes para el último momento. Se acerca la fecha de

entrega de un trabajo importante, pero no ha avanzado lo suficiente. La presión aumenta, el estrés se apodera de él y la calidad del trabajo se ve afectada. Las repercusiones de la mala gestión del tiempo son notables:

- Retrasos: El incumplimiento de plazos y fechas límite se vuelve habitual, lo que puede afectar negativamente la calidad del trabajo y la reputación académica.
- Estrés y ansiedad: La sensación de no tener tiempo suficiente para completar todas las tareas genera estrés, ansiedad y afecta la salud mental.
- Disminución de la productividad: La mala gestión del tiempo reduce la eficiencia en el estudio y el trabajo, lo que impacta negativamente en el rendimiento académico.

Pérdida de información:

Cuando no se tiene un sistema organizado para guardar información importante, es difícil encontrar lo que se necesita cuando se necesita (Milbrath, 2021).

La pérdida de información es un problema común que afecta a muchos estudiantes, quienes se encuentran con la dificultad para mantener un registro organizado de datos importantes, como fechas de entrega, apuntes de clase, materiales de estudio o información de contacto. Esto puede deberse a la falta de un sistema de organización adecuado, la desorganización física de los espacios de estudio o la pérdida de dispositivos electrónicos.

En el caso de un estudiante que no cuenta con un sistema de organización para sus apuntes de clase, a medida que avanzan las semanas, sus cuadernos se llenan de notas desordenadas, sin una estructura clara. Llega el momento de estudiar para un examen importante y no puede encontrar la información que necesita. La frustración lo invade, el estrés aumenta y su rendimiento académico se ve afectado. Las consecuencias de la pérdida de información son significativas:

- Retrasos y errores: La falta de acceso a información importante puede generar retrasos en la entrega de tareas, errores en los trabajos académicos y dificultades para comprender los contenidos.
- Frustración y estrés: La pérdida de información importante puede generar frustración, ansiedad y afectar la salud mental del estudiante.
- Disminución de la eficiencia: La dificultad para encontrar la información necesaria para el estudio reduce la eficiencia y aumenta el tiempo dedicado a tareas improductivas.

Dificultad para priorizar:

La incapacidad para distinguir entre tareas urgentes e importantes contribuye a retrasos y frustraciones. (Pérez, 2016).

La priorización efectiva es una habilidad fundamental para el éxito en cualquier ámbito, y el entorno académico no es una excepción. Sin embargo, muchos estudiantes experimentan dificultades para distinguir entre tareas urgentes e importantes, lo que los lleva a retrasos, frustraciones y una disminución en la eficiencia. Normalmente esta incapacidad puede llevar a concentrarse en las cosas equivocadas; cuando no se sabe qué tareas son las más importantes, es más probable que se desperdicie tiempo y recursos.

En otro caso, un estudiante que tiene una lista de tareas por completar, pero no sabe por dónde empezar, no sabe cuáles son las tareas más importantes que requieren su atención inmediata y cuáles pueden posponerse un poco. Dedica tiempo a tareas menos relevantes, mientras que las tareas urgentes se acumulan y la fecha límite se acerca. El estrés aumenta, la frustración lo invade y la eficiencia en su trabajo se ve afectada; las consecuencias de la dificultad para priorizar son considerables:

- Desconcentración: La falta de claridad sobre las tareas prioritarias genera dificultades para concentrarse y mantener el enfoque en los objetivos académicos.
- Pérdida de tiempo: Dedicar tiempo a tareas menos importantes o postergar las tareas urgentes significa perder tiempo valioso que podría ser utilizado de manera más eficiente.
- Frustración: La sensación de no poder cumplir con todas las tareas a tiempo y la acumulación de responsabilidades generan frustración, estrés y ansiedad.

Las problemáticas específicas que se han descrito en este capítulo representan un conjunto de desafíos significativos que obstaculizan el éxito académico y el bienestar de los estudiantes. La falta de planificación efectiva, la mala gestión del tiempo, la sobrecarga de tareas, la pérdida de información y la dificultad para priorizar son solo algunos ejemplos de las dificultades que enfrentan día a día.

Es necesario tomar medidas contundentes para abordar estas problemáticas de manera integral y promover hábitos de organización personal efectivos en los estudiantes. No se puede seguir ignorando las dificultades que enfrentan y el impacto negativo que estas tienen en su rendimiento académico, su salud mental y su motivación.

2.1.2 Problemática General

En el mundo actual, marcado por un ritmo de vida acelerado y una gran cantidad de información, la organización personal se ha convertido en una habilidad fundamental para el éxito en diversos ámbitos, incluyendo el académico. Sin embargo, muchas personas, especialmente estudiantes, experimentan dificultades significativas para gestionar sus tareas, tiempo y responsabilidades de manera efectiva.

Las personas organizadas tienen más probabilidades de completar sus tareas a tiempo y con un alto nivel de calidad.

La falta de habilidades y hábitos de organización personal se manifiesta en una serie de desafíos que obstaculizan el logro de metas y objetivos. Planificar, gestionar, dar seguimiento, priorizar y guardar información sobre múltiples tareas se convierte en una lucha constante, generando consecuencias negativas como:

- Retrasos y errores: La dificultad para planificar y gestionar el tiempo de manera efectiva conduce a retrasos en la entrega de tareas, errores en el trabajo y una disminución en la calidad del rendimiento.
- Estrés y ansiedad: La acumulación de tareas pendientes, la sensación de no tener tiempo suficiente y la falta de control sobre las responsabilidades generan estrés, ansiedad y afectan negativamente la salud mental.
- Disminución de la productividad: La desorganización y la dificultad para enfocarse en las tareas importantes reducen la eficiencia en el trabajo y el estudio, impactando negativamente en el rendimiento académico y profesional.
- Agotamiento físico y mental: La sobrecarga de tareas y la constante lucha por cumplir con las responsabilidades pueden generar fatiga física, estrés crónico y problemas de salud mental.

Teniendo esto en cuenta se puede resumir la problemática general en lo siguiente:

“La problemática general radica en la falta de habilidades y hábitos de organización personal, lo que dificulta a los estudiantes gestionar de manera efectiva sus tareas, tiempo y responsabilidades en un entorno cada vez más complejo y demandante. Esta situación afecta negativamente su rendimiento académico, su bienestar emocional y su calidad de vida en general. Todo esto porque no es fácil planificar, gestionar, dar seguimiento, priorizar y guardar información sobre múltiples tareas.”

En la siguiente sección se describe la solución que se pretende desarrollar para ayudar a los estudiantes con esta problemática identificada. Esto sentará las bases para la propuesta de un sistema web de organización personal, diseñado específicamente para atender las necesidades de los estudiantes y ayudarles a superar los obstáculos que enfrentan en materia de organización personal, permitiéndoles alcanzar sus metas académicas y mejorar su calidad de vida.

2.2. Propuesta de Solución

En el ámbito educativo, la gestión del tiempo y la organización personal son habilidades fundamentales para el éxito de los estudiantes. La falta de estas habilidades puede generar dificultades para cumplir con tareas y fechas límite, lo que puede traducirse en estrés, ansiedad y un rendimiento académico deficiente.

En respuesta a dicha necesidad, se propone una posible solución el desarrollo de un sistema web de organización personal capaz de proporcionar herramientas personalizables para hacer seguimiento de tareas, es por lo que nace "**Tempus Victor**", un sistema web de organización personal para estudiantes orientado en tiempo y tareas. Este sistema web no solo combate la desorganización, sino que también provee a los estudiantes de dos metodologías muy útiles para el seguimiento de tareas: la **Matriz Eisenhower** y la **Técnica Pomodoro**.

- La **Matriz Eisenhower** brinda a los estudiantes la capacidad de clasificar sus tareas según su urgencia e importancia, permitiéndoles enfocarse en las tareas que realmente impactan en su éxito académico.
- La **Técnica Pomodoro** introduce intervalos de trabajo concentrado de 25 minutos cada uno, seguidos de breves descansos, fomentando la concentración y la productividad. La integración de estas dos metodologías en **Tempus Victor** lo convierte en una herramienta invaluable para estudiantes que buscan optimizar su tiempo, reducir el estrés y alcanzar sus metas académicas.

En la siguiente sección se explica que significa el nombre propuesto para el sistema web así como su descripción.

2.2.1 Tempus Victor nombre del sistema Web

Significado:

- Tempus: La palabra latina para "tiempo".
- Victor: La palabra latina para "vencedor" o "conquistador".

Descripción:

Tempus Victor es un nombre poderoso y evocador que transmite la idea de conquistar o dominar el tiempo. Sugiere una persona que toma el control de su tiempo y lo utiliza de forma eficaz y eficiente.

El nombre también puede interpretarse como una referencia a la superación de los obstáculos y desafíos que el tiempo presenta. Implica una actitud resiliente y victoriosa ante las limitaciones temporales.

Atributos:

- Poder: El nombre emana fuerza y determinación.
- Control: Evoca la idea de tomar las riendas del tiempo.
- Eficiencia: Sugiere un enfoque efectivo en la gestión del tiempo.
- Resiliencia: Transmite una actitud positiva ante los desafíos temporales.
- Victoria: Inspira a los usuarios a alcanzar sus objetivos y metas.

En la siguiente sección se definen y plasman los objetivos para el trabajo terminal, así como para el desarrollo del sistema web de organización personal **Tempus Victor**.

2.3. **Objetivo General**

Dado el trabajo de análisis realizado de acuerdo con la investigación sobre problemáticas y las posibles soluciones, se planteó el objetivo con la siguiente redacción:

“Diseñar y desarrollar un sistema web que proporcione apoyo para la organización personal haciendo uso de herramientas personalizables enfocadas en tiempo y tareas.”

En la siguiente sección se expondrán los objetivos específicos que se consideran esenciales para completar el desarrollo de Tempus Victor.

2.4. **Objetivos Específicos**

El sistema de organización personal debe proporcionar a los usuarios las herramientas necesarias para gestionar su tiempo y tareas. Para ello, se plantean los siguientes objetivos específicos:

- Desarrollar un **módulo de notas** que permita a los usuarios crear, editar, eliminar notas simples, archivar notas y consultar notas archivadas.

- Desarrollar un **módulo de tareas** que permita a los usuarios crear, editar, eliminar, asignar fecha y hora límite, prioridad y categoría a las tareas.
- Desarrollar un **módulo de calendario** que permita ver las tareas en una vista de calendario en donde se organizará por fechas, se tendrá un apartado para las tareas sin fecha límite, y para las tareas de día. Así como acceso a todas las opciones posibles del módulo de tareas.
- Desarrollar un **módulo de matriz** de Eisenhower que permita ver las tareas en una matriz en donde se presentan priorizar las tareas dependiendo la urgencia e importancia para el usuario.
- Desarrollar un **módulo de temporizador** Pomodoro para ayudar a los usuarios a concentrarse en las tareas, facilitando la segmentación del trabajo en intervalos con pausas planificadas, haciendo un seguimiento de los tiempos dedicados a cada tarea. También poder crear y ver un historial de tiempos para análisis detallado y retroalimentación al usuario.
- Desarrollar un **módulo de personalización** que permita a los usuarios adaptar el sistema a sus necesidades específicas. Las personalizaciones deben incluir la duración de intervalos de trabajo y descanso, las opciones de recordatorios y la interfaz del sistema.
- Desarrollar un **módulo de métricas** de productividad para proporcionar información sobre la productividad de los usuarios. La información debe incluir todas las tareas de la cuenta del estudiante. Así como la información relacionada a las sesiones Pomodoro.

Alcanzar estos objetivos específicos permitirá cumplir con el objetivo general de desarrollar un sistema web de organización personal que contribuya significativamente al éxito académico y el bienestar de los estudiantes en general.

En la siguiente sección se expone la justificación para este sistema web que se propone diseñar y desarrollar, haciendo énfasis en las necesidades específicas y los beneficios potenciales para los estudiantes.

2.5. Justificación

En el ámbito académico, la gestión del tiempo y la organización personal son habilidades cruciales para el éxito de los estudiantes. La falta de estas habilidades puede generar dificultades para cumplir con tareas antes de sus fechas límite, lo que puede traducirse en estrés, ansiedad y un rendimiento académico deficiente.

Los estudiantes enfrentan una gran cantidad de tareas, responsabilidades y compromisos académicos. Deben gestionar su tiempo entre clases, tareas, proyectos, exámenes y actividades extracurriculares. La falta de herramientas adecuadas para organizar y planificar estas actividades puede generar:

- Dificultades para cumplir con fechas límite: Esto puede traducirse en entregas tardías, penalizaciones y una reducción en la calificación final.
- Estrés y ansiedad: La sensación de desorganización y la presión por cumplir con múltiples tareas pueden generar estrés y ansiedad, afectando negativamente la salud mental y el bienestar de los estudiantes.
- Disminución del rendimiento académico: La falta de organización puede dificultar la concentración, el estudio efectivo y la comprensión de los conceptos, lo que puede llevar a un rendimiento académico deficiente.

El sistema web de organización personal que se propone surge como una herramienta innovadora y personalizada diseñada específicamente para estudiantes, con el fin de brindarles las herramientas necesarias para optimizar su tiempo, enfocarse en sus tareas y alcanzar sus metas académicas.

En las siguientes secciones, se detallan las principales razones que justifican la creación de este sistema para estudiantes.

2.5.1 Combatir la falta de planificación en estudiantes

La planificación es fundamental para el éxito académico. El sistema web de organización personal para estudiantes incluye:

- Módulo de calendario: Permite visualizar las tareas y fechas límite en un solo lugar, facilitando la organización del tiempo y la identificación de posibles conflictos de horario.

2.5.2 Superar la mala gestión del tiempo en estudiantes

La procrastinación y la mala gestión del tiempo son problemas comunes entre los estudiantes. El sistema web de organización personal ofrece herramientas para combatir estos problemas:

- Módulo de temporizador Pomodoro: Basado en la técnica Pomodoro, este módulo ayuda a los estudiantes a concentrarse en sus estudios mediante la segmentación del trabajo en intervalos de 25 minutos cada uno, con pausas planificadas. Esto les permite mantenerse enfocados, reducir la distracción y aumentar la productividad durante el estudio.
- Módulo de métricas de estudio: Este módulo proporciona información detallada sobre el tiempo dedicado al estudio, las tareas a las que se ha dedicado más tiempo y la cantidad de tareas completadas. Esta información permite a los estudiantes identificar áreas de mejora y ajustar sus estrategias de estudio.
- Módulo de personalización: El sistema es altamente personalizable, permitiendo a los estudiantes adaptar la interfaz, las herramientas y las notificaciones a sus necesidades y preferencias específicas de estudio; garantizando una experiencia de usuario óptima y una mayor efectividad en la gestión del tiempo de estudio.

2.5.3 Minimizar la pérdida de información académica

La información académica es un recurso valioso que debe ser gestionado de manera eficiente. El sistema web de organización personal para estudiantes ofrece:

- Módulo de gestión de tareas: Permite crear, editar, eliminar, asignar fecha y hora límite, prioridad y categoría a las tareas. Esto facilita la organización y el seguimiento de las tareas pendientes, asegurando que no se olviden si son importantes.
- Módulo de notas de estudio: Permite crear, editar y buscar, garantizando que la información esté accesible y organizada en un solo lugar.

2.5.4 Facilitar la priorización de tareas académicas

La priorización de tareas académicas es esencial para enfocarse en lo que realmente importa. El sistema web de organización personal para estudiantes incluye:

- Módulo de matriz de Eisenhower para tareas: Esta herramienta permite clasificar las tareas académicas según su importancia y urgencia, facilitando la identificación de las tareas que requieren atención inmediata y las que pueden posponerse. Esto ayuda a los

estudiantes a optimizar su tiempo y enfocarse en las tareas que tienen mayor impacto en su rendimiento académico.

2.5.5 [Beneficios del sistema web para estudiantes](#)

La implementación de este sistema web de organización personal para estudiantes tiene el potencial de ofrecer beneficios significativos:

Mejora del rendimiento académico: El sistema puede ayudar a los estudiantes a ser más productivos y eficientes en su tiempo de estudio, permitiéndoles:

- Mejorar la comprensión de los conceptos: Al optimizar su tiempo de estudio y enfocarse en las tareas más importantes, los estudiantes pueden dedicar más tiempo a comprender a fondo los conceptos y materias que se imparten en clase.
- Completar tareas a tiempo: La organización y planificación del tiempo de estudio permiten a los estudiantes cumplir con las fechas límite de tareas, proyectos y exámenes, evitando penalizaciones y reduciendo el estrés asociado a las entregas tardías.
- Obtener mejores calificaciones: Un mejor rendimiento académico, producto de la mayor comprensión de los conceptos, la eficiencia en el estudio y el cumplimiento oportuno de las tareas se traduce en mejores calificaciones y un mayor éxito en el ámbito académico.

Reducción del estrés y la ansiedad: La organización y la planificación pueden ayudar a reducir el estrés y la ansiedad asociados con la falta de control sobre las tareas académicas y la presión por cumplir con múltiples compromisos. El sistema web de organización personal para estudiantes puede contribuir a:

- Disminuir la sensación de agobio: Al tener una visión clara de las tareas pendientes, las fechas límite y el tiempo disponible, los estudiantes pueden sentirse más seguros y menos abrumados por sus responsabilidades académicas.
- Mejorar la calidad del sueño: Una adecuada gestión del tiempo de estudio y la organización de las tareas pueden permitir a los estudiantes conciliar el sueño de manera más regular y tener un descanso adecuado, lo cual es fundamental para su salud mental y física.

- Aumentar la motivación: El cumplimiento de metas académicas y la obtención de buenas calificaciones pueden aumentar la motivación de los estudiantes, impulsándolos a continuar esforzándose y buscando su superación personal.

Impacto positivo a largo plazo: El desarrollo de hábitos de organización, planificación y gestión del tiempo durante la etapa de estudios puede tener un impacto positivo a largo plazo en la vida de los estudiantes:

- Habilidades transferibles: La capacidad de organizar y gestionar el tiempo de manera efectiva son habilidades valiosas que pueden ser aplicadas en diversos ámbitos de la vida, como el ámbito laboral, personal y social.
- Mejora de la autoestima: El éxito académico y la sensación de control sobre las responsabilidades pueden contribuir a mejorar la autoestima y la confianza en sí mismos de los estudiantes.
- Preparación para el futuro: Desarrollar hábitos de organización y responsabilidad durante los estudios puede preparar a los estudiantes para afrontar los desafíos y responsabilidades que enfrentarán en su vida profesional y personal.

2.5.6 Relevancia de la tarea dentro del Trabajo Terminal

La definición precisa de "tarea" es fundamental para el desarrollo del sistema web para la organización personal por las siguientes razones:

- Unidad fundamental de trabajo: La tarea es la base del sistema. Todo lo que el sistema hace gira en torno a la gestión de tareas, desde su creación y organización hasta su seguimiento y evaluación.
- Elemento central de la organización personal: La organización personal se basa en la gestión eficiente de las tareas. El sistema web busca proporcionar herramientas para que los usuarios puedan organizar sus tareas y alcanzar sus objetivos.
- Impacto en el desarrollo del sistema: La definición de la tarea determina las funcionalidades del sistema. Los módulos y herramientas que se desarrollarán dependerán de las características y necesidades de las tareas que el sistema gestione.
- Enfoque en la usabilidad: El sistema debe ser intuitivo y fácil de usar para que los usuarios puedan crear, editar y gestionar sus tareas sin complicaciones.

En resumen, la definición precisa de "tarea" es crucial para el éxito del sistema web para la organización personal.

2.5.7 Relevancia de Pomodoro dentro del Trabajo Terminal

La técnica Pomodoro es una herramienta de gestión del tiempo que se basa en dividir el trabajo en intervalos de 25 minutos (Pomodoros), separados por pausas breves. Su implementación en el Trabajo Terminal tiene las siguientes ventajas:

- Mejora la concentración: La técnica Pomodoro ayuda a los usuarios a concentrarse en la tarea que están realizando durante 25 minutos, lo que aumenta la productividad y reduce la procrastinación.
- Favorece la organización del tiempo: La división del trabajo en intervalos fijos ayuda a los usuarios a distribuir su tiempo de forma eficiente y a evitar la sensación de agobio.
- Reduce el estrés: Las pausas regulares que forman parte de la técnica Pomodoro ayudan a reducir el estrés y la fatiga mental, mejorando el bienestar del usuario.
- Se integra fácilmente al sistema web: La técnica Pomodoro se puede integrar fácilmente al sistema web mediante la creación de un módulo que gestione los intervalos de trabajo y las pausas.

En resumen, la técnica Pomodoro es una herramienta valiosa que puede mejorar la productividad, la organización del tiempo y el bienestar de los usuarios del sistema web para la organización personal.

2.5.8 Relevancia de matriz Eisenhower dentro del TT

La matriz Eisenhower es una herramienta de priorización que ayuda a los usuarios a clasificar sus tareas en cuatro categorías: Importante y Urgente, Importante y No Urgente, No Importante y Urgente, y No Importante y No Urgente; su implementación en el Trabajo Terminal ofrece los siguientes beneficios:

- Priorización efectiva de tareas: La matriz Eisenhower ayuda a los usuarios a determinar la importancia y urgencia de cada tarea, permitiéndoles enfocarse en las que realmente importan.
- Optimización del tiempo y la energía: Al enfocarse en las tareas importantes, los usuarios pueden optimizar su tiempo y energía, evitando la pérdida de tiempo en tareas irrelevantes.

- Reducción del estrés: La priorización de tareas ayuda a reducir el estrés y la ansiedad, ya que los usuarios se sienten más seguros de que están trabajando en las tareas correctas.
- Integración con el sistema web: La matriz Eisenhower se puede integrar al sistema web mediante la creación de un módulo que permita a los usuarios clasificar sus tareas según la matriz.

En resumen, la matriz Eisenhower es una herramienta útil para que los usuarios del sistema web para la organización personal puedan priorizar sus tareas de forma efectiva y optimizar su tiempo y energía.

Como se describió anteriormente, la creación de un sistema web de organización personal para estudiantes está justificada por las múltiples necesidades y desafíos que enfrentan en su vida académica. Este sistema tiene el potencial de mejorar significativamente la gestión del tiempo, la organización y la priorización de tareas, lo cual puede traducirse en un rendimiento académico superior, reducción del estrés y ansiedad, y una mayor satisfacción personal. Además, la integración de metodologías como la técnica Pomodoro y la matriz Eisenhower proporcionará a los estudiantes herramientas efectivas para optimizar su tiempo y esfuerzos, preparándolos mejor para futuros desafíos tanto académicos como profesionales.

En la siguiente sección se presenta la comparativa de diversos sistemas web de organización personal como aliados en la lucha contra la desorganización estudiantil. Entre ellos, encontramos Google Calendar, Todoist, Evernote, TickTick y Notion, cada uno con sus propias fortalezas y debilidades.

2.6. Estado del Arte

En el contexto de la desorganización estudiantil, el uso de un sistema que pueda resolver la problemática planteada es de gran ayuda para los estudiantes, es por esta razón que las siguientes características son consideradas útiles para un sistema de organización personal según la investigación y la propuesta realizada anteriormente.

- A) **Contar con herramientas para la planificación efectiva:** Esto puede a los estudiantes a establecer objetivos claros, organizar su tiempo, crear planes de estudio y gestionar sus tareas de manera eficiente.
- B) **Contar con estrategias para la gestión del tiempo:** Esto puede hacer que los estudiantes aprendan técnicas para priorizar tareas, evitar distracciones y optimizar su tiempo de estudio.
- C) **Implementar métodos para la organización de la información:** Esto puede hacer que los estudiantes aprendan técnicas para mantener un registro organizado de apuntes, materiales de estudio, fechas de entrega y otra información importante.
- D) **Brindar recursos para el desarrollo de la priorización:** Esto puede ayudar a los estudiantes a identificar las tareas urgentes e importantes, establecer un orden de prioridades y tomar decisiones efectivas sobre cómo gestionar su tiempo.
- E) **Contar con plataforma web:** Esto puede lograr que cualquier usuario pueda acceder de forma sencilla a través de un navegador a su sitio web.
- F) **Implementar la técnica Pomodoro:** Puede ayudar al usuario a dar seguimiento en tiempo real de una tarea, con la finalidad de mejorar el estudio.
- G) **Implementar la matriz Eisenhower:** Puede ayudar al usuario a usar una correcta priorización de manera fácil y rápida, dependiendo de la urgencia o la importancia.

Considerando la premisa de que un sistema efectivo debe integrar las características mencionadas anteriormente, se realizó un estudio comparativo de los sistemas web disponibles en el mercado que cumplen con algunas de estas características, así como del sistema web de organización personal propuesto.

En la **Tabla 1** se presentan los sistemas web seleccionados, indicando qué características cumplen y cuáles no, así como otros factores relevantes que influyen en su efectividad y utilidad para los estudiantes. Esta comparación nos permite identificar las fortalezas y debilidades de cada

sistema, proporcionando una visión clara de las mejores opciones disponibles y de cómo el sistema propuesto puede ofrecer una solución más completa y eficaz.

Tabla 1:Sistemas web de organización personal existentes

Sistema	A	B	C	D	E	F	G	Otros Factores
Google Calendar	✓	✗	✓	✗	✓	✗	✗	Sincronización con Gmail, integración con Google Drive, gratuito para uso básico
Todoist	✓	✓	✓	✗	✓	✗	✗	Aplicación móvil, gestión de tareas colaborativas, plan premium con funciones avanzadas
Evernote	✓	✗	✓	✓	✓	✗	✗	Almacenamiento de notas, escaneo de documentos, integración con web clipper
TickTick	✓	✗	✓	✓	✗	✗	✗	Aplicación móvil, gestión de tareas por listas, integración con calendario
Notion	✓	✓	✓	✓	✓	✗	✗	Wiki personal, gestión de proyectos, integración con múltiples aplicaciones
Tempus Victor	✓	✓	✓	✓	✓	✓	✓	Enfoque en la gestión del tiempo y tareas, técnica Pomodoro, matriz Eisenhower

Nota: Elaboración propia

Todos los sistemas web de organización personal mencionados ofrecen características similares entre sí, sin embargo, cada sistema tiene sus propias fortalezas y debilidades. A continuación, se describe brevemente cada uno de los sistemas analizados.

2.6.1 Google Calendar:

Google Calendar es un calendario en línea gratuito que permite a los usuarios gestionar sus citas, eventos y tareas de manera eficiente. Ofrece una amplia gama de funcionalidades, incluyendo:

- Creación de eventos recurrentes: Permite programar eventos que se repiten con regularidad, como citas médicas o clases.
- Establecimiento de recordatorios: Envía notificaciones por correo electrónico o alertas en el calendario para que los usuarios no olviden sus compromisos.
- Compartir calendarios: Permite compartir calendarios con otros usuarios para colaborar en proyectos o eventos grupales.
- Integración con otros servicios de Google: Se sincroniza con Gmail, Google Drive y Google Docs, facilitando la gestión del tiempo y la organización personal.

Fortalezas:

- Gratuito y accesible para todos los usuarios.
- Interfaz simple y fácil de usar.
- Amplia gama de funcionalidades para la gestión de eventos y tareas.
- Integración con otros servicios de Google.

Debilidades:

- Funcionalidades limitadas para la gestión de tareas en comparación con otras aplicaciones.
- No ofrece herramientas avanzadas para la priorización y organización de tareas complejas.
- Opciones de personalización limitadas.

2.6.2 Todoist:

Todoist es una aplicación de gestión de tareas que ayuda a los usuarios a organizar, priorizar y completar sus tareas de manera eficiente. Ofrece una amplia gama de funcionalidades, incluyendo:

- Creación y organización de tareas: Permite crear listas de tareas, establecer fechas límite y priorizar tareas según su importancia.
- Recordatorios y notificaciones: Envía notificaciones por correo electrónico, alertas en el dispositivo móvil o notificaciones push para recordar a los usuarios sus tareas pendientes.
- Subtareas y comentarios: Permite dividir tareas complejas en subtareas más pequeñas y agregar comentarios para colaborar con otros usuarios.
- Colaboración en equipo: Permite compartir tareas y listas de tareas con otros usuarios para trabajar en proyectos de manera colaborativa.

Fortalezas:

- Amplia gama de funcionalidades para la gestión de tareas.
- Herramientas avanzadas para la priorización y organización de tareas complejas.
- Opciones de personalización para adaptar la aplicación a las necesidades de cada usuario.
- Posibilidad de colaborar en equipo en proyectos.

Debilidades:

- La versión gratuita tiene algunas limitaciones, como el número máximo de tareas y colaboradores.
- La interfaz puede resultar un poco compleja para usuarios principiantes.
- No ofrece herramientas específicas para la gestión del tiempo, como la técnica Pomodoro.

2.6.3 Evernote:

Evernote es una aplicación de gestión de notas que permite a los usuarios capturar, organizar y compartir ideas, información y contenido importante. Ofrece una amplia gama de funcionalidades, incluyendo:

- Creación y edición de notas: Permite crear notas de texto, adjuntar archivos multimedia, como imágenes y audios, y escanear documentos.
- Organización de notas: Permite organizar las notas en libretas, etiquetas y pilas para una mejor clasificación.
- Búsqueda avanzada: Permite buscar notas por texto, etiquetas, fecha de creación y otros criterios.
- Compartir notas: Permite compartir notas con otros usuarios de forma pública o privada.

Fortalezas:

- Versatilidad para capturar y organizar todo tipo de información.
- Potentes herramientas de búsqueda para encontrar notas rápidamente.
- Posibilidad de compartir notas con otros usuarios.
- Integración con diversos dispositivos y plataformas.

Debilidades:

- Funcionalidades limitadas para la gestión de tareas y la organización del tiempo.
- La interfaz puede resultar un poco abrumadora para usuarios principiantes.
- La versión gratuita tiene algunas limitaciones, como el almacenamiento de notas.

2.6.4 TickTick:

TickTick es una aplicación de gestión de tareas que ayuda a los usuarios a organizar, priorizar y completar sus tareas de manera eficiente. Ofrece una amplia gama de funcionalidades:

- Creación y organización de tareas: Permite crear listas de tareas, establecer fechas límite, priorizar tareas según su importancia y agregar subtareas.
- Recordatorios y notificaciones: Envía notificaciones por correo electrónico, alertas en el dispositivo móvil o notificaciones push para recordar a los usuarios sus tareas pendientes.
- Colaboración en equipo: Permite compartir tareas y listas de tareas con otros usuarios para trabajar en proyectos de manera colaborativa.
- Análisis de productividad: Brinda informes y estadísticas sobre el rendimiento del usuario, como el tiempo dedicado a cada tarea y el número de tareas completadas.

Fortalezas:

- Amplia gama de funcionalidades para la gestión de tareas.
- Herramientas específicas para la gestión del tiempo.
- Análisis de productividad para mejorar el rendimiento.
- Integración con otras aplicaciones populares.
- Interfaz simple y fácil de usar.

Debilidades:

- La versión gratuita tiene algunas limitaciones, como el número máximo de tareas y colaboradores.
- Algunas funcionalidades avanzadas, como la gestión de subtareas y la colaboración en equipo, solo están disponibles en la versión premium.

2.6.5 Notion:

Notion es una aplicación de productividad que permite a los usuarios gestionar sus tareas, notas, proyectos y más. Ofrece una gran flexibilidad y posibilidades de personalización, lo que la convierte en una buena opción para usuarios con necesidades complejas. Funcionalidades:

- Gestión de tareas: Permite crear listas de tareas, establecer fechas límite, priorizar tareas y agregar subtareas.
- Notas y documentos: Permite crear y editar notas, documentos, tablas, bases de datos y más.

- Proyectos: Permite crear y organizar proyectos, asignar tareas a miembros del equipo y colaborar en tiempo real.
- Personalización: Permite personalizar la interfaz, crear plantillas y flujos de trabajo personalizados.

Fortalezas:

- Gran flexibilidad y personalización: Permite adaptar la aplicación a las necesidades específicas de cada usuario.
- Amplia gama de funcionalidades: Gestiona tareas, notas, proyectos y más.
- Ideal para usuarios con necesidades complejas: Ofrece herramientas avanzadas para la organización y la colaboración.
- Interfaz intuitiva y fácil de usar.

Debilidades:

- Curva de aprendizaje para principiantes: Puede ser un poco compleja de usar al principio.
- Versión gratuita con algunas limitaciones: Almacenamiento limitado y algunas funcionalidades solo disponibles en la versión premium.

2.6.6 Tempus Victor

Tempus Victor es un sistema web innovador, diseñado específicamente para satisfacer las necesidades de los estudiantes y ayudarlos a combatir la desorganización académica; ofrece una combinación de metodologías probadas y herramientas personalizadas para optimizar la gestión del tiempo y aumentar la productividad. Las funcionalidades principales incluyen:

- Metodologías probadas: Utiliza la **matriz Eisenhower** para clasificar tareas según su importancia y urgencia, y la **técnica Pomodoro** para dividir el trabajo en sesiones de 25 minutos cada una, con descansos breves.
- Seguimiento en tiempo real: Monitorea el progreso de las tareas en tiempo real, asegurando de no perder ningún detalle.
- Módulos personalizados: Permite ajustar los intervalos de trabajo y descanso, las opciones del temporizador y la interfaz para adaptarse a necesidades específicas.
- Métricas de productividad: Proporciona información sobre el rendimiento, ayudando a identificar áreas de mejora y optimizar el enfoque en las tareas.

- Interfaz intuitiva: Un diseño simple y fácil de usar que facilita la navegación y el uso del sistema, incluso para estudiantes con poca experiencia en tecnología.

Fortalezas:

- Enfoque en estudiantes: Diseñado específicamente para las necesidades académicas, lo que lo hace más relevante y efectivo para este grupo.
- Personalización: Los módulos personalizados permiten a los usuarios ajustar el sistema a sus necesidades específicas, mejorando la eficacia del uso.

Debilidades:

- Público objetivo limitado: Diseñado principalmente para estudiantes, puede no ser tan útil para profesionales o usuarios con necesidades diferentes.
- Curva de aprendizaje: A pesar de su interfaz intuitiva, la adopción de nuevas metodologías y personalizaciones puede requerir tiempo y adaptación.
- Dependencia de la conexión a internet: Como es un sistema web, requiere una conexión constante a internet para su uso óptimo, lo que puede ser una limitación en algunas situaciones.

El siguiente capítulo, el Marco Teórico, proporciona una base conceptual y teórica esencial para comprender los elementos clave involucrados en el diseño y desarrollo del sistema web de organización personal propuesto.

3. MARCO TEÓRICO

Ahora que ya se tiene fundamentado el Trabajo Terminal este capítulo proporciona una introducción a los conceptos básicos necesarios para entender en primer lugar, qué es una tarea y en segundo lugar, lo que es la organización personal; de la misma forma se presentan varios conceptos importantes necesarios para entender de mejor forma todo el contexto en el que se desarrolló Tempus Victor, desde la diferencia entre tarea y otras actividades, hasta conceptos técnicos que ayudaran en la comprensión de los siguientes capítulos.

Finalmente, se profundiza en la metodología de desarrollo que se aplicó, el Modelo en Espiral, explicando sus fases y ventajas para abordar los riesgos de manera temprana y adaptarse a los cambios en los requisitos.

3.1. Tareas

En esta sección, se presenta una definición completa y precisa de una tarea, incluyendo sus características esenciales, diferenciándola de otros conceptos relacionados y ofreciendo una clasificación útil.

3.1.1 Definición

La palabra tarea, tiene un origen árabe, proviene de la palabra "tariha", que se usaba para referirse al "trabajo que había que realizar en poco tiempo"; al mismo tiempo se deriva de "trh", que es el sinónimo de "asignar" (Pérez Porto, 2022).

Desde la teoría de sistemas, una tarea se considera como una acción específica que debe realizarse y que puede completarse en un periodo de tiempo definido (García, 2014). Esta perspectiva se alinea con el enfoque sistémico que analiza los elementos que componen un sistema y sus relaciones, incluyendo las entradas y salidas de información; respecto al mundo exterior una tarea es una parte integrante de un sistema más amplio y su realización contribuye al funcionamiento y la dinámica del sistema en su conjunto.

Para fin de este Trabajo terminal, una tarea es una unidad de trabajo independiente, acotada y con un propósito específico.

3.1.2 Características

Las tareas tienen las siguientes características (Prabhat, 2016):

- Objetivo SMART: Una tarea debe tener un objetivo claro, definido, específico, medible, alcanzable, relevante y temporal. Para Tempus Victor, el objetivo será el contexto del título que el usuario le coloque a cada tarea.
- Inicio y fin definidos: Una tarea debe tener un punto de partida y un punto de finalización identificables y medibles. Para Tempus Victor, esto se ve reflejado en la fecha límite para alguna tarea.
- Responsable asignado: Una tarea debe tener a una persona o equipo claramente definido con la responsabilidad de completarla. Para Tempus Victor, cada usuario que cree una tarea será el único responsable de completarla.
- Recursos necesarios: Una tarea debe tener los recursos necesarios para su correcta ejecución. Los recursos pueden incluir tiempo, dinero, materiales o personas. Para Tempus Victor, los recursos se relacionarán con el tiempo de las tareas, y sus eventos a lo largo de su vida útil.

3.1.3 Diferenciación de conceptos

Es importante distinguir una tarea de otros términos:

- Una actividad: No tiene propósito específico.
- Un objetivo: Es el resultado que se quiere alcanzar, mientras que una tarea es la actividad que se realiza para alcanzar el objetivo.
- Una meta: Es un objetivo a largo plazo, mientras que una tarea es un objetivo a corto plazo.
- Un proyecto: Es un conjunto de tareas que se realizan para alcanzar un objetivo complejo.

Ejemplos de lo que no es una tarea:

- Aprender a tocar el piano: Esto es un objetivo.
- Ser un buen parente: Esto es una meta.
- Construir una casa: Esto es un proyecto.

3.1.4 Tipos

Las tareas se pueden clasificar de diferentes maneras, según su objetivo, su complejidad, su duración o su ámbito (Valdez, 2023).

Estas clasificaciones sirven para agrupar ciertas tareas con fines o contenidos similares, y están completamente relacionadas y personalizadas por cada usuario del sistema Tempus Victor.

Las tareas se pueden clasificar **según su objetivo** en:

- **Tareas de aprendizaje:** Estas tareas están diseñadas para ayudar a las personas a aprender algo nuevo. Por ejemplo, una tarea de aprendizaje puede ser leer un libro, ver un video o completar un ejercicio.
- **Tareas de trabajo:** Estas tareas están diseñadas para ayudar a las personas a realizar su trabajo. Por ejemplo, una tarea de trabajo puede ser escribir un informe, preparar una presentación o realizar una llamada telefónica.
- **Tareas domésticas:** Estas tareas están diseñadas para mantener el hogar limpio y ordenado. Por ejemplo, una tarea doméstica puede ser lavar los platos, limpiar el baño o sacar la basura.

Las tareas se pueden clasificar **según su complejidad** en:

- **Tareas simples:** Estas tareas son relativamente fáciles de completar. Por ejemplo, una tarea simple puede ser escribir una oración, sumar dos números u ordenar una lista.
- **Tareas complejas:** Estas tareas son más difíciles de completar. Por ejemplo, una tarea compleja puede ser escribir un ensayo, resolver un problema matemático o diseñar un plan de marketing.

Las tareas se pueden clasificar **según su duración** en:

- **Tareas cortas:** Estas tareas se pueden completar en un período de tiempo relativamente corto. Por ejemplo, una tarea corta puede ser leer una página, resolver un problema matemático o preparar una comida sencilla.
- **Tareas largas:** Estas tareas requieren más tiempo para ser completadas. Por ejemplo, una tarea larga puede ser leer un libro, resolver un problema complejo o preparar una comida elaborada.

Las tareas se pueden clasificar **según su ámbito** en:

- **Tareas individuales:** Estas tareas son completadas por una sola persona. Por ejemplo, una tarea individual puede ser leer un libro, escribir un ensayo o preparar una comida.
- **Tareas grupales:** Estas tareas son completadas por un grupo de personas. Por ejemplo, una tarea grupal puede ser planificar una fiesta, organizar un evento o realizar un proyecto de investigación.

3.1.5 Ciclo de Vida de una Tarea

- Una Tarea puede ser creada por un usuario agregándolas siguientes características: título, descripción, comentarios, fecha límite, hora límite, categorías, importancia y urgencia.
- Una Tarea puede ser visualizada en cualquier momento desde cualquier vista del sistema.
- Una tarea puede ser editada en cualquier momento por el usuario con posibilidad de actualizar cada una de las características antes mencionadas
- Una tarea puede ser completada en cualquier momento que el usuario decida, ya sea dentro de la vista de matriz Eisenhower o dentro de la vista de calendario, dándole clic al checkbox de la tarea seleccionada, esto quitara la tarea de las tareas no completadas y la colocara en una lista de tareas completadas.
- Una tarea puede cambiar su estado a no completada nuevamente si es que el usuario así lo decide, dentro de la lista de taras no completadas.
- Una tarea puede cambiar su importancia si el usuario la selecciona y arrastra a un nuevo sector de la matriz Eisenhower.
- Una tarea puede cambiar su urgencia si el usuario la selecciona y arrastra a un nuevo sector de la matriz Eisenhower.
- Una tarea puede cambiar su fecha limite si el usuario la selecciona y arrastra a un nuevo día del calendario.
- Una tarea puede ser eliminada en cualquier momento por el usuario, confirmando que en efecto desea eliminar la tarea seleccionada anteriormente.

3.2. Organización personal

En esta sección, se presenta una definición más completa de la organización personal, abarcando sus diversos aspectos y su impacto en la vida de las personas.

3.2.1 Definición

La organización personal se define como un proceso dinámico y continuo que integra la planificación, organización, gestión y optimización del tiempo, espacio, recursos y energía de una persona (Capitalismo consciente, 2022).

La organización personal no solo se limita a la gestión de tareas, sino que impacta en diferentes áreas de la vida, como lo son:

- Productividad: Optimización del tiempo y recursos para alcanzar objetivos con mayor eficiencia.
- Equilibrio: Armonización entre las diferentes áreas de la vida (personal, profesional, social, etc.).
- Bienestar: Reducción del estrés, mejora del estado de ánimo y aumento de la satisfacción personal.
- Desarrollo personal: Crecimiento personal y profesional a través de la gestión efectiva de las habilidades y capacidades.

3.2.2 Características

La organización personal tiene como base los siguientes conceptos:

- Objetivos: Los objetivos son los resultados que una persona quiere alcanzar. La organización personal puede ayudar a las personas a definir sus objetivos y a desarrollar un plan para alcanzarlos.
- Planificación: La planificación se define como el proceso de establecer metas, objetivos y de desarrollar un plan para alcanzarlos (Salvador, 2014).
- Priorización: La priorización es el proceso de determinar la importancia relativa de las tareas o actividades. (Salvador, 2014).
- Gestión del tiempo: La gestión del tiempo es la habilidad de utilizar el tiempo de forma eficaz (Colaborador del sitio DocuSign, 2022).

En la siguiente sección se explican las distintas herramientas que se eligieron para este Trabajo Terminal que ofrecen un apoyo para la organización personal.

3.3. Herramientas para la organización personal

Existen muchas herramientas disponibles para ayudar a las personas a organizarse (Sitio Indeed, 2023). De la gran variedad de herramientas se seleccionaron las siguientes porque se considera que ayudan a la organización personal:

- Calendarios: Los calendarios pueden ayudar a los estudiantes a rastrear sus tareas a lo largo de los días, semanas y meses.
- Listas de tareas: Las listas de tareas pueden ayudar a los estudiantes a priorizar sus tareas y a mantenerse organizadas.

- Sistemas de archivo: Los sistemas de archivo pueden ayudar a los estudiantes a mantener sus notas y tareas de forma organizada.
- Software de organización personal: El software de organización personal puede ayudar a las personas a automatizar algunas tareas de organización personal.

En la siguiente sección se explican las técnicas de organización personal que se eligieron para el desarrollo de este Trabajo Terminal.

3.4. Técnicas de organización personal

Existen varias técnicas de organización personal disponibles para implementar dentro de un sistema web para mejorar la organización personal. A continuación, se presenta un análisis más detallado de las dos técnicas seleccionadas, explorando sus beneficios, limitaciones y aplicaciones prácticas:

3.4.1 Técnica Pomodoro:

La técnica Pomodoro divide el trabajo en intervalos de 25 minutos, conocidos como "Pomodoros", separados por descansos de 5 minutos. Tras completar cuatro Pomodoros, se realiza un descanso más largo de 20-30 minutos (Todoist, 2023).

Beneficios:

- Mejora la concentración y el enfoque.
- Reduce la procrastinación y la fatiga mental.
- Aumenta la productividad y la eficiencia.
- Favorece la gestión del tiempo y la organización.

Limitaciones:

- Requiere disciplina y autocontrol para mantener la estructura de trabajo.
- No es adecuada para tareas largas o complejas que requieren un flujo continuo.
- Puede ser difícil de implementar en entornos de trabajo con interrupciones frecuentes.

Aplicaciones prácticas:

- Estudio y aprendizaje.
- Redacción y escritura.
- Trabajo administrativo y de oficina.
- Tareas creativas y proyectos personales.

3.4.2 Matriz de Eisenhower:

La matriz de Eisenhower clasifica las tareas en cuatro categorías según su importancia y urgencia:

- Importante y urgente: Tareas que deben realizarse de inmediato.
- Importante y no urgente: Tareas que pueden planificarse para un momento posterior.
- No importante y urgente: Tareas que pueden delegarse o eliminarse.
- No importante y no urgente: Tareas que pueden eliminarse o posponerse indefinidamente.

La matriz de Eisenhower puede ayudar a las personas a priorizar sus tareas y a determinar cuáles deben abordarse de inmediato (Asana, 2022).

Beneficios:

- Permite priorizar las tareas de forma eficaz.
- Ayuda a tomar decisiones sobre cómo invertir el tiempo y la energía.
- Reduce el estrés y la sensación de agobio.
- Favorece la concentración en las tareas más relevantes.

Limitaciones:

- Requiere una evaluación precisa de la importancia y urgencia de cada tarea.
- Puede ser difícil clasificar algunas tareas en una sola categoría.
- No ofrece una guía para la ejecución de las tareas.

Aplicaciones prácticas:

- Planificación diaria y semanal.
- Gestión de proyectos y tareas complejas.
- Toma de decisiones estratégicas.
- Priorización de objetivos personales y profesionales.

En la siguiente sección se explica la metodología seleccionada para el desarrollo del sistema Tempus Victor.

3.5. Metodología

En este apartado se presenta la metodología propuesta para el desarrollo del sistema web de gestión personal, utilizando el modelo en espiral de Boehm como base fundamental. Se inicia con una descripción general del modelo, seguido de sus fases y las ventajas que aporta al proyecto. Finalmente, se detalla cómo se aplicará esta metodología al desarrollo del sistema en cuestión.

3.5.1 Introducción al modelo en espiral

El modelo en espiral es una metodología de desarrollo de software que combina elementos de los modelos iterativos y del modelo en cascada; este tipo de modelo se divide en una serie de fases que se repiten de forma iterativa. En cada fase, se realizan actividades de planificación, análisis, diseño, implementación, prueba y evaluación en cada ciclo (Modelo Espiral Deloitte, 2024).

Las fases del modelo en espiral se representan gráficamente como una espiral, donde cada vuelta representa una iteración del proceso. Esta estructura permite un enfoque cíclico y progresivo en el desarrollo, adaptándose a los cambios y refinando el producto de forma incremental.

3.5.2 Fases del modelo en espiral

El modelo en espiral de Boehm, adaptado al proyecto de sistema web de gestión personal, se compone de cuatro fases principales que se repiten en cada iteración del desarrollo:

1. Determinar objetivos: (*Objetivos*)

- Definición de objetivos específicos: Se establecen los objetivos concretos para la iteración actual, considerando las funcionalidades, mejoras o nuevos módulos que se implementarán.
- Comunicación de objetivos: Se comunican los objetivos definidos a todas las partes interesadas, asegurando la alineación y el compromiso con el desarrollo de la iteración.

2. Identificar y resolver riesgos: (*Riesgos*)

- Análisis de riesgos: Se realiza un análisis exhaustivo para identificar y evaluar los riesgos potenciales que podrían afectar la iteración o el proyecto en general.
- Priorización de riesgos: Se priorizan los riesgos identificados en función de su probabilidad de ocurrencia y su impacto potencial en el proyecto.

- Desarrollo de planes de contingencia: Se elaboran planes de contingencia para cada riesgo priorizado, definiendo acciones preventivas y correctivas para mitigar su impacto.

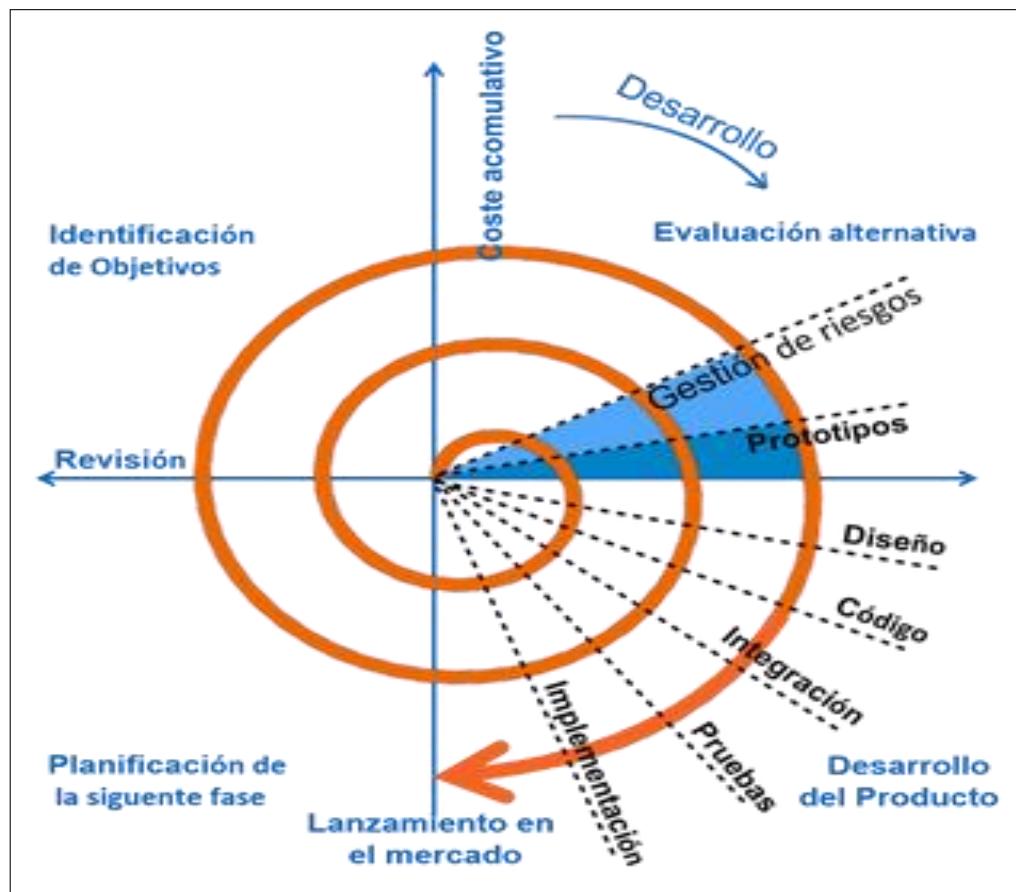
3. Desarrollar y probar: (Desarrollo)

- Codificación e implementación: Se desarrolla el código fuente de las funcionalidades, mejoras o nuevos módulos definidos en la fase de objetivos, utilizando las tecnologías y herramientas seleccionadas.
- Pruebas unitarias: Se realizan pruebas unitarias de cada módulo desarrollado para verificar su correcto funcionamiento de forma individual.
- Integración y pruebas de sistema: Se integran los módulos desarrollados en un sistema funcional completo y se realizan pruebas de sistema para verificar que el software cumple con los requisitos y objetivos de la iteración.

4. Planificar la siguiente iteración: (Planificación)

- Revisión del progreso: Se evalúa el avance de la iteración actual, identificando los logros alcanzados, los desafíos enfrentados y las lecciones aprendidas.
- Ajuste de objetivos y alcance: Se revisan y ajustan los objetivos y el alcance de las siguientes iteraciones en función del progreso actual, los comentarios recibidos y los cambios en los requisitos.
- Planificación de la siguiente iteración: Se crea un plan detallado para la siguiente iteración, definiendo las actividades, cronograma, responsables, recursos y presupuesto necesarios.
- Comunicación del plan: Se comunica el plan de la siguiente iteración a todas las partes interesadas, asegurando la alineación y el compromiso con el desarrollo del proyecto.

En cada iteración, se repiten estas cuatro fases, permitiendo un desarrollo incremental y controlado del sistema web de gestión personal, adaptándose a los cambios, refinando el producto y abordando los riesgos de forma temprana. En la imagen 1 se puede apreciar de manera completa una iteración completa de este modelo en espiral.

Imagen 1: Modelo en espiral

Nota: Elaboración propia

3.5.3 Ventajas del modelo en espiral

El modelo en espiral de Boehm ofrece diversas ventajas significativas para el desarrollo del sistema web de gestión personal, las cuales se detallan a continuación:

1. Abordaje temprano de riesgos:

El modelo en espiral permite identificar y evaluar riesgos potenciales de forma temprana en cada iteración del desarrollo. Esto facilita la implementación de medidas preventivas y correctivas oportunas, reduciendo significativamente la probabilidad de que estos riesgos se materialicen y afecten negativamente al proyecto.

Al abordar los riesgos de forma temprana, se minimizan los costos adicionales y el tiempo de desarrollo que podrían generarse si se dejan sin atención hasta etapas más avanzadas del proyecto.

2. Flexibilidad y adaptabilidad:

El modelo en espiral se caracteriza por su flexibilidad, permitiendo adaptarse a los cambios en los requisitos o en las condiciones del proyecto sin comprometer la calidad o el progreso del desarrollo.

En cada iteración, se realiza una revisión y ajuste de los objetivos, el alcance y el plan de trabajo en función de las nuevas necesidades o circunstancias que puedan surgir. Esta flexibilidad permite al equipo de desarrollo responder de manera ágil y efectiva a los cambios, asegurando que el sistema final se ajuste a las expectativas de los usuarios.

3. Producción de un sistema de alta calidad:

El modelo en espiral pone un fuerte énfasis en la validación de los requisitos y en las pruebas exhaustivas del sistema, lo que conduce a la entrega de un producto final de alta calidad.

La constante evaluación y refinamiento de los requisitos a lo largo del desarrollo garantizan que el sistema cumpla con las necesidades y expectativas de los usuarios. Las pruebas unitarias, de integración y de sistema realizadas en cada iteración permiten identificar y corregir errores tempranamente, previniendo defectos y asegurando el correcto funcionamiento del software.

En resumen, el modelo en espiral ofrece un enfoque estructurado y adaptable para el desarrollo del sistema web de gestión personal, proporcionando las siguientes ventajas:

- Reducción de costos y tiempo de desarrollo al abordar los riesgos de forma temprana.
- Mayor flexibilidad para adaptarse a cambios en los requisitos o condiciones del proyecto.
- Entrega de un sistema de alta calidad que cumple con las expectativas de los usuarios.

Al adoptar el modelo en espiral, se puede garantizar un proceso de desarrollo eficiente, controlado y orientado a la calidad, lo que aumenta las posibilidades de éxito del proyecto en tiempo y forma, cumpliendo los objetivos planteados del trabajo terminal.

3.5.4 Aplicación al desarrollo del sistema

El modelo en espiral se presenta como una metodología idónea para el desarrollo del sistema de organización personal "Tempus Victor", debido a su enfoque iterativo y controlado que permite abordar la complejidad del proyecto de manera organizada y eficiente. A continuación, se describe la aplicación de cada fase del modelo en espiral al contexto específico del sistema "Tempus Victor":

Fase 1: Determinar objetivos: La fase inicial del modelo en espiral para el desarrollo del sistema "Tempus Victor" se centra en definir claramente los objetivos y el alcance del proyecto. Esta etapa fundamental establece la base para el éxito del proyecto, asegurando que todos los involucrados comprendan las expectativas y trabajen hacia un objetivo común.

Objetivo principal:

- Establecer objetivos claros y medibles para el sistema "Tempus Victor", enfocándose en el impacto positivo en la organización personal y el éxito académico de los estudiantes.

Actividades clave:

- Definición de objetivos generales: Redactar objetivos generales que describan el propósito principal del sistema y su contribución al bienestar de los estudiantes.
- Establecimiento de objetivos específicos: Desglosar los objetivos generales en objetivos específicos, medibles, alcanzables, relevantes y con un plazo determinado (SMART).
- Identificación de requisitos funcionales y no funcionales: Detallar las características y funcionalidades que el sistema debe ofrecer, así como los aspectos no funcionales como la usabilidad, seguridad, rendimiento y escalabilidad.
- Validación de objetivos y requisitos: Revisar y validar los objetivos y requisitos con las partes interesadas, incluyendo a los estudiantes, profesores, administradores y expertos en educación.
- Documentación de objetivos y requisitos: Formalizar los objetivos y requisitos en un documento claro y accesible para todos los miembros del equipo de desarrollo.

Fase 2: Identificar y resolver riesgos: En esta fase crucial, se realiza un análisis exhaustivo para identificar, evaluar y mitigar los riesgos potenciales que podrían afectar el desarrollo del sistema "Tempus Victor". El objetivo es prevenir o minimizar el impacto negativo de estos riesgos, asegurando un desarrollo controlado y exitoso del proyecto.

Objetivo principal:

- Identificar, evaluar y mitigar los riesgos potenciales que podrían afectar el desarrollo del sistema "Tempus Victor", garantizando un proceso controlado y exitoso.

Actividades clave:

- Identificación de riesgos: Realizar un análisis exhaustivo del proyecto para identificar todos los posibles riesgos, considerando tanto factores internos como externos.
- Evaluación de riesgos: Evaluar la probabilidad de ocurrencia y el impacto potencial de cada riesgo identificado.
- Priorización de riesgos: Clasificar los riesgos en función de su probabilidad e impacto para enfocar los esfuerzos en los más críticos.
- Desarrollo de planes de mitigación: Elaborar planes específicos para cada riesgo priorizado, definiendo acciones preventivas y correctivas para reducir su probabilidad de ocurrencia o mitigar su impacto.
- Monitoreo y seguimiento de riesgos: Realizar un seguimiento continuo de los riesgos identificados y actualizar los planes de mitigación en función de la evolución del proyecto.

Fase 3: Desarrollar y probar: En esta fase central, se lleva a cabo el desarrollo e implementación de las funcionalidades del sistema "Tempus Victor". Se realizan pruebas unitarias, de integración y de sistema para garantizar el correcto funcionamiento del sistema y su cumplimiento con los requisitos establecidos.

Objetivo principal:

- Desarrollar e implementar las funcionalidades del sistema "Tempus Victor" de manera eficiente y segura, asegurando su correcto funcionamiento y cumplimiento con los requisitos.

Actividades clave:

- Desarrollo: Implementar el código fuente del sistema utilizando las tecnologías seleccionadas: Angular para el Frontend, Express para el Backend y MySQL para la base de datos.
- Pruebas de sistema: Desarrollar pruebas de sistema para garantizar que el sistema cumple con todos los requisitos y funciona correctamente.

Fase 4: Planificar la siguiente iteración: En la fase final del modelo en espiral, se define el plan para la siguiente iteración, ajustando objetivos, alcance y actividades en función del progreso, retroalimentación y lecciones aprendidas.

Objetivo principal:

- Establecer un plan detallado para la siguiente iteración del desarrollo.

Actividades clave:

- Revisar el progreso: Analizar logros, desafíos y lecciones aprendidas.
- Evaluar el impacto de los riesgos: Revisar la efectividad de las medidas de mitigación e identificar nuevos riesgos.
- Recopilar retroalimentación: Obtener comentarios de sinodales, director y usuarios de prueba.
- Ajustar objetivos y alcance: Revisar objetivos generales y definir el alcance de la siguiente iteración.
- Priorizar funcionalidades: Priorizar las funcionalidades pendientes para la siguiente iteración.
- Planificar en detalle: Crear un cronograma, estimar el presupuesto y asignar recursos.
- Actualizar el plan general: Incorporar cambios en el plan general del proyecto.

Al aplicar el modelo en espiral al desarrollo del sistema "Tempus Victor", se garantiza un proceso controlado, flexible y adaptable que permite abordar los desafíos de manera efectiva, minimizar los riesgos potenciales y entregar un producto final que cumpla con los objetivos establecidos y satisfaga las necesidades de los usuarios.

Este enfoque iterativo facilita la gestión del proyecto, la optimización del desarrollo y la entrega de un sistema de organización personal robusto, escalable y seguro que empodera a los estudiantes para alcanzar sus metas académicas.

La implementación exitosa del modelo en espiral permitirá el desarrollo de un sistema "Tempus Victor" de alta calidad que tendrá un impacto positivo en la vida de los estudiantes, en la mejora de su productividad y de su rendimiento académico.

En la siguiente sección se explicarán los conceptos técnicos necesarios para entender el desarrollo del sistema Tempus Victor.

3.6. Conceptos técnicos

A continuación, se presentan los conceptos técnicos necesarios para entender el desarrollo del sistema web de organización personal Tempus Victor.

3.6.1 Modelo de tres capas

El modelo de tres capas es una arquitectura de software que separa las responsabilidades en tres niveles distintos: presentación, negocio y datos. (Arquitectura tres niveles, IBM, 2024)

- Capa de presentación: Es la interfaz con la que interactúa el usuario. Se encarga de mostrar los datos y captar las acciones del usuario. En el caso de Tempus Victor, esta capa incluye las páginas web y los componentes de la interfaz de usuario.
 - Función: Interactúa con el usuario.
 - Componentes: Interfaces de usuario (UI), páginas web.
 - Características: Diseño visual, usabilidad, interactividad.
 - Ejemplos: HTML, CSS, JavaScript, frameworks.
- Capa de negocio: Contiene la lógica de la aplicación, las reglas de negocio y el procesamiento de datos. En Tempus Victor, esta capa maneja las operaciones de organización personal como la gestión de tareas, gestión de notas y de sesiones Pomodoro.
 - Función: Maneja la lógica de la aplicación y las reglas de negocio.
 - Componentes: Servicios, controladores, lógica de procesamiento.
 - Características: Abstracción de la lógica de negocio, procesamiento de datos, validación de reglas de negocio.
 - Ejemplos: Archivos de lógica en Node.js, servicios RESTful.
- Capa de datos: Gestiona el acceso a la base de datos. Se encarga de la persistencia y recuperación de los datos. En el sistema Tempus Victor, esta capa se comunica con la base de datos para almacenar y recuperar información sobre las actividades del usuario.
 - Función: Gestiona la persistencia de datos.
 - Componentes: Bases de datos, ORM, consultas SQL.
 - Características: Almacenamiento y recuperación de datos, integridad de datos, rendimiento.
 - Ejemplos: MySQL, Prisma (ORM).

3.6.2 API (Interfaz de Programación de Aplicaciones)

Una API, o Interfaz de Programación de Aplicaciones, es un conjunto de definiciones y protocolos que permite a las aplicaciones comunicarse entre sí (API, Amazon, 2024). Las APIs definen la manera en que los componentes de software deben interactuar y permiten que los desarrolladores usen funciones ya existentes sin tener que escribir el código desde cero.

Puntos clave:

- Interoperabilidad: Facilita la comunicación entre diferentes sistemas y plataformas.
- Abstracción: Oculta la complejidad del sistema, proporcionando interfaces sencillas para interactuar con él.
- Reusabilidad: Permite reutilizar funciones y servicios en múltiples aplicaciones.

Características:

- Estándares abiertos: Muchas APIs siguen estándares abiertos que aseguran su compatibilidad y facilidad de uso (por ejemplo, HTTP para APIs web).
- Documentación: Las APIs bien diseñadas están acompañadas de documentación detallada que explica cómo usarla, los endpoints disponibles, los métodos permitidos, y los formatos de datos esperados.
- Seguridad: Implementan mecanismos de seguridad como autenticación y autorización (por ejemplo, a través de tokens JWT).

3.6.3 Servicio RESTful

Los servicios RESTful son un tipo de API que siguen los principios de la arquitectura REST (Representational State Transfer). REST es una arquitectura de estilo que utiliza los estándares y protocolos de la web, como HTTP, para permitir la comunicación entre sistemas (API REST, Red Hat, 2023).

Puntos clave:

- Stateless: Cada solicitud del cliente al servidor debe contener toda la información necesaria para entender y procesar la solicitud. No se mantiene estado en el servidor entre solicitudes.
- Recursos y URIs: Los datos y funcionalidades se exponen como recursos que se identifican mediante URIs (Uniform Resource Identifiers).
- Operaciones CRUD: Utiliza los métodos HTTP estándar para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre los recursos.
 - GET: Recuperar información de un recurso.
 - POST: Crear un nuevo recurso.
 - PUT: Actualizar un recurso existente.
 - DELETE: Eliminar un recurso.

Características:

- Uniformidad de Interfaz: Define una interfaz uniforme para interactuar con los recursos, lo que mejora la simplicidad y la visibilidad de las interacciones.
- Escalabilidad: Los servicios RESTful pueden escalar horizontalmente, manejando grandes cantidades de tráfico y datos.
- Cacheabilidad: Las respuestas de las solicitudes pueden ser cacheadas para mejorar el rendimiento.
- Separación cliente-servidor: El cliente y el servidor están completamente separados. Esto permite que ambos evolucionen de manera independiente.
- Representaciones de recursos: Los recursos pueden tener múltiples representaciones (por ejemplo, JSON, XML) que permiten su fácil manipulación y visualización.

3.6.4 Frontend

El Frontend es la parte del desarrollo web que se encarga de la interfaz de usuario y la experiencia del usuario (UX). Es lo que el usuario ve e interactúa en un navegador web (Frontend y Backend, AWS, 2024). En Tempus Victor, el Frontend incluye todos los elementos visuales y funcionales con los que el usuario interactúa como formularios, botones, menús y gráficos.

Puntos clave:

- Desarrollo UI/UX: Diseño y desarrollo de la interfaz y experiencia de usuario.
- Tecnologías: HTML, CSS, JavaScript, frameworks como Angular.

Características:

- Interactividad: Responde a las acciones del usuario.
- Diseño Responsivo: Adaptación a diferentes tamaños de pantalla y dispositivos.
- Accesibilidad: Garantiza que la aplicación sea utilizable por personas con diferentes capacidades.

3.6.5 Backend

El Backend es la parte del desarrollo web que se encarga de la lógica del servidor, las bases de datos y la integración con otras aplicaciones y servicios (Frontend y Backend, AWS, 2024). En Tempus Victor, el Backend maneja las solicitudes del Frontend, procesa la lógica del negocio y gestiona las operaciones de base de datos.

Puntos clave:

- Gestión de datos: Manipulación y almacenamiento de datos.
- Seguridad: Autenticación, autorización, y protección de datos.

- Integración: Comunicación con otros servicios y APIs.

Características:

- Rendimiento: Optimización para manejar múltiples solicitudes simultáneamente.
- Escalabilidad: Capacidad de crecer y manejar más carga.
- Fiabilidad: Garantiza la disponibilidad y consistencia de los servicios.

3.6.6 Base de datos

Una base de datos es un sistema que permite almacenar, organizar y gestionar grandes cantidades de información de manera eficiente (DB, Oracle, 2024). En Tempus Victor, la base de datos almacena toda la información relacionada con las tareas, eventos, usuarios y otros datos relevantes.

Puntos clave:

- Modelo de datos: Relacional (tablas) o NoSQL (documentos, gráficos, etc.).
- Consultas: Lenguajes de consulta como SQL.

Características:

- Persistencia: Almacenamiento duradero de datos.
- Consistencia: Garantiza que los datos sean correctos y coherentes.
- Integridad: Mantenimiento de la exactitud y validez de los datos.

3.6.7 MySQL

MySQL es un sistema de gestión de bases de datos relacional de código abierto muy popular. Utiliza SQL (Structured Query Language) para gestionar y manipular los datos almacenados (MySQL, 2023). En Tempus Victor, MySQL podría ser utilizado para manejar todas las operaciones de almacenamiento y recuperación de datos.

Puntos clave:

- SQL: Utiliza el lenguaje SQL para gestionar los datos.
- Relacional: Estructura los datos en tablas con relaciones definidas.

Características:

- Rendimiento: Optimizado para velocidad y eficiencia.
- Escalabilidad: Capacidad de manejar grandes volúmenes de datos.
- Seguridad: Soporte para autenticación y autorización.

3.6.8 ORM

ORM (Object-Relational Mapping) es una técnica de programación que permite convertir datos entre sistemas de tipo de objetos y sistemas de bases de datos relacionales (ORM, Deloitte, 2024). Facilita la manipulación de datos en la base de datos a través de objetos en el código del programa. En Tempus Victor, un ORM como Prisma ayuda a interactuar con la base de datos MySQL de una manera más intuitiva y menos propensa a errores.

Puntos clave:

- Mapeo: Transforma datos entre bases de datos y objetos de programación.
- Abstracción: Simplifica las consultas a la base de datos.

Características:

- Productividad: Reduce la cantidad de código SQL manual.
- Mantenibilidad: Facilita la actualización y mantenimiento del código.
- Seguridad: Minimiza riesgos de inyección SQL.

3.6.9 Prisma

Prisma es un ORM para Node.js y TypeScript que facilita la interacción con bases de datos. Permite definir el modelo de datos de manera declarativa y genera un cliente de base de datos para realizar consultas seguras y eficientes (Prisma, 2024). En Tempus Victor, Prisma simplifica la manipulación de datos en MySQL.

Puntos clave:

- Schema: Define el modelo de datos de manera declarativa.
- Generación de Cliente: Crea un cliente de base de datos para realizar consultas.

Características:

- Consultas tipadas: Seguridad en las consultas mediante tipos.
- Migraciones: Facilita la gestión de cambios en el esquema de la base de datos.
- Integración: Compatible con múltiples bases de datos como MySQL

3.6.10 Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft (Visual Studio Code, 2024). Es muy popular entre los desarrolladores debido a sus potentes características como depuración, control de versiones, y una gran cantidad de extensiones disponibles. Es una herramienta ideal para el desarrollo de Tempus Victor, ya que soporta múltiples lenguajes de programación y tecnologías.

Puntos clave:

- Multilenguaje: Soporte para múltiples lenguajes de programación.
- Extensiones: Gran variedad de extensiones para aumentar la funcionalidad.

Características:

- Depuración: Herramientas integradas para depurar código.
- Control de Versiones: Integración con sistemas como Git.
- Personalización: Temas, atajos y configuraciones personalizables.

3.6.11 Angular

Angular es un Framework de desarrollo web de código abierto mantenido por Google (Angular, 2023). Se utiliza para crear aplicaciones web de una sola página (SPA) con una experiencia de usuario rica. En Tempus Victor, Angular se encarga del desarrollo del Frontend, proporcionando una estructura sólida y herramientas para construir una interfaz de usuario dinámica y reactiva.

Puntos clave:

- SPA: Diseñado para crear aplicaciones de una sola página.
- TypeScript: Utiliza TypeScript para el desarrollo.

Características:

- Componentes: Basado en una arquitectura de componentes.
- MVVM: Modelo-Vista-Vista-Modelo para una separación clara de responsabilidades.
- Herramientas: CLI, testing, y optimización integradas.

3.6.12 Express

Express es un Framework web para Node.js, diseñado para construir aplicaciones web y APIs de manera rápida y sencilla (Express, 2023). En Tempus Victor, Express se usa en el Backend para manejar las rutas, gestionar las solicitudes y respuestas HTTP, y conectar la lógica de negocio con la base de datos.

Puntos clave:

- Rutas: Definición de rutas para gestionar solicitudes HTTP.
- Middleware: Funcionalidades adicionales mediante funciones intermedias.

Características:

- Minimalista: Esqueleto básico para aplicaciones web.
- Flexibilidad: Gran cantidad de middleware y plugins disponibles.
- Desempeño: Alto rendimiento y eficiencia.

3.6.13 Node.js

Node.js es un entorno de ejecución para JavaScript que permite ejecutar código JavaScript en el servidor (Node.js, 2024). Es ideal para construir aplicaciones escalables y de alto rendimiento. En Tempus Victor, Node.js proporciona el entorno necesario para ejecutar el Backend de la aplicación.

Puntos clave:

- V8 Engine: Basado en el motor V8 de Google para ejecutar JavaScript.
- Asincronía: Utiliza un modelo de E/S no bloqueante.

Características:

- Rendimiento: Alta capacidad de procesamiento de solicitudes concurrentes.
- Comunidad: Gran ecosistema de módulos y paquetes.
- Escalabilidad: Adecuado para aplicaciones escalables en tiempo real.

3.6.14 TypeScript

TypeScript es un lenguaje de programación desarrollado por Microsoft que extiende JavaScript añadiendo tipos estáticos (TypeScript, 2024). Ayuda a escribir código más claro y menos propenso a errores. En Tempus Victor, TypeScript se utiliza tanto en el Frontend (con Angular) como en el Backend (con Node.js) para mejorar la calidad y mantenibilidad del código.

Puntos clave:

- Superset: Extiende JavaScript con características adicionales.
- Tipos: Adición de tipos estáticos y anotaciones.

Características:

- Transpilación: Se transpila a JavaScript para compatibilidad.
- Mantenibilidad: Código más claro y menos propenso a errores.
- Herramientas: Mejores herramientas de desarrollo y depuración.

3.6.15 JWT

JWT (JSON Web Token) es un estándar para crear tokens de acceso que permiten verificar la autenticidad de un usuario en sistemas distribuidos (JWT, 2024). En Tempus Victor, JWT se usa para gestionar la autenticación y autorización de los usuarios, asegurando que solo los usuarios autenticados puedan acceder a ciertos recursos o realizar ciertas acciones.

Puntos clave:

- Compacto: Formato de token ligero.

- Seguro: Firmado digitalmente para asegurar su integridad.

Características:

- Interoperabilidad: Utilizado en diferentes sistemas y plataformas.
- Transporte: Enviado a través de HTTP headers.
- Claims: Incluye información sobre el usuario y permisos.

3.6.16 Node Package Manager (npm)

Node Package Manager es el gestor de paquetes predeterminado para Node.js. Permite instalar, compartir y gestionar dependencias y bibliotecas de código (NPM, 2024). En Tempus Victor, npm se utiliza para gestionar todas las dependencias del proyecto, incluyendo librerías y Frameworks necesarios tanto para el Frontend como para el Backend.

Puntos clave:

- Repositorio: Gran colección de paquetes y librerías.
- Gestión de dependencias: Instalación y administración de dependencias del proyecto.

Características:

- Scripts: Ejecución de scripts definidos en el proyecto.
- Comunidad: Amplio soporte y contribución de la comunidad.
- Versionado: Manejo de versiones y actualizaciones de paquetes.

En este capítulo, se ha proporcionado una introducción a los conceptos básicos de la organización personal y las tareas. Se ha definido una tarea como una actividad que debe ser completada dentro de un período, y se han discutido las características de lo que sí y lo que no se considera como tarea. También se ha definido la organización personal como el proceso de planificar, organizar y gestionar el tiempo, el espacio y los recursos de un estudiante. Asimismo, se explicaron conceptos técnicos importantes para comprender en su mayoría los siguientes capítulos.

Estos conceptos son fundamentales para comprender la estructura y el funcionamiento de un sistema web como Tempus Victor, asegurando una implementación eficiente y robusta.

4. TRABAJO REALIZADO

En este capítulo se presenta la descripción de todas las actividades correspondientes al trabajo realizado durante todo el Trabajo Terminal y el desarrollo del sistema web "Tempus Victor".

Empezando con el Cronograma de actividades Final, el cual tiene la finalidad de estimar tiempos para el desarrollo del Trabajo Terminal. El cronograma se divide en cinco iteraciones que en su conjunto tienen una duración de dos semestres o un año para el Trabajo Terminal. Está dividido por meses y por semanas, tiene cuadros pintados de morado que representan las semanas en las que se trabajó su actividad correspondiente; dentro del cronograma las dos primeras iteraciones corresponden al análisis y diseño del sistema, mientras tanto las ultimas 3 iteraciones corresponden al desarrollo del sistema.

Cronograma de actividades Final:

Iteración 1:

- Establecimiento de objetivos para la iteración.
- Planteamiento de la problemática y su análisis.
- Planteamiento de la propuesta de solución.
- Investigación del estado del arte.
- Identificación de actores del sistema.
- Investigación de desarrollo del marco teórico.
- Identificación de requerimientos funcionales y no funcionales.
- Análisis y diseño del diagrama relacional.
- Identificación y análisis de riesgos.
- Corrección y actualización de la documentación.

Iteración 2:

- Establecimiento de objetivos para la iteración.
- Análisis de factibilidad económica.
- Análisis de factibilidad técnica.
- Diseño de interfaces graficas del sistema.

- Planteamiento de casos de uso.
- Identificación y descripción de reglas de negocio.
- Desarrollo y corrección de interfaces graficas del sistema.
- Pruebas y corrección de errores del sistema.
- Corrección y actualización de la documentación.
- Planificación de la siguiente iteración.

Iteración 3:

- Establecimiento de objetivos para la iteración.
- Identificación y evaluación de riesgos.
- Desarrollo y corrección de interfaces graficas del sistema.
- Desarrollo y corrección de la base de datos del sistema.
- Implementación y corrección de casos de uso.
- Implementación y corrección de reglas de negocio.
- Pruebas y corrección de errores del sistema.
- Redacción y corrección de la documentación.
- Planificación de la siguiente iteración.

Iteración 4:

- Las mismas actividades que la iteración 3.

Iteración 5:

- Las mismas actividades que la iteración 3, más los resultados, conclusiones, trabajo a futuro.

Aplicación del modelo en espiral: Las fases del modelo en espiral se aplicaron de manera flexible a lo largo de las iteraciones, adaptándose a las necesidades específicas de cada etapa:

Fase 1: Revisión y refinamiento de objetivos.

Fase 2: Identificación y evaluación de nuevos riesgos.

Fase 3: Pruebas continuas para garantizar el funcionamiento y la satisfacción del usuario.

Fase 4: Planificación dinámica basada en resultados y nuevas oportunidades.

4.1. Cronograma de Actividades

Imagen 2: Cronograma de actividades TT 2024-A136

I te ra cio	Actividades	2023												2024												M es S em			
		AGO			SEP			OCT			NOV			DIC			ENE			FEB			MAR			ABR			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Establecimiento de objetivos para la iteración																												
	Planteamiento de la problemática y su análisis																												
	Planteamiento de la Propuesta de solución																												
	Investigación del estado del arte																												
	Identificación de Actores del sistema																												
	Investigación y Desarrollo del marco teórico																												
	Identificación de requerimientos funcionales y no funcionales																												
	Análisis y diseño de diagrama relacional																												
	Identificación y Análisis de riesgos																												
	Corrección y actualización de la documentación																												
2	Establecimiento de objetivos para la iteración																												
	Análisis de factibilidad económica																												
	Análisis de factibilidad técnica																												
	Diseño de Interfaces gráficas del sistema																												
	Planteamiento de caso de uso																												
	Identificación y descripción de reglas de negocio																												
	Desarrollo y corrección de Interfaces gráficas del sistema																												
	Pruebas y corrección de errores del sistema																												
	Corrección y actualización de la documentación																												
	Planificación de la siguiente iteración																												
<i>Evaluación de TT1</i>																													
3	Establecimiento de objetivos para la iteración																												
	Identificación y evaluación de riesgos																												
	Desarrollo y corrección de Interfaces gráficas del sistema																												
	Desarrollo y corrección de la base de datos del sistema																												
	Implementación y corrección de casos de uso																												
	Implementación y corrección de reglas de negocio																												
	Pruebas y corrección de errores del sistema																												
	Redacción y corrección de la documentación																												
	Planificación de la siguiente iteración																												
	Establecimiento de objetivos para la iteración																												
4	Identificación y evaluación de riesgos																												
	Desarrollo y corrección de Interfaces gráficas del sistema																												
	Desarrollo y corrección de la base de datos del sistema																												
	Implementación y corrección de casos de uso																												
	Implementación y corrección de reglas de negocio																												
	Pruebas y corrección de errores del sistema																												
	Redacción y corrección de la documentación																												
	Planificación de la siguiente iteración																												
	Establecimiento de objetivos para la iteración																												
	Identificación y evaluación de riesgos																												
5	Desarrollo y corrección de Interfaces gráficas del sistema																												
	Desarrollo y corrección de la base de datos del sistema																												
	Implementación y corrección de casos de uso																												
	Implementación y corrección de reglas de negocio																												
	Pruebas y corrección de errores del sistema																												
	Redacción y corrección de la documentación																												
	Resultados, Conclusiones, Trabajo a futuro, y entrega de documentación.																												
	<i>Evaluación de TT2</i>																												10

Nota: Elaboración propia

En las secciones siguientes se presentan las actividades descritas en el cronograma de manera más extensa, en algunas secciones se hará la referencia al material completo en el capítulo de Anexos, encontrándose de manera complementaria al final del documento. Este material complementario puede ayudar a mejorar la comprensión del trabajo desarrollado, por lo que se recomienda leerlo cuando se requiera.

4.2. Primera iteración

En esta sección se describe detalladamente el trabajo realizado durante la primera iteración del desarrollo del sistema web "Tempus Victor". Esta iteración marca el inicio del proceso de desarrollo, donde se establecieron las bases y se definieron los lineamientos iniciales para el proyecto, siguiendo las fases del modelo en espiral.

La primera iteración incluyó una serie de actividades fundamentales, comenzando con el establecimiento de objetivos claros para esta fase inicial que guiaron las acciones y decisiones posteriores. Se llevó a cabo un análisis exhaustivo de la problemática a resolver, seguido del planteamiento de una propuesta de solución detallada. La investigación del Estado del Arte proporcionó un contexto actualizado sobre las tecnologías y enfoques disponibles, mientras que la identificación de actores del sistema permitió definir claramente los roles y responsabilidades de los futuros usuarios.

Se desarrolló un marco teórico sólido que sustentará el desarrollo del sistema, complementado con la identificación de los requerimientos funcionales y no funcionales necesarios para cumplir con las expectativas de los usuarios. El análisis y diseño del diagrama relacional estableció la estructura de la base de datos, asegurando una organización coherente de la información. Además, se identificaron y analizaron los riesgos potenciales, proponiendo estrategias para su mitigación.

Finalmente, se llevaron a cabo las correcciones y actualizaciones de la documentación pertinente, garantizando que toda la información reflejara con precisión el progreso y las decisiones tomadas durante esta iteración. De esta manera, la iteración 1 sentó las bases para el desarrollo ordenado y sistemático del sistema "Tempus Victor", preparando el camino para las siguientes fases del proyecto.

4.2.1 Establecimiento de objetivos para la iteración actual

En este paso se definieron los objetivos específicos a alcanzar durante esta primera fase, estableciendo un enfoque claro para guiar las actividades y asegurar el avance del desarrollo del sistema.

Los objetivos de la primera iteración se definieron como:

- Analizar y definir la problemática.
- Definir una propuesta de solución.
- Investigar sistemas web en el mercado que se parezcan a la propuesta.
- Identificar los actores del sistema.
- Definir el alcance del sistema.
- Definir la metodología a trabajar.
- Actualizar el cronograma de actividades.
- Investigar los conceptos que pueden ayudar al desarrollo del proyecto.
- Identificar los requerimientos funcionales y no funcionales del sistema.
- Identificar y analizar posibles riesgos a lo largo del desarrollo.
- Redactar con formalidad y de forma correcta el avance en esta iteración.

4.2.2 Planteamiento de la problemática y su análisis

En este paso se identificó y analizó la problemática que el sistema "Tempus Victor" pretende solucionar, considerando el contexto y las necesidades de los usuarios finales.

Después de la investigación sobre las problemáticas específicas se llegó a la problemática general definida anteriormente como:

"La problemática general radica en la falta de habilidades y hábitos de organización personal, lo que dificulta a los estudiantes gestionar de manera efectiva sus tareas, tiempo y responsabilidades en un entorno cada vez más complejo y demandante. Esta situación afecta negativamente su rendimiento académico, su bienestar emocional y su calidad de vida en general. Todo esto porque no es fácil planificar, gestionar, dar seguimiento, priorizar y guardar información sobre múltiples tareas."

4.2.3 Planteamiento de la propuesta de solución

En este paso se planteó una propuesta inicial de solución, detallando cómo el sistema puede abordar la problemática identificada, y se bosquejaron las principales funcionalidades del sistema.

El sistema web para organización personal recibió el nombre de “Tempus Victor”, un nombre que evoca dedicación a realizar las tareas en tiempo y forma. Se definió ese nombre, puesto que llamarle sistema web de organización personal cada vez que se quiere hacer referencia al sistema es un nombre muy largo; entonces, para enfocar y simplificar los esfuerzos se le dio el nombre en latín “Tempus Victor”.

Asimismo, se definieron los objetivos específicos para el trabajo terminal, los cuales se enfocaron a desarrollar módulos de notas, de tareas, de calendario, de matriz Eisenhower, de temporizador Pomodoro, módulo de personalización y finalmente un módulo de métricas de productividad.

Y al final, dado el trabajo de análisis realizado de acuerdo con la investigación sobre problemáticas y las posibles soluciones, se planteó el objetivo general con la siguiente redacción:

“Diseñar y desarrollar un sistema web que proporcione apoyo para la organización personal haciendo uso de herramientas personalizables enfocadas en tiempo y tareas.”

Con todo eso se llegó a la conclusión de que la creación de un sistema web de organización personal para estudiantes está justificada por las múltiples necesidades y desafíos que enfrentan en su vida académica. Este sistema tiene el potencial de mejorar significativamente la gestión del tiempo, la organización de tareas y la priorización de actividades, lo cual puede traducirse en un rendimiento académico superior, reducción del estrés y ansiedad y, una mayor satisfacción personal. Además, la integración de metodologías como la técnica Pomodoro y la matriz Eisenhower proporcionará a los estudiantes herramientas efectivas para optimizar su tiempo y esfuerzos, preparándolos mejor para futuros desafíos tanto académicos como profesionales.

4.2.4 Investigación del Estado del Arte

En este paso se llevó a cabo una investigación sobre el estado del arte, incluyendo una revisión de sistemas similares existentes, tecnologías disponibles y metodologías aplicables.

Dentro del análisis comparativo de diversos sistemas web de organización personal se analizaron los sistemas: Google Calendar, Todoist, Evernote, TickTick y Notion, cada uno con sus propias fortalezas y debilidades.

Las características que se consideraron útiles para hacer el análisis fueron:

- A) Contar con herramientas para la planificación efectiva.
- B) Contar con estrategias para la gestión del tiempo.
- C) Implementar métodos para la organización de la información.
- D) Brindar recursos para el desarrollo de la priorización.
- E) Contar con plataforma web.
- F) Implementar la técnica Pomodoro.
- G) Implementar la matriz Eisenhower.

Estas características llevaron a la creación de una tabla presentada anteriormente que indica las características que cumplen los sistemas y cuáles no, así como otros factores relevantes que influyen en su efectividad y utilidad para los estudiantes.

Esta comparación permite identificar las fortalezas y debilidades de cada sistema, proporcionando una visión clara de las mejores opciones disponibles y de cómo el sistema propuesto puede ofrecer una solución más completa y eficaz.

4.2.5 Identificación de actores del sistema

En este paso se identificaron y definieron los actores que interactuarán con el sistema, estableciendo sus roles y responsabilidades dentro del mismo.

Un actor es una entidad externa que interactúa con el sistema para realizar una función específica. Los actores pueden ser personas, sistemas informáticos o incluso hardware. Los actores representan los requisitos del usuario del sistema. Identificar a los actores, ayuda a comprender de mejor manera las necesidades de los usuarios y así poder diseñar un sistema que los satisfaga.

Los actores son los usuarios del sistema; por lo que se identificaron los siguientes:

Estudiante: Es un rol genérico que representa a cualquier usuario del sistema. Este rol es independiente de la edad, género, ocupación o nivel de educación del usuario.

Responsabilidades: Las responsabilidades del estudiante en el sistema web de organización personal incluyen:

- Gestionar sus tareas: crear, editar, priorizar y marcar como completadas sus tareas.
- Gestionar sus notas: crear, editar y organizar sus notas.
- Hacer seguimiento de sus tareas con la técnica Pomodoro.
- Priorizar sus tareas con la matriz de Eisenhower.
- Organizar sus tareas con el calendario.

Perfil: Para poder utilizar el sistema web, se debe cumplir con los siguientes requisitos:

- Acceso a un dispositivo con conexión a internet: La persona debe tener acceso a un dispositivo, como una computadora, un teléfono inteligente o una tableta, con conexión a internet.
- Conocimientos básicos de informática: La persona debe tener conocimientos básicos de informática, como el navegar por la web, usar un ratón, usar un teclado, abrir y guardar archivos.
- Alfabetización digital: La persona debe tener alfabetización digital, que es la capacidad de comprender y utilizar la información digital.

4.2.6 Investigación y desarrollo del Marco Teórico

En este paso se investigaron y documentaron los conceptos teóricos relevantes que sustentan el desarrollo del sistema, proporcionando una base sólida para las decisiones técnicas y de diseño.

Los conceptos que se presentan en el capítulo del Marco Teórico proporcionan una introducción a los conceptos básicos necesarios para entender en primer lugar qué es una tarea y en segundo lugar qué es la organización personal; de la misma forma se presentan varios conceptos importantes necesarios para entender mejor el contexto en el que se desarrolló Tempus Victor, desde la diferencia entre tarea y otras actividades, hasta conceptos técnicos que ayudan en la comprensión de los siguientes capítulos.

Finalmente, se profundiza en la metodología de desarrollo que se aplicó, el modelo en Espiral, explicando sus fases y ventajas para abordar los riesgos de manera temprana y adaptarse a los cambios en los requisitos.

Los conceptos técnicos importantes que se definieron fueron: Modelo de tres capas, API, Servicio RESTful, Frontend, Backend, Base de datos, MySQL, ORM, Prisma, Visual Studio Code, Angular, Express, Node.js, TypeScript, JWT, Node Package Manage(npm).

4.2.7 Identificación de requerimientos funcionales y no funcionales

En este paso se identificaron y detallaron los requerimientos funcionales y no funcionales del sistema, estableciendo las características y especificaciones que deberá cumplir el sistema para satisfacer las necesidades de los usuarios.

Los requisitos funcionales del Tempus Victor más significativos fueron:

- RF1 Registrar usuario.

- RF8 Crear nota.
- RF16 Crear tarea.
- RF25 Vista matriz Eisenhower.
- RF34 Vista calendario.
- RF41 Temporizador Pomodoro.
- RF55 Métricas de tareas.
- RF56 Métricas de sesiones Pomodoro.

Es importante resaltar que la parte del sistema con más utilidad para el estudiante es la parte de gestión de tareas con la matriz Eisenhower y el seguimiento de tiempos por medio del método Pomodoro. Para ver la descripción completa de los requerimientos dirigirse a la sección de **Requerimientos funcionales**.

Los requisitos no funcionales del sistema se definen como los requisitos que no son funciones específicas del sistema, pero que afectan de alguna manera. Los requisitos no funcionales del sistema web de organización personal identificados fueron:

- RNF1 Autenticación.
- RNF2 Robustez.
- RNF3 Funcionalidad.

Para ver la descripción completa de los requerimientos dirigirse a la sección de **Requerimientos No Funcionales**.

4.2.8 Análisis y diseño del diagrama relacional

En este paso se realizó el análisis y diseño del diagrama relacional de la base de datos, definiendo la estructura y las relaciones entre las diferentes entidades que compondrán el sistema.

El diagrama relacional es un modelo de datos que representa la estructura de la base de datos. El diagrama relacional de Tempus Victor se basa en el modelo de tres capas de la arquitectura del sistema.

La capa de datos del sistema consta de las siguientes Entidades:

- Usuario: Representa a los usuarios del sistema.
- Configuración: Representa la configuración del sistema asociada a la cuenta.
- Nota: Almacena las notas que los usuarios pueden crear.
- Categoría: Almacena las categorías personalizadas del usuario y las del sistema.
- Tareas: Almacena las tareas que el usuario puede crear.

- Eventos: Guarda los eventos de las tareas.
- Sesiones Pomodoro: Registra las sesiones de trabajo con la técnica Pomodoro.
- Eventos Pomodoro: Guarda los eventos de las sesiones Pomodoro.

Para ver la descripción completa dirigirse a la sección de [Modelo de la base de datos](#).

4.2.9 Identificación y análisis de riesgo

En este paso se identificaron y analizaron los posibles riesgos asociados al desarrollo del sistema, evaluando su probabilidad e impacto y planteando estrategias de mitigación. El análisis de riesgos se realizó utilizando técnicas como la tormenta de ideas, el análisis de causa-efecto y el análisis de escenarios. El objetivo del análisis fue identificar los riesgos potenciales del proyecto y desarrollar un plan para mitigarlos.

Entre los riesgos identificados se encuentran:

- Mala estimación de complejidad del proyecto.
- Tiempo limitado para el desarrollo.
- Planificación demasiado optimista.

Se crearon sus respectivas fichas de riesgo y se describieron los planes de mitigación en caso de que ocurra dicho riesgo.

Para ver la descripción completa de los riesgos dirigirse a la sección de [Análisis de riesgos](#).

4.2.10 Corrección y actualización de la documentación

En este paso se realizaron las correcciones y actualizaciones necesarias en la documentación, asegurando que toda la información reflejara con precisión el progreso y las decisiones tomadas durante esta iteración.

4.2.11 Resumen Iteración

La Iteración 1 sentó las bases para el desarrollo del sistema "Tempus Victor", proporcionando una comprensión clara de la problemática, una propuesta de solución bien fundamentada y un plan de acción detallado para las fases del proyecto posteriores.

4.3. Segunda iteración

En esta sección se describe el trabajo realizado durante la iteración 2 del desarrollo del sistema web "Tempus Victor". Durante esta fase, se avanzó en la implementación de componentes clave del sistema, basándose en las bases establecidas en la iteración 1. Siguiendo el modelo en espiral, esta iteración se centró en la validación de la factibilidad del proyecto, el diseño y desarrollo de interfaces gráficas y la identificación de reglas de negocio.

4.3.1 Establecimiento de objetivos para la iteración actual

En este paso se definieron los objetivos específicos para la iteración 2, asegurando que todas las actividades estuvieran alineadas con los objetivos del proyecto y que se pudieran medir los avances de manera efectiva.

Los objetivos de la segunda iteración se definieron como:

- Realizar el análisis de factibilidad económica.
- Realizar el análisis de factibilidad técnica.
- Definir la arquitectura del sistema a desarrollar.
- Diseñar las primeras interfaces gráficas del sistema.
- Plantear y definir los casos de uso para el sistema.
- Identificación de las primeras reglas del negocio.
- Desarrollar el CU1 Registrar usuario, y el CU2 Iniciar sesión.
- Hacer pruebas y corregir errores en el sistema.
- Actualizar la documentación hasta la segunda iteración.
- Planificar qué se requiere para la siguiente iteración.

4.3.2 Análisis de factibilidad económica

En este paso se realizó un análisis detallado de los costos asociados al desarrollo del sistema y de la estimación de esfuerzo a lo largo del tiempo disponible, todo esto para determinar si el proyecto era rentable o no.

Entre los factores considerados en el análisis se encuentran:

- Estimación de esfuerzo
- Estimación de costos

Se calculó una estimación de esfuerzo de **5 meses y 23** días para el desarrollo, la cual es factible considerando que sólo se tiene un semestre para su implementación.

Se calculó una estimación de costos de **\$0.00 MXN** por lo que se consideró como un proyecto rentable.

Análisis de costos

Para la realización de este análisis se hará uso de la métrica de punto de función. La métrica de punto de función (PF) tiene como objetivo principal cuantificar la funcionalidad desde la perspectiva del usuario final. Esto permite comparar y estimar proyectos de desarrollo de software de manera objetiva. El PF se basa en la idea de que la funcionalidad de un sistema se puede descomponer en diferentes elementos funcionales y cada elemento se mide en términos de su tamaño funcional. Los puntos de función se utilizan como una unidad de medida para cuantificar esta funcionalidad.

Para obtener el PF se consideran los valores de dominio de la información que se definen en la Tabla 2.

Tabla 2: Análisis de puntos por función

Tipo/Complejidad	Baja	Media	Alta
(EI) Entrada externa	3 PF	4 PF	6 PF
(EO) Salida externa	4 PF	5 PF	7 PF
(EQ) Consulta Externa	3 PF	4 PF	6 PF
(ILF) Archivo lógico interno	7 PF	10 PF	15 PF
(EIF) Archivo de interfaz externo	5 PF	7 PF	10 PF

Nota: Valores de PF indicados para cada tipo de componente (PrezDev, 2015)

Para poder realizar el cálculo de los puntos por función, debemos tomar como base la tabla anterior y la tabla de requerimientos funcionales, específicamente la columna de PF, la cual especifica de qué tipo es el requerimiento.

En la Tabla 3 se realiza la multiplicación del total de requerimientos de cada tipo por el valor estándar del mismo, obteniendo así el valor total en cada tipo, posteriormente se realiza la suma de cada uno de los totales para obtener el PFSA (Puntos de Función Sin Ajustar).

Tabla 3: Análisis de puntos de función sin ajustar

Tipo/Complejidad	Baja	Media	Alta
(EI) Entrada externa	9 x 3 PF	7 x 4 PF	3 x 6 PF
(EO) Salida externa	5 x 4 PF	1 x 5 PF	0 x 7 PF
(EQ) Consulta externa	3 x 3 PF	5 x 4 PF	8 x 6 PF
(ILF) Archivo lógico interno	2 x 7 PF	2 x 10 PF	4 x 15 PF
(EIF) Archivo de interfaz externo	0 x 5 PF	0 x 7 PF	0 x 10 PF
PFSA	269		

Nota: Elaboración propia

A continuación, se muestra de manera más específica el total de puntos de cada uno de los tipos.

- Entrada externa (EI): Total: 73
- Salida externa (EO): Total: 25
- Consulta externa (EQ): Total: 77
- Archivo lógico interno (ILF): Total: 94
- Archivo de interfaz externo (EIF): Total: 0
- Puntos de función sin ajustar (PFSA): 269

Los puntos por función obtenidos servirán para realizar el cálculo de los puntos por función ajustados. Para realizar este cálculo también se necesita obtener el factor de ajuste. El cálculo del factor de ajuste está basado en 14 características generales de los sistemas que miden la funcionalidad general de la aplicación. A cada característica se le atribuye un peso que varía de 0 a 5 e indica el grado o nivel de influencia que cada característica tiene en la aplicación que está siendo estudiada. A continuación, la *Tabla 4* presenta las 14 características generales de los sistemas listadas con su respectivo puntaje para el desarrollo de este sistema en concreto. Tal como se explicó en el párrafo anterior, el puntaje se establece en un rango de 0 a 5, siendo 0 un puntaje de presencia o peso nulo dentro del sistema y siendo 5 un puntaje de presencia o peso esencial en el sistema.

Tabla 4:Tabla de factor de ajuste

Características Generales del Sistema (GSC)	Puntaje
Comunicación de datos	5
Procesamiento distribuido	2
Objetivos de rendimiento	0
Configuración del equipamiento	0
Tareas de transacciones	0
Entrada de datos en línea	5
Interfaz con el usuario	3
Actualizaciones en línea	0
Procesamiento complejo	2
Reusabilidad del código	0
Facilidad de implementación	0
Facilidad de operación	1
Instalaciones múltiples	2
Facilidad de cambios	2

Nota: Elaboración propia

$$VAF = 0,65 + 0,01 \cdot \sum_{i=1}^{14} F_i = 0,65 + 0,01 \cdot 22$$

$$VAF = 0,87$$

Siendo F_i el valor adjudicado a cada GSC y VAF el factor de ajuste calculado.

Teniendo todos los datos anteriores se realizó el cálculo de los puntos de función (PF) con la siguiente formula:

$$PFA = PFS A \cdot VAF$$

Donde: PSFA: Puntos de función sin ajustar, PFA: Puntos de función ajustados

$$PFA = 269 \cdot 0,87$$

$$PFA = 234,03 \approx 234$$

Una vez realizado el ajuste el total de puntos por función da un aproximado de 234 puntos, este resultado permite realizar el cálculo de la estimación de esfuerzo y costos.

Estimación de esfuerzo

Esta sección se refiere a la cantidad de trabajo necesario para desarrollar la aplicación. Este esfuerzo se mide en horas/hombre, meses/hombre, años/hombre. Este cálculo se realiza teniendo en cuenta que los puntos de función son en cierto modo una medida subjetiva y que la cantidad de horas/hombre por punto de función es algo difícil e impreciso de valorar, de forma global.

Una tabla común que se utiliza para estimar el esfuerzo de desarrollo necesario para un proyecto de software es la tabla de líneas de código por punto de función de la International Function Point Users Group (IFPUG). Esta tabla muestra el número de líneas de código que se requieren típicamente para implementar un punto de función en diferentes lenguajes de programación. La tabla mencionada corresponde a la Tabla 5 siguiente:

Tabla 5: Estimación de esfuerzo

Lenguaje	Horas PF promedio	Líneas de código por PF
Ensamblador	10	300
Lenguajes de bajo nivel	5	50
Lenguajes de 3ra. generación	4	25
Lenguajes de 4ta. generación	2	10
Lenguajes de 5ta. generación	1	5

Nota: Impacto variable en el tiempo y la eficiencia de desarrollo en los lenguajes de programación

Para realizar el cálculo de horas-hombre se usa la siguiente fórmula:

$$H/H = PFA \cdot \text{Horas PF promedio}$$

Donde las PFA son los puntos por función ajustados para las horas promedio y, las horas promedio, para el caso de este proyecto específicamente en la parte de tecnologías, se hará uso de un lenguaje de 4ta generación el cual es TypeScript; por lo tanto, los valores quedan de la siguiente manera.

$$H/H = 234 \cdot 2$$

$$H/H = 468 \text{ Horas Hombre}$$

Al ser un proyecto realizado por una sola persona, se tiene contemplado que para el desarrollo del sistema se trabajan 5 horas diarias de lunes a domingo, lo que en total suma 7 días; para este estudio en específico, el cálculo de la estimación de duración del proyecto fue el siguiente:

4 horas diarias de trabajo

5 días de trabajo semanal

Para este estudio en específico, se asume que un mes tiene un promedio de 4 semanas para calcular la duración del proyecto. Es importante destacar que la duración real de un mes puede variar ligeramente en la realidad, ya que algunos meses pueden tener más o menos de 4 semanas.

$$4 \text{ horas} \cdot 5 \text{ días} = 20 \text{ horas a la semana}$$

$$\frac{468 \text{ horas hombre}}{20 \text{ horas/semana}} = 23,4 \approx 23 \text{ semana}$$

$$\frac{23 \text{ semanas}}{4 \text{ semanas/mes}} = 5.75$$

Tomado las consideraciones anteriormente descritas, desarrollar el sistema trabajando 4 horas diarias de lunes a viernes nos daría como resultado un estimado de **5 meses y 23 días** de desarrollo.

Estimación de esfuerzo del proyecto

Con la incorporación de este último dato, se procede a llevar a cabo el cálculo del presupuesto del proyecto. Para tal fin, se requiere considerar las siguientes premisas:

1. El proyecto se elaboró con la participación de un único desarrollador.
2. Para el desarrollo del sistema no se cuentan con otros gastos, todo el hardware es personal, por lo que no involucra un gasto. En cuanto al software de desarrollo y las tecnologías utilizadas son de licencia gratuita.
3. No se prevé la asignación de fondos para sueldos.

Teniendo en cuenta las premisas anteriores se procede a realizar el siguiente cálculo:

Cálculo del presupuesto del proyecto:

- Número de desarrolladores: 1
- Otros costos del proyecto: \$0.00 MXN
- Sueldo mensual: \$0.00 MXN

Costo = (No. de desarrolladores · Duración proyecto en meses · sueldos) + otros costos

Costo = $(1 \cdot 6 \cdot 0) + 0,00$

Costo = \$0,00MXN

En conclusión, el análisis de costos revela que el costo total estimado para llevar a cabo el desarrollo del sistema es de \$0.00 MXN. Esto sugiere que el proyecto es financieramente viable y se ajusta a las expectativas presupuestarias, lo que contribuye a su factibilidad y éxito potencial.

4.3.3 Análisis de factibilidad técnica

En este paso se evaluaron las capacidades técnicas necesarias para el desarrollo del sistema, identificando las tecnologías más adecuadas y los recursos técnicos disponibles para asegurar que el proyecto fuera técnicamente viable.

Para determinar si un sistema propuesto puede ser desarrollado, se debe realizar un análisis de factibilidad técnica; este análisis evalúa la disponibilidad de recursos, como hardware y software, así como la competencia del personal encargado de desarrollar el sistema.

Entre los factores considerados en el análisis se encuentran:

- El sistema operativo.
- Lenguaje de desarrollo.
- Sistema Gestor de base de datos.
- Características del hardware disponible.
- El análisis concluyó que el sistema podría ser desarrollado con las tecnologías y recursos disponibles.

Sistema operativo

Este componente es un factor importante que considerar, dado que debe satisfacer ciertos criterios fundamentales como estabilidad, gestión eficiente, velocidad, facilidad de uso y capacidad para soportar múltiples usuarios. A continuación, se presenta una lista con distintos sistemas operativos que cumplen con las características descritas:

Linux Ubuntu.

Windows 10/11.

MacOS Ventura.

Lenguaje de desarrollo

El lenguaje de desarrollo debe cumplir con las siguientes características:

Soporte en base de datos.

Baja curva de aprendizaje para que los desarrolladores puedan trabajar con él.

Compatible con los sistemas operativos y las tecnologías web que se requieren para implementar el sistema.

A continuación, se presenta una lista con los diferentes lenguajes de desarrollo que cumplen con las características mencionadas anteriormente.

- TypeScript/JavaScript.
- PHP.
- Java.
- Python.

Sistema Gestor de base de datos

Este es un factor muy importante ya que determinara la manera en que es guardada la información, la velocidad de procesamiento y el respaldo de los datos.

El sistema gestor de base de datos debe cumplir con las siguientes características:

- Ser estable.
- Soportar grandes cantidades de información.
- Tener conexión con diferentes lenguajes de programación.

A continuación, se presenta una lista de base de datos que cumplen con las características arriba mencionadas:

- MySQL.
- Oracle DB.
- PostgreSQL.
- Microsoft SQL Server.

Características del hardware disponible para el desarrollo

La Tabla 6 contiene las especificaciones del equipo de computo con el que se cuenta para el desarrollo del proyecto.

Tabla 6:Especificación del hardware disponible

Equipo	Elemento	Capacidad
Thinkpad T480	Memoria RAM	8GB
	Almacenamiento	256GB
	Procesador	Intel Core i7-8th
	Sistema operativo	Windows 10

Nota: Elaboración propia

El análisis de factibilidad técnica ha determinado que el proyecto es viable desde el punto de vista técnico. El equipo y los recursos necesarios están disponibles y el desarrollador cuenta con las habilidades y conocimientos necesarios para completar el proyecto con éxito.

4.3.4 Tecnologías seleccionadas

Después de realizar una comparación entre distintas tecnologías de desarrollo que existen, se concluyó que las que mejor se adaptan a la propuesta del sistema son las siguientes:

Frontend:

Angular es un Framework web de código abierto desarrollado por Google (Angular, 2024). Es una herramienta poderosa y flexible que se puede utilizar para crear aplicaciones web complejas. Angular fue elegido como tecnología de Frontend debido a su capacidad para crear aplicaciones web complejas y escalables.

Angular es una buena opción para el desarrollo de este sistema porque:

- Es un Framework moderno y flexible que permite crear aplicaciones web de alta calidad.
- Cuenta con una gran comunidad de desarrolladores y recursos disponibles.
- Es una tecnología escalable que puede soportar grandes volúmenes de usuarios.

Backend:

Express es un Framework de desarrollo de APIs REST de código abierto basado en Node.js (Express, 2024). Se utiliza para crear APIs RESTful fáciles de usar y escalables. Express fue elegido como tecnología de Backend debido a su flexibilidad y rendimiento. Express es una buena opción para el desarrollo de este sistema porque:

- Es un Framework ligero y eficiente que permite crear APIs RESTful de alta calidad.
- Cuenta con una gran comunidad de desarrolladores y recursos disponibles.
- Es una tecnología escalable que puede soportar grandes volúmenes de tráfico.

Base de datos:

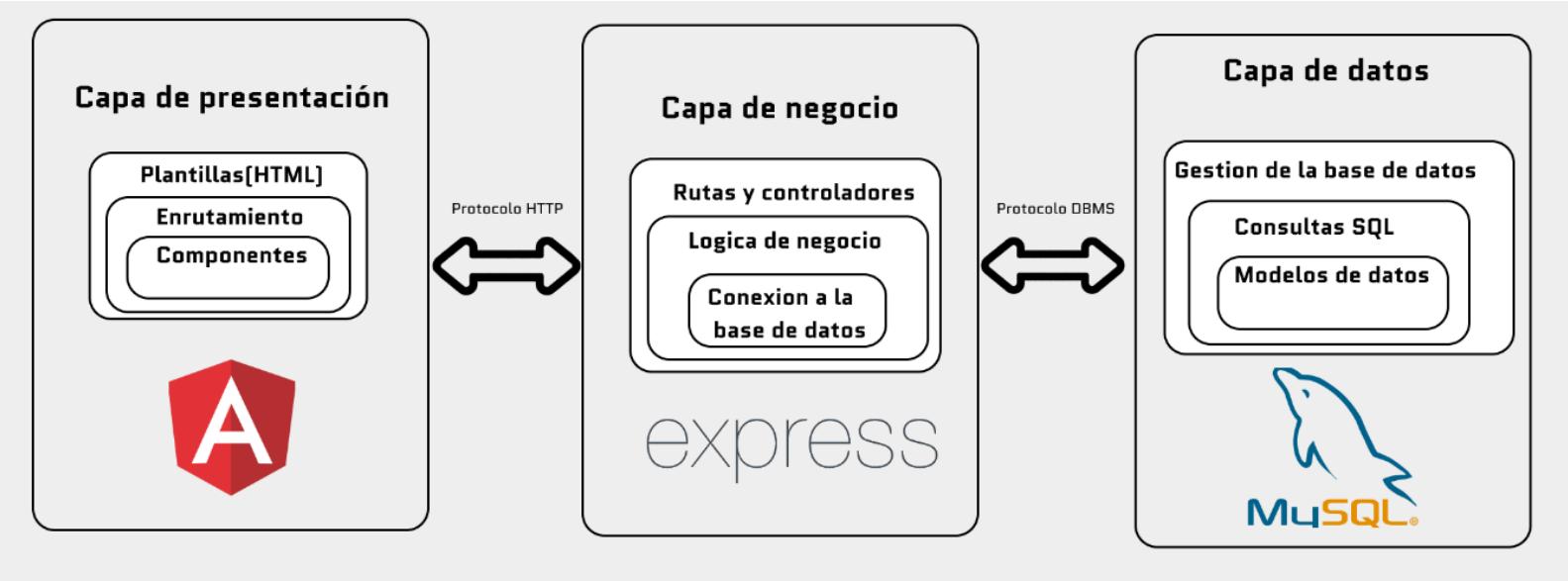
MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto (MySQL, 2024). Es una herramienta popular y confiable que se puede utilizar para almacenar grandes cantidades de datos. MySQL fue elegido como la tecnología de base de datos debido a su facilidad de uso e implementación; MySQL es una buena opción para el desarrollo de este sistema porque:

- Es un sistema de gestión de bases de datos popular y fiable.
- Es una tecnología escalable que puede soportar grandes volúmenes de datos.
- Es fácil de usar e implementar.

4.3.5 Arquitectura del sistema

La arquitectura del sistema se basa en un modelo de tres capas, que separa las responsabilidades en tres niveles: presentación, negocio y datos. A continuación, se describe de manera más detallada cada una de las capas de la arquitectura, así como el funcionamiento y la manera en la que interactuarán unas con otras.

Imagen 3:Arquitectura del sistema de tres capas



Nota: Elaboración propia

Capa de presentación: La capa de presentación es la interfaz con el usuario. En este caso, se utiliza el Framework de Angular para el desarrollo de la capa de presentación. Funcionamiento: Cuando el usuario realiza una acción en la interfaz de usuario de la aplicación, el Framework Angular genera una solicitud HTTP al servidor, el verbo HTTP y los datos de la

solicitud se especifican en el código de Angular; los datos de la solicitud se envían en formato JSON. Esta capa debe mostrar las páginas web y los componentes que forman la interfaz con el usuario. La capa de presentación debe comunicarse con la capa de negocio para obtener datos y realizar acciones. Algunos de los componentes más importantes de esta capa son:

- Página de inicio.
- Página de registro.
- Página de inicio de sesión.

Capa de negocio: La capa de negocio es la responsable de la lógica del negocio. Se utiliza el Framework Express para el desarrollo de toda la lógica del negocio. Esta capa debe mostrar los servicios, las rutas y las clases que implementan la lógica del negocio. La capa de negocio debe comunicarse con la capa de datos para almacenar y recuperar datos.

Algunos de los servicios que se incluyen en esta capa son:

- Servicios de notas.
- Servicios de tareas
- Servicios de Pomodoro.

Capa de datos: La capa de datos es responsable del almacenamiento de los datos. Se utiliza MySQL para el almacenamiento de los datos. Esta capa debe mostrar la base de datos y las tablas que almacenan los datos. La capa de datos debe regresar la información dependiendo las consultas y peticiones realizadas por la capa de negocio.

Algunas de las tablas que contendrá la capa de datos son:

- Tabla de notas.
- Tabla de tareas.
- Tabla de métricas sobre tiempos.

Comunicación entre capas: La comunicación entre las capas se realiza a través de una API REST.

- La capa de presentación realizará peticiones a la capa de negocio a través de la API REST.
- La capa de negocio realizará peticiones a la capa de datos a través de la API REST.
- La API REST se implementará utilizando el Framework Express.
- La API REST proporcionará un conjunto de endpoints que podrán ser utilizados por la capa de presentación para acceder a los datos y realizar acciones en el sistema.

4.3.6 Diseño de interfaces gráficas del sistema

En este paso se llevó a cabo el diseño inicial de las interfaces gráficas del sistema, asegurando que fueran intuitivas y fáciles de usar, basándose en principios de diseño centrado en el usuario. El diseño y análisis de las interfaces de usuario es una tarea fundamental en el desarrollo de cualquier sistema informático. Este proceso consiste en definir cómo se verá y cómo funcionará el sistema desde la perspectiva del usuario. Se debe crear una interfaz que sea atractiva y fácil de usar, que cumpla con las necesidades de los usuarios y que se ajuste a los objetivos del sistema.

Las pantallas más importantes para lograr conseguir el objetivo general, así como los especificados son las siguientes pantallas:

Tabla 7: Pantallas diseñadas importantes

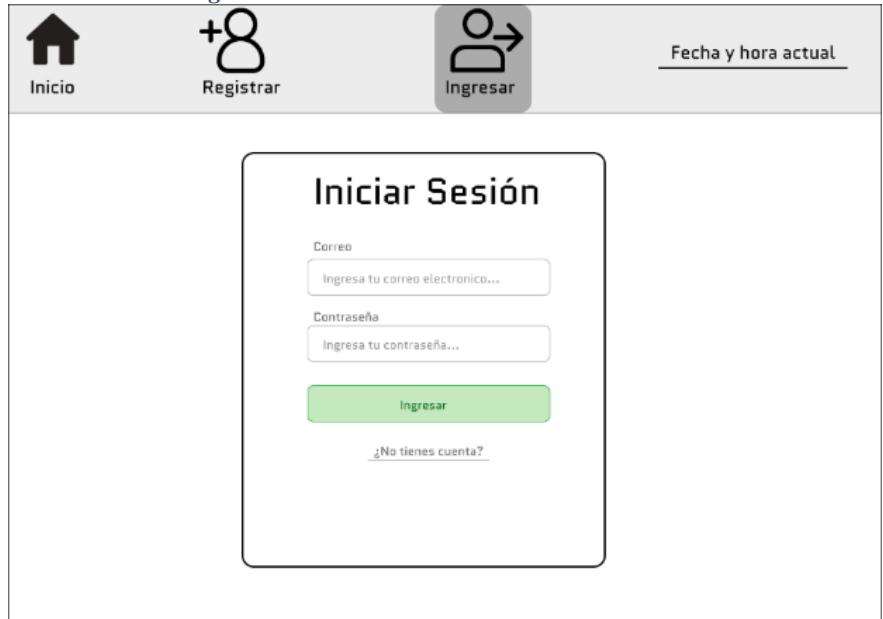
IU2 Registro de usuario

Objetivo	Diseño
Permitir al usuario crear una nueva cuenta, registrándose en el sistema.	Pantalla de Registro de usuario que requiere nombre de usuario, correo y contraseña para poder registrar un nuevo usuario.
Entradas	<p><i>Imagen 4: Diseño Iu2 Pantalla de Registro de usuario</i></p> <div style="border: 1px solid #ccc; padding: 10px; width: fit-content;"> <p>Registrar</p> <p>Nombre de Usuario Ingresa un nombre de usuario...</p> <p>Correo Ingresa tu correo electrónico...</p> <p>Contraseña Ingresa tu contraseña...</p> <p>Contraseña Confirma tu contraseña...</p> <p>Confirmar Registro</p> </div>
Salidas	<ul style="list-style-type: none"> - Campos de textos requeridos - Mensaje de <i>Error</i> en cada campo si es que está vacío o no cumple con su especificación. - Mensaje de <i>Error</i> si las contraseñas no son iguales. - Mensaje de advertencia si el correo electrónico ya existe. - Mensaje de <i>Éxito</i> si el registro de usuario fue exitoso.

Nota: Elaboración Propia

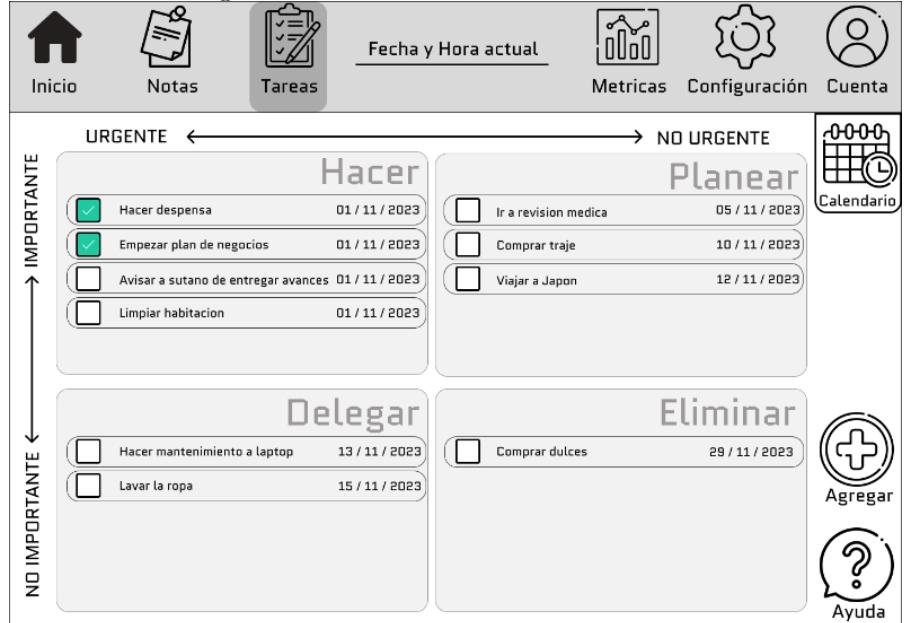
IU3 Inicio de sesión

Objetivo	Diseño
Permitir al usuario iniciar sesión en su cuenta y acceder al sistema.	Pantalla de Inicio de sesión que requiere correo y contraseña previamente registrados para poder acceder al sistema.
Entradas	<p>Campos de texto: Correo electrónico, contraseña</p> <p>Botones:</p> <ul style="list-style-type: none"> - Ingresar: Manda los datos al servidor para verificar si el correo existe, si la contraseña es correcta para este correo lleva a la IU4 Vista de Tareas Eisenhower - ¿No tienes una cuenta?: Lleva al usuario a la IU2 Pantalla de Registro de usuario
Salidas	<ul style="list-style-type: none"> - Campos de textos requeridos. - Mensaje de <i>Error</i> en cada campo si es que está vacío o no cumple con su especificación. - Mensaje de <i>Error</i> si el usuario no existe. - Mensaje de <i>Error</i> si la contraseña es incorrecta.

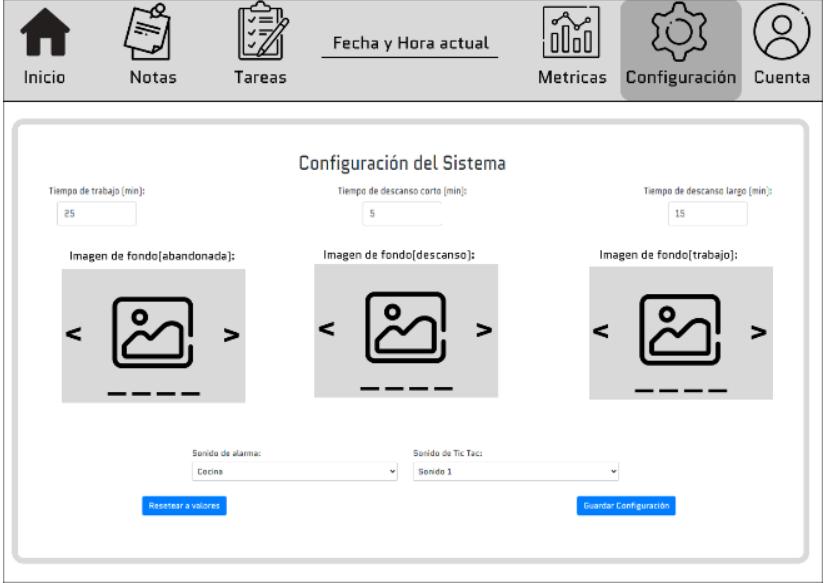
Imagen 5: Diseño IU3 Pantalla de Inicio de sesión*Nota: Elaboración Propia*

IU4 Tareas Eisenhower

Objetivo	Diseño
Permitir al usuario ver todas las tareas sin completar en una vista de matriz Eisenhower.	En esta vista se pueden ver las tareas divididas en cuadrantes dependiendo de su urgencia e importancia, botones para interactuar con las tareas y las vistas de tareas.
Entradas	<p>Checkbox en cada tarea, para poder completarla.</p> <p>Cada tarea es un objeto que se puede arrastrar entre los distintos cuadrantes.</p> <p>Botones:</p> <ul style="list-style-type: none"> - Agregar: Lleva a la IU8 Pantalla para Agregar Tarea - Ayuda: Lleva a la IU30 Pantalla de Ayuda para esta pantalla - Calendario: Cambia de vista a la IU6 Vista de Tareas Calendario
Salidas	<ul style="list-style-type: none"> - Listas de tareas, las cuales contienen checkbox, título de la tarea, cantidad de Pomodoros estimados y fecha y hora límites. - Mensaje de Información de cada elemento usando ToolTips.

Imagen 6: Diseño IU4 Vista de Tareas Eisenhower*Nota: Elaboración Propia*

IU6 Tareas Calendario	
Objetivo	Diseño
Permitir al usuario ver todas las tareas sin completar en una vista de calendario.	En esta vista se pueden ver las tareas acomodadas en el día de la fecha límite si es que tiene, si no tiene fecha límite aparece en una lista de tareas sin fecha límite. Imagen 7: Diseño IU6 Vista de Tareas Calendario
Entradas	
Checkbox en cada tarea, para poder completarla. Cada tarea es un objeto que se puede arrastrar entre los días del calendario.	
Botones:	
- Agregar: Lleva a la pantalla IU8 Pantalla para Agregar Tarea	
- Ayuda: Lleva a la pantalla IU30 Pantalla de Ayuda para esta pantalla	
- Eisenhower: Cambia de vista a la IU4 Vista de Tareas Eisenhower	
Salidas	
- Listas de tareas, las cuales contienen checkbox, título de la tarea, cantidad de Pomodoros estimados y fecha y hora límites.	
- Mensaje de Información de cada elemento usando ToolTips.	
IU17 Notas	
Objetivo	Diseño
Permitir al usuario ver todas las notas creadas en una lista de notas.	Pantalla que muestra la lista de todas las notas creadas en la cuenta del usuario.
Entradas	
Botones:	
- Agregar: Lleva a la IU19 Pantalla para Agregar Nota	
- Ayuda: Lleva a la IU30 Pantalla de Ayuda para esta pantalla	
Salidas	
- Listas de notas, las cuales contienen título, descripción, fecha de creación y fecha de última modificación.	
- Mensaje de información de cada elemento usando ToolTips.	
Imagen 8: Diseño IU17 Pantalla de Notas 	
<i>Nota: Elaboración Propia</i>	

IU23 Cuenta	
Objetivo	Diseño
Permitir al usuario ver la información de su cuenta dando la opción de gestionar la contraseña, la foto de perfil y el nombre de usuario.	La pantalla contiene la foto de perfil de la cuenta, el nombre de usuario, el correo electrónico y los botones para gestionar la cuenta.
Entradas	
Botones: - Cerrar Sesión: Lleva a la IU24 Pantalla para Cerrar Sesión - Editar: Lleva a la IU25 Pantalla para Editar Cuenta - Cambiar Contraseña: Lleva a la IU26 Pantalla para Cambiar Contraseña - Eliminar Cuenta: Lleva a la IU27 Pantalla para Eliminar Cuenta	 Imagen 9: Diseño IU23 Pantalla de Cuenta La interfaz de usuario (UI) para la cuenta muestra un menú superior con iconos para Inicio, Notas, Tareas, Fecha y Hora actual, Metricas, Configuración y Cuenta. La sección central titulada "Mi cuenta" muestra una foto de perfil vacía y campos para Nombre de Usuario, Correo Electrónico y Correo Electrónico. A la derecha, hay botones para Cerrar Sesión, Editar, Cambiar Contraseña y Eliminar Cuenta.
Salidas	
- Mensaje de Información de cada elemento usando ToolTips.	
IU28 Configuración	
Objetivo	Diseño
Permitir al usuario gestionar la configuración del sistema, específicamente la configuración del temporizador Pomodoro.	La pantalla contiene los elementos que permiten gestionar la duración del tiempo de trabajo, los tiempos de descanso, las imágenes de fondo del temporizador dependiendo su estado, el sonido de fondo y el sonido de la alarma del temporizador.
Entradas	
Campos de texto: Duración de trabajo, duración descanso corto, duración descanso largo. Campos de selección: Sonido de alarma, sonido de fondo. Carrusel de imágenes: Carrusel estado abandonada, carrusel estado trabajando, carrusel estado descansando. Botones: - Resetear valores por defecto: Cambia los valores actuales por los valores por defecto del sistema. - Guardar valores actuales: Guarda los valores actuales en la base de datos del usuario actual.	 Imagen 10: Diseño IU28 Pantalla de Configuración del Sistema La interfaz de usuario para la configuración del sistema muestra un menú superior similar a la IU23. La sección central titulada "Configuración del Sistema" incluye campos para Tiempo de trabajo [min] (25), Tiempo de descanso corto [min] (5) y Tiempo de descanso largo [min] (15). Hay tres carruseles para Imagen de fondo: abandonada, descanso y trabajo, cada uno con un selector de imágenes y flechas para navegar. Sección inferior para Sonido de alarma y Sonido de Tic Tac, ambos con dropdowns para Cocina y Sonido 1. Botones para Restaurar a valores y Guardar Configuración.
Salidas	
- Mensaje de éxito si se resetearon los valores correctamente. - Mensaje de éxito si se guardaron los valores correctamente. - Mensaje de Información de cada elemento usando ToolTips	

IU29 Métricas	
Objetivo	Diseño
Permite al usuario ver las métricas de su cuenta con relación a las tareas y a las sesiones Pomodoro.	Esta pantalla contiene los diagramas de pastel que muestran las distintas proporciones de las tareas y sesiones Pomodoro que contiene la cuenta del usuario.
Entradas	<p>Botones:</p> <ul style="list-style-type: none"> - Regresar: Lleva a la <u>IU4 Vista de Tareas Eisenhower</u>
Salidas	<ul style="list-style-type: none"> - Mensaje de información de cada elemento usando ToolTips
	<p>Imagen 11: Diseño IU29 Pantalla de Métricas de la Cuenta</p> <p>Métricas</p> <p>Tareas</p> <p>Sesiones Pomodoro</p> <p>Notas: Elaboración Propia</p>
IU14 Temporizador Pomodoro	
Objetivo	Diseño
Permitir al usuario trabajar en la tarea seleccionada haciendo uso del temporizador Pomodoro el cual implementa la técnica Pomodoro.	Pantalla que implementa la técnica Pomodoro, que es un método de gestión del tiempo que consiste en usar un temporizador para trabajar en intervalos fijos, llamados Pomodoros, de 25 minutos de actividad, seguidos de 5 minutos de descanso, con pausas más largas cada cuatro Pomodoros.
Entradas	<p>Botones:</p> <ul style="list-style-type: none"> - Regresar: Lleva a la vista de tareas anterior - Iniciar: Inicia la sesión Pomodoro, y empieza a correr el tiempo del temporizador - Pausar: Pausa la sesión Pomodoro actual - Terminar: Termina la sesión Pomodoro actual - Habilitar/Deshabilitar Sonido: Gestiona el sonido de fondo del temporizador.
Salidas	<ul style="list-style-type: none"> - Mensaje de sesión Pomodoro creada. - Mensaje de Bienvenida al temporizador Pomodoro. - Información sobre la tarea, título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia. - Lista de tiempos de la sesión Pomodoro actual. - Modal para iniciar sesión Pomodoro. - Mensaje de Información de cada elemento usando ToolTips.
	<p>Imagen 12: Diseño IU14 Pantalla de Temporizador Pomodoro</p> <p>POMODORO</p> <p>Título: Hacer Despensa</p> <p>Descripción: -Huevos -Pechuga de Pollo 1kg -Arroz</p> <p>Comentarios: -Solo había medio kilo -Ya está el Pollo -Todavía falta arroz</p> <p>Regresar</p> <p>Tiempo total: 00 : 00 : 00</p> <p>Tiempo de trabajo: 00 : 00 : 00</p> <p>Tiempo descanso: 00 : 00 : 00</p> <p>Notas: Elaboración Propia</p>

Para ver la descripción completa dirigirse a la sección de [Interfaces Reales del Sistema](#).

4.3.7 Planteamiento de casos de uso

En este paso se desarrollaron casos de uso detallados que describen las interacciones entre los usuarios y el sistema, proporcionando una visión clara de cómo se utilizaría el sistema en diferentes escenarios.

Los casos de uso son descripciones de las interacciones entre los usuarios y el sistema. Los casos de uso se utilizan para definir los requisitos funcionales del sistema.

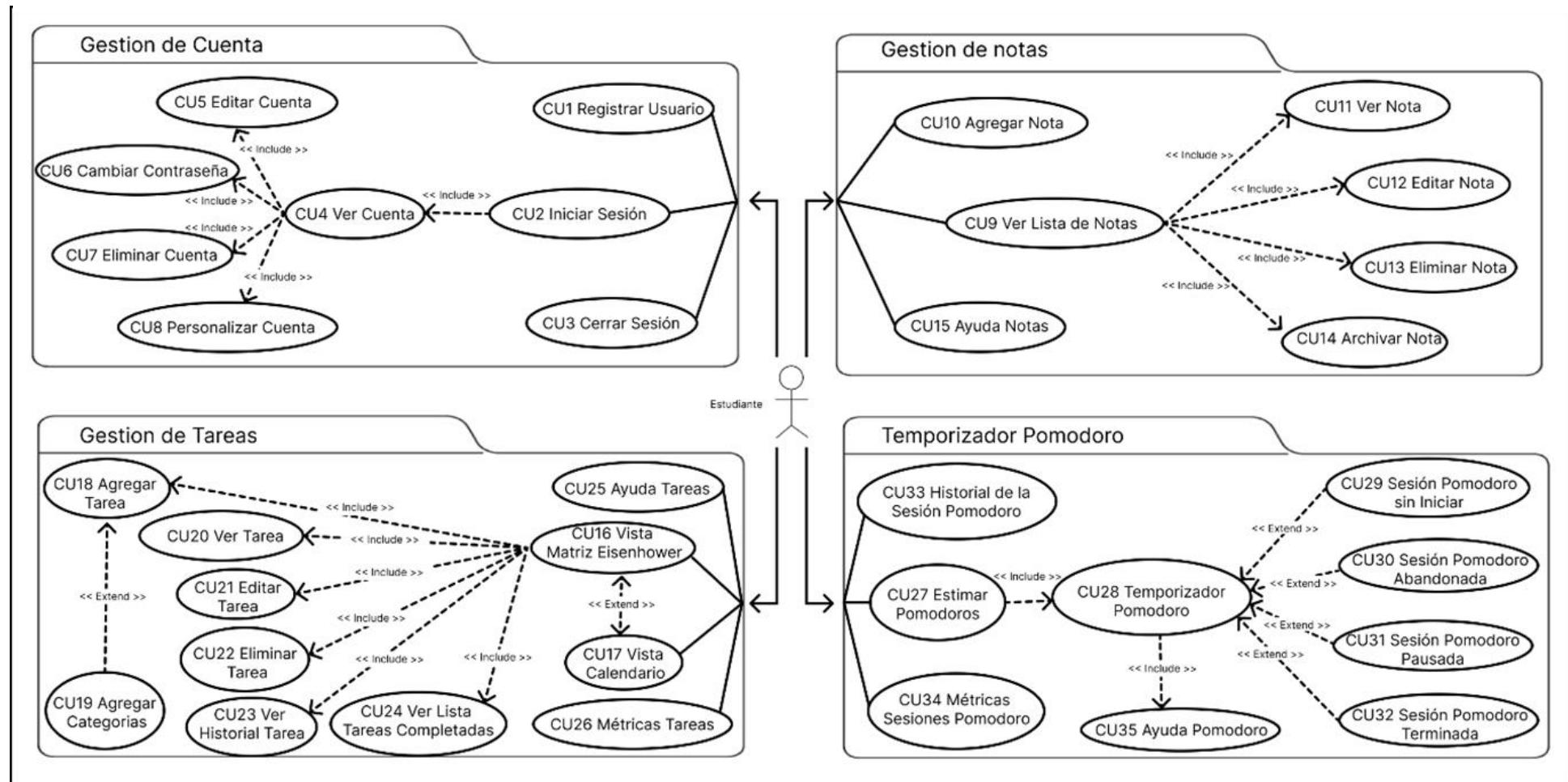
Los casos de uso del sistema web de organización personal se diseñaron para cubrir todas las funcionalidades del sistema, tanto administrativas como de organización personal.

El diagrama se divide en 4 partes esenciales del sistema, las cuales son:

- Gestión de cuenta: Aquí se incluyen los requisitos funcionales relacionados con la creación y modificación de una cuenta, así como la personalización de esta.
- Gestión de Notas: Aquí se incluyen los requisitos funcionales relacionados con la creación y modificación de las notas, crear nota, buscar nota, ver nota, editar nota, eliminar nota, seleccionar nota.
- Gestión de Tareas: Aquí se incluyen los requisitos funcionales relacionados con la creación y modificación de las tareas, crear tarea, ver historial tarea, seleccionar tarea, editar tarea, eliminar tarea, cambiar estado de tarea, agregar recordatorios, ver lista de tareas finalizadas, métricas de tareas, vista de calendario, vista de matriz Eisenhower.
- Temporizador Pomodoro: Aquí se incluyen los requisitos funcionales relacionados con el seguimiento en tiempo real de una tarea seleccionada por medio del método Pomodoro haciendo uso de un temporizado, métricas de sesiones Pomodoro, sesión Pomodoro desatendida, ver historial de sesión, estimación de tiempos, alerta sonora.

A continuación, se resume el diagrama de casos de uso para el sistema:

Imagen 13: Diagrama de casos de uso



Nota: Elaboración Propia

Para ver la descripción completa dirigirse a la sección de [Modelo de la base de datos](#).

4.3.8 Identificación y descripción de reglas del negocio

En este paso se identificaron y documentaron las reglas de negocio que rigen el funcionamiento del sistema, asegurando que todos los procesos se alineen con los objetivos del negocio para el sistema Tempus Victor.

Las reglas de negocio son restricciones que se aplican a los datos del sistema. Las reglas de negocio se utilizan para garantizar la integridad y la coherencia de los datos. Las reglas de negocio del sistema web de organización personal se diseñaron para cubrir todos los requisitos no funcionales del sistema.

Se identificaron las reglas del negocio relacionados con la gestión de cuenta, esto es todas las reglas que existen para: Registro de usuario, inicio de sesión, cierre de sesión, ver información de la cuenta, edición de la cuenta, cambio de contraseña y, la eliminación de la cuenta. Algunas de las reglas importantes que nacieron aquí son:

- [BR1](#): Un usuario solo puede registrarse proporcionando un nombre de usuario, una dirección de correo electrónico válida/única y una contraseña que cumpla con los requisitos de seguridad (Mínimo 6 caracteres).
- [BR3](#): El sistema debe validar que la contraseña sea la misma que ingresa en la confirmación de contraseña.
- [BR6](#): El sistema debe validar que la contraseña es correcta para el correo proporcionado
- [BR7](#): El usuario puede cerrar sesión en cualquier momento desde cualquier dispositivo.
- [BR9](#): El usuario puede editar su nombre de usuario y foto de perfil.
- [BR13](#): La eliminación de la cuenta es permanente y todos los datos asociados serán eliminados.

Para ver la descripción completa dirigirse a la sección de [Modelo del Negocio](#).

4.3.9 Desarrollo y corrección de interfaces gráficas del sistema

En este paso se procedió al desarrollo de las interfaces gráficas previamente diseñadas, seguido de una fase de corrección basada en pruebas iniciales para mejorar la usabilidad y funcionalidad.

En primer lugar, se tuvo que instalar lo necesario para el proyecto, lo primero que se realizó en esta parte fue empezar con el Frontend del sistema, sobre todo en la parte de registro e inicio de sesión al sistema. Para crear un proyecto en Angular se realizó lo siguiente:

Abrir el IDE Visual Studio Code, y crear una nueva carpeta llamada TEMPUS_VICTOR,

Una vez creada la carpeta se crearon otras dos más, Backend y Frontend. Se abrió una terminal dentro del VS Code y se ubicó en la carpeta de Frontend con el siguiente comando:

```
$ cd Frontend
```

Como es necesario instalar Node.js se procedió a descargar el instalador desde la página oficial, instalándolo siguiendo los pasos que se van solicitando.

Imagen 14: Descarga de instalador Node.js

Downloads

Latest LTS Version: 14.17.3 (includes npm 6.14.13)

Download the Node.js source code or a pre-built installer for your platform, and start developing today.



Nota: Captura tomada del sitio web oficial de Node.js

Una vez descargado Node.js se procedió a instalar Angular con el siguiente comando:

```
$ npm install -g @angular/cli
```

Posteriormente fue creado el nuevo proyecto con el siguiente comando:

```
$ ng new tempus-victor
```

El comando ng new solicitó información sobre las funciones que debe incluir la aplicación inicial, se aceptaron los valores predeterminados presionando la tecla Enter o Return. La CLI de Angular instala los paquetes npm de Angular necesarios y otras dependencias; esto

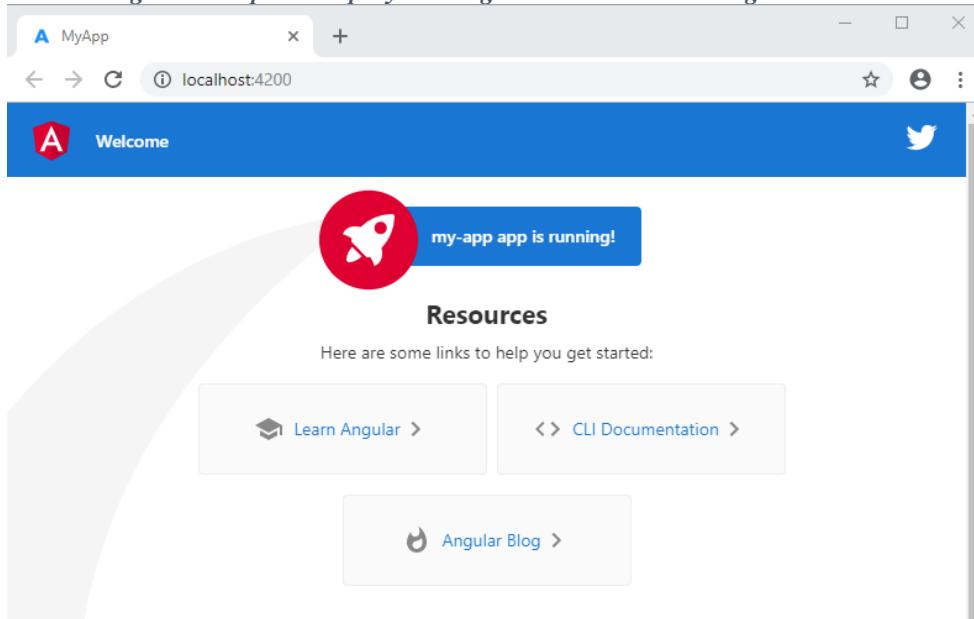
puede tardar unos minutos. La CLI crea el nuevo espacio de trabajo y una aplicación de bienvenida simple, lista para ejecutarse. Posteriormente se ejecutó el comando:

```
$ cd tempus-victor
$ ng serve --open
```

El comando `ng serve` inicia el servidor, observa los archivos y reconstruye la aplicación a medida que se realizan cambios en esos archivos. La opción (`--open`) simplemente abre automáticamente el navegador en `http://localhost:4200/`.

Si la instalación y configuración fue exitosa, se puede ver una página similar a la siguiente.

Imagen 15: Captura de proyecto Angular corriendo en navegador



Nota: Elaboración Propia

Una vez que se tuvo la aplicación base para el desarrollo del Frontend se crean los componentes que se tienen como objetivo, los cuales son los siguientes:

Register, Login, Cuenta, Config, generando los componentes con los siguientes comandos:

```
$ ng generate component components/register
$ ng generate component components/login
$ ng generate component components/cuenta
$ ng generate component components/config
```

Este comando genera cada componente que se necesita para cumplir con los objetivos de esta iteración. Cuando se crea un componente, éste se crea con 4 archivos: un `.html` para la parte de la estructura, un `.scss` o `.css` dependiendo como se configuró la instalación. En este caso, se

escogió el .scss para la parte de los estilos, un .spec.ts para hacer pruebas, y un .ts donde va toda la lógica del negocio y la interacción con los servicios y el Backend.

Imagen 16: Componente de Registro de usuario

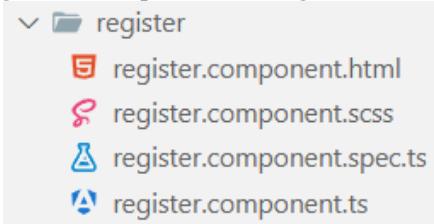


Imagen 17: Componente de Inicio de sesión

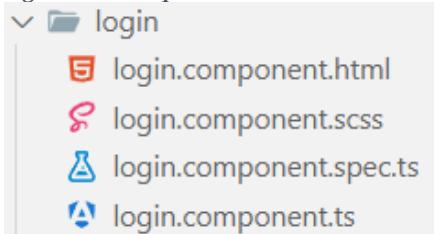


Imagen 18: Componente de Cuenta de usuario

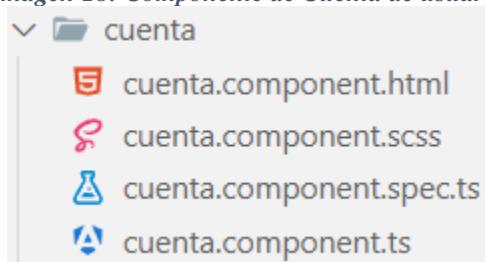
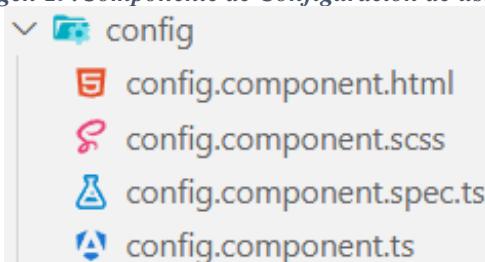


Imagen 19: Componente de Configuración de usuario



Una vez con dichos componentes se procedió a instalar lo necesario, pero ahora en la carpeta del Backend, que es donde se ubica el servidor.

En primer lugar, se necesita instalar express, por lo que es importante asegurarse de tener node.js instalado; una vez verificado se procede con la instalación según la guía de la página oficial de express:

Ya instalado Node.js, se crea un directorio para que contenga la aplicación y se convierte en el directorio de trabajo.

```
$ mkdir myapp  
$ cd myapp
```

Debe utilizarse el mandato npm init para crear un archivo package.json para la aplicación. Para obtener más información sobre cómo funciona package.json, es necesario consultar las especificaciones del manejo de package.json dentro de npm.

```
$ npm init
```

Este mandato solicita varios elementos como, por ejemplo, el nombre y la versión de la aplicación, sin embargo, sólo se requiere pulsar ENTER para aceptar los valores predeterminados para la mayoría de ellos.

Se especifica app.js o el nombre deseado para el archivo principal. Si se desea que sea index.js, se pulsa ENTER para aceptar el nombre de archivo predeterminado recomendado.

A continuación, se instala Express en el directorio de la aplicación y se guarda en la lista de dependencias. Ejemplo:

```
$ npm install express --save
```

Para instalar Express temporalmente y no añadirlo a la lista de dependencias, es necesario omitir la opción --save:

```
$ npm install express
```

Así es como logra instalarse express para el Backend.

Lo siguiente que se hace es crear la base de datos, la cual guardará la información de los próximos usuarios, para eso se requiere la instalación del ORM prisma y MySQL.

Para instalar MySQL:

Se descarga el instalador de la página oficial, se instala el archivo descargado, se aceptan los términos y condiciones, se elige una instalación completa, se ejecutan las instalaciones, se espera a que se instalen, se da next a los parámetros por defecto que vayan apareciendo y cuando se llegue a la parte de ingresar una contraseña, poner mucha atención para ingresar ésta, pues será muy útil más adelante. Se finaliza la instalación y se reinicia la computadora. Así es como queda instalado MySQL, y en especial el Workbench de MySQL. Una vez instalado lo anterior se procede a instalar el ORM de prisma de la siguiente manera:

Como primer paso, se accede al directorio de Backend:

```
$ cd Backend
```

A continuación, es necesario inicializar un proyecto TypeScript y se agrega Prisma CLI como una dependencia de desarrollo:

```
$ npm init -y  
$ npm install prisma typescript ts-node @types/node --save-dev
```

Esto crea una package.json configuración inicial para su aplicación TypeScript. A continuación, se inicializa TypeScript:

```
$ npx tsc --init
```

Ahora se puede invocar la CLI de Prisma prefijándola con npx:

```
$ npx prisma
```

A continuación, se configura el proyecto Prisma ORM creando un archivo de esquema Prisma con el siguiente comando:

```
$ npx prisma init
```

Este comando hace dos cosas:

- Crea un nuevo directorio llamado (/prisma) que contiene un archivo llamado schema.prisma, que contiene el esquema de Prisma con la variable de conexión de su base de datos y los modelos de esquema.
- Crea el archivo (.env) en el directorio raíz del proyecto, que se utiliza para definir variables de entorno (como la conexión de la base de datos).

Ahora se debe conectar la base de datos de la siguiente forma:

Para conectar la base de datos, se debe configurar el URL de la base de datos en el esquema de Prisma en la URL de conexión de la base de datos:

Imagen 20: URL de Conexión con la Base de Datos y Prisma

```
1 datasource db {  
2   provider = "mysql"  
3   url      = env("DATABASE_URL")  
4 }
```

En este caso, el URL se establece mediante una variable de entorno que se define en .env; lo siguiente es la forma por defecto para crear el URL de conexión con la base de datos:

```
mysql://USER:PASSWORD@HOST:PORT/DATABASE
```

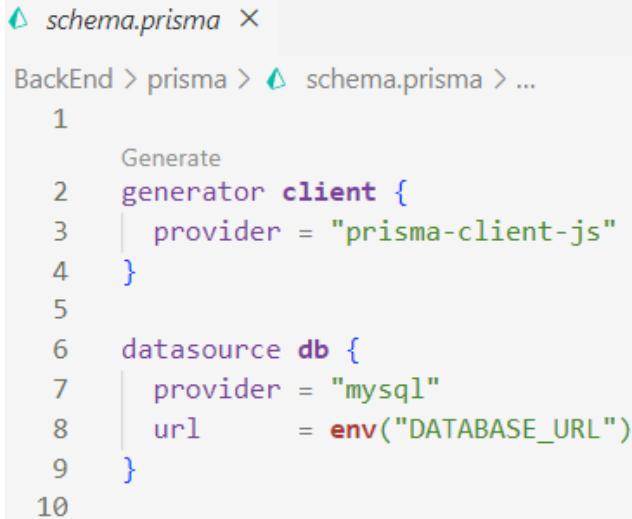
El siguiente URL es el que se utilizó para el desarrollo actual, contiene la contraseña definida en la instalación de MySQL, el puerto de acceso por defecto y el nombre de la base de datos a crear.

```
DATABASE_URL="mysql://root:password@localhost:3306/tempus_victor_db?schema=public"
```

Ahora que se cuenta con la conexión es necesario crear el esquema de la base de datos para las tablas o entidades necesarias para implementar los módulos objetivo para la iteración.

El esquema de prisma quedó de la siguiente manera:

Imagen 21: Esquema Prisma Conexión DB



```
BackEnd > prisma > schema.prisma > ...
1
  Generate
2  generator client {
3    provider = "prisma-client-js"
4  }
5
6  datasource db {
7    provider = "mysql"
8    url      = env("DATABASE_URL")
9  }
10
```

Imagen 22: Entidad Usuario Prisma



```
model User {
  idUser Int      @id @default(autoincrement())
  username String
  email   String  @unique
  password String
  image   String? @db.LongText
  setting Setting?
  notes Note[]
  tasks Task[]
  categories Category[]
}
```

Imagen 23:Entidad Configuración Prisma

```

24 model Setting{
25   idSetting Int @id @default(autoincrement())
26   pomodoroWorkDuration Int
27   shortBreakDuration Int
28   longBreakDuration Int
29   workImg Int
30   breakImg Int
31   neglectedImg Int
32   alarmSound Int
33   tictacSound Int
34   userId Int @unique
35   user User @relation(fields: [userId], references: [idUser], onDelete:
36   Cascade)
36 }

```

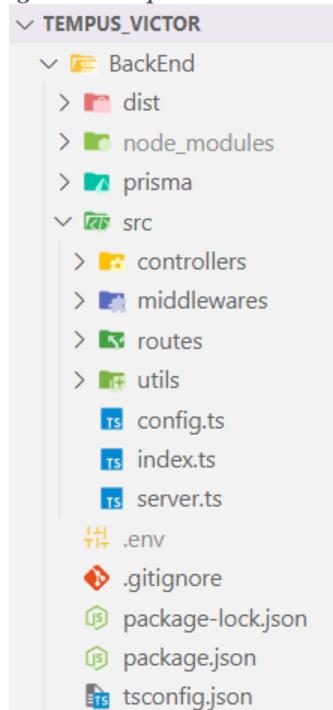
En consiguiente, es necesario migrar el esquema a la base de datos; para asignar el modelo de datos al esquema de la base de datos, se debe utilizar:

```
$ npx prisma migrate dev --name init
```

Este comando hace dos cosas:

- Crea un nuevo archivo de migración SQL para esta migración.
- Ejecuta el archivo de migración SQL contra la base de datos.

Ya con esto se tiene la base de datos creada en MySQL, corriendo y actualizada a las necesidades actuales. Ahora se puede empezar a desarrollar el Backend; el Backend en si es un archivo de servidor, pero por motivos de hacer un desarrollo ordenado se dividió el Backend en las siguientes carpetas:

Imagen 24: Carpeta Backend Inicial

Carpetas raíz: La carpeta raíz del proyecto contiene los siguientes archivos y carpetas:

Carpetas:

- dist: La carpeta dist contiene los archivos compilados de la aplicación.
- node_modules: La carpeta node_modules contiene las dependencias del proyecto.
- prisma: La carpeta prisma contiene los archivos de Prisma.
- src: La carpeta src contiene los archivos principales del proyecto.

Archivos:

- .env: El archivo .env se utiliza para almacenar variables de entorno sensibles, como la URL de la base de datos y la clave secreta. Este archivo no se debe incluir en el repositorio de Git, ya que contiene información confidencial.
- .gitignore: El archivo .gitignore especifica los archivos y carpetas que no se deben incluir en el repositorio de Git.
- package-lock.json: El archivo package-lock.json contiene una lista de las versiones exactas de las dependencias del proyecto que se instalaron cuando se ejecutó el comando npm install.
- package.json: El archivo package.json contiene información sobre el proyecto, como el nombre, la versión, las dependencias y los scripts.
- tsconfig.json: El archivo tsconfig.json contiene la configuración del compilador TypeScript.

Carpetas src: contiene los archivos principales del proyecto.

Carpetas controllers: La carpeta controllers contiene los controladores de la aplicación.

Los controladores son clases que se encargan de responder a las solicitudes HTTP.

Carpetas middlewares: La carpeta middlewares contiene los middlewares de la aplicación.

Los middlewares son funciones que se ejecutan antes de que se llame a un controlador. Los middlewares se pueden utilizar para realizar tareas como la autenticación o la validación de datos.

Carpetas routes: La carpeta routes contiene las rutas de la aplicación. Las rutas son especificaciones que definen cómo se mapean las solicitudes HTTP a los controladores.

Carpetas utils: La carpeta utils contiene utilidades para la aplicación. Las utilidades son funciones que se utilizan para realizar tareas comunes, como el formateo de fechas o la generación de tokens aleatorios.

Archivos:

- config.ts: El archivo config.ts contiene la configuración de la aplicación, como la URL de la base de datos y la clave secreta.
- index.ts: El archivo index.ts es el punto de entrada de la aplicación. Es donde se inicia el servidor y se registran las rutas.
- server.ts: El archivo server.ts contiene el código del servidor. Este código es responsable de escuchar las solicitudes HTTP y responder con los datos correspondientes.

Después de configurar todo lo necesario se desarrollaron las pantallas de la interfaz del sistema que hacen capaz la implementación de los casos de uso planeados:

IU1 Pantalla De Bienvenida al Sistema

IU2 Registro de Usuario

IU3 Inicio de Sesión

Nota: Para ver su especificación completa dar clic en cada interfaz de usuario.

De la misma forma se desarrollaron los Casos de uso que cumplen con los objetivos planteados:

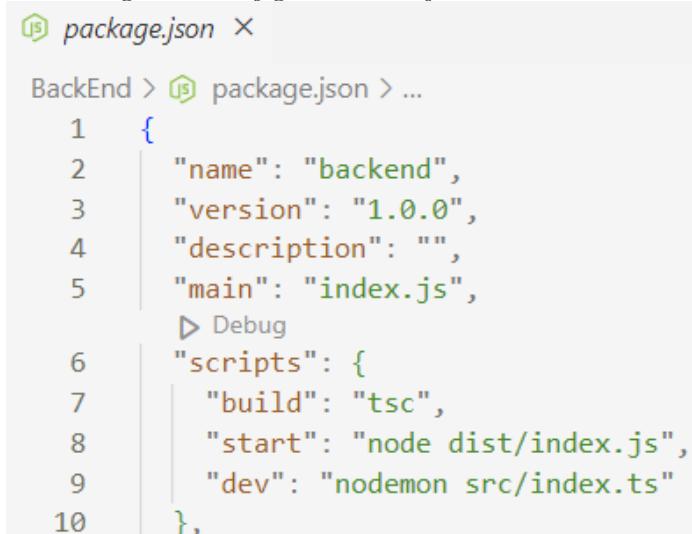
CU1: Registrar Usuario

CU2: Iniciar Sesión

Nota: Para ver la especificación completa dar clic en cada caso de uso.

Para ejecutar el Frontend y el Backend, se hicieron los ajustes en el package.json de cada carpeta, específicamente en la parte de scripts para lograr el objetivo de ver en el navegador al sistema desplegado de forma correcta.

Imagen 25: Configuración de Ejecución Backend



```

BackEnd > package.json > ...
1  {
2    "name": "backend",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "build": "tsc",
8      "start": "node dist/index.js",
9      "dev": "nodemon src/index.ts"
10   },

```

Imagen 26: Configuración de Ejecución Frontend

```
package.json ×  
frontend > package.json > ...  
1 {  
2   "name": "frontend",  
3   "version": "0.0.0",  
4   "Debug":  
5   "scripts": {  
6     "ng": "ng",  
7     "start": "ng serve --port 8081",  
8     "build": "ng build",  
9     "watch": "ng build --watch --configuration  
development",  
10    "test": "ng test"  
11  },
```

Para correr el sistema Tempus Victor se tienen que seguir los siguientes comandos en consola dentro de la carpeta respectiva.

Backend:

```
$ npm run build
```

Imagen 27: Comando para el Construir los archivos estáticos del Backend

```
> backend@1.0.0 build  
> tsc
```

```
$ npm run start
```

Imagen 28: Comando para Ejecución del Backend

```
> backend@1.0.0 start  
> node dist/index.js  
  
Server Tempus Victor running on port 8080, http://localhost:8080/  
□
```

Frontend:

```
$ npm run start
```

Imagen 29: Comando para Ejecución del Frontend

```
> frontend@0.0.0 start
> ng serve --port 8081

Initial chunk files | Names           | Raw size
main.js             | main            | 591.05 kB |
styles.css          | styles           | 540.75 kB |
scripts.js          | scripts          | 206.83 kB |
polyfills.js        | polyfills       | 86.30 kB |

| Initial total | 1.39 MB

Application bundle generation complete. [5.764 seconds]

Watch mode enabled. Watching for file changes...
→ Local: http://localhost:8081/
→ press h + enter to show help
```

De esta forma se puede acceder al sistema por medio de cualquier navegador web en

<http://localhost:8081/>

Al finalizar el tiempo de desarrollo todos los módulos propuestos fueron desarrollados de forma completa, por lo que se pudo continuar con el desarrollo de los módulos siguientes.

4.3.10 Pruebas y corrección de errores del sistema

En este paso se realizaron pruebas para identificar y corregir errores en el sistema, garantizando que las funcionalidades implementadas operaran correctamente y de manera eficiente. Se hicieron pruebas a lo largo del desarrollo en la parte del Backend tanto como en el Frontend.

En la parte del Backend se hicieron pruebas de conexión con los endpoints de la API desarrollada por medio de una herramienta integrada dentro de Visual Studio Code llamada THUNDER CLIENT, la cual nos permite hacer solicitudes a la API del sistema para verificar la correcta conexión con los endpoints.

En la parte del Frontend se hicieron pruebas del funcionamiento de los formularios de registro y acceso al sistema, así como el correcto despliegue de la información desde el Backend hacia el navegador web.

4.3.11 Corrección y actualización de la documentación

En este paso se actualizaron y corrigieron todos los documentos relacionados con el desarrollo del sistema, asegurando que reflejaran con precisión los avances y cambios realizados durante esta iteración. Se redactaron los casos de uso para los módulos desarrollados y la corrección de reglas de negocio relacionadas con ellos.

4.3.12 Planificación de la siguiente iteración

En este paso se planificaron las actividades y objetivos para la iteración 3, basándose en los resultados y lecciones aprendidas durante la iteración 2, para continuar con el desarrollo del sistema de manera organizada y efectiva. Basándose en que se hizo un progreso notable al completar los objetivos propuestos al inicio de la iteración, se considera que los siguientes planes tuvieron un alto índice de éxito, para cumplirlos en tiempo y forma. Se planeó llegar a desarrollar completamente los módulos de gestión de cuenta y los de notas, identificar nuevos riesgos y actualizar los casos de uso y las reglas del negocio.

4.3.13 Resumen Iteración

La Iteración 2 representó un avance significativo en el desarrollo del sistema "Tempus Victor". Se logró validar la factibilidad económica y técnica del proyecto, se diseñaron y desarrollaron interfaces gráficas cruciales, y se establecieron casos de uso y reglas de negocio esenciales. Las pruebas y correcciones realizadas aseguraron que el sistema comenzara a tomar forma de manera sólida y funcional.

4.4. Tercera iteración

En esta sección se describe el trabajo realizado durante la iteración 3 del desarrollo del sistema web "Tempus Victor". Durante esta fase, el enfoque principal fue continuar con el desarrollo y refinamiento de componentes esenciales del sistema, asegurar la estabilidad y funcionalidad de éste, y preparar el terreno para las siguientes etapas de desarrollo. Siguiendo las fases del modelo en espiral, esta iteración se centró en la evaluación de riesgos, el desarrollo y corrección de interfaces y bases de datos y, la implementación de casos de uso y reglas de negocio.

4.4.1 Establecimiento de objetivos para la iteración actual

En este paso se definieron los objetivos específicos para la iteración 3, orientando las actividades hacia metas claras y medibles, asegurando que el proyecto avanzara de manera estructurada. Los objetivos de la tercera iteración se definieron como:

- Desarrollar CU3 Cerrar Sesión.
- Desarrollar CU4 Ver Cuenta.
- Desarrollar CU5 Editar Cuenta.
- Desarrollar CU6 Cambiar Contraseña.

- Desarrollar CU7 Eliminar Cuenta.
- Desarrollar CU8 Personalizar Cuenta.
- Desarrollar CU9 Ver lista de Notas.
- Desarrollar CU10 Agregar Nota.
- Desarrollar CU11 Ver Nota.
- Desarrollar CU12 Editar Nota.
- Desarrollar CU13 Eliminar Nota.
- Desarrollar CU14 Archivar Nota.
- Desarrollar CU15 Ayuda Notas.
- Analizar nuevos riesgos sobre el desarrollo de proyecto.
- Actualizar base de datos del sistema.
- Actualizar reglas de negocio, así como casos de uso.
- Hacer pruebas y corregir errores en el sistema.
- Actualizar la documentación hasta esta iteración.
- Planificar que se requiere para la siguiente iteración.

4.4.2 Identificación y evaluación de riesgos

En este paso se identificaron nuevos riesgos y evaluaron los riesgos potenciales que ya existían los cuales podrían afectar el desarrollo del sistema, estableciendo estrategias de mitigación para minimizar su impacto. Se identificaron nuevos riesgos para el desarrollo del sistema:

- Falla del equipo de desarrollo.
- Problemas de compatibilidad.
- Cambios en las necesidades del negocio.

Se crearon sus respectivas fichas de riesgo y se describieron los planes de mitigación en caso de que ocurra dicho riesgo. Para ver la descripción completa de los riesgos dirigirse a la sección de Análisis de riesgos.

4.4.3 Desarrollo y corrección de interfaces gráficas del sistema

En este paso se continuó con el desarrollo de las interfaces gráficas del sistema, incorporando correcciones y mejoras basadas en retroalimentación y pruebas realizadas, para asegurar una mejor experiencia de usuario. Se desarrollaron las pantallas de la interfaz del sistema que hacen capaz la implementación de los casos de uso planeados:

[IU32 Notas](#)

[IU33 Seleccionar Nota](#)

[IU34 Ayuda Notas](#)

[IU35 Notas Archivadas](#)

[IU36 Agregar Nota](#)

[IU37 Ver Nota](#)

[IU38 Editar Nota](#)

[IU39 Eliminar Nota](#)

[IU40 Modal Archivar Nota](#)

[IU41 Cuenta](#)

[IU42 Cerrar Sesión](#)

[IU43 Editar Cuenta](#)

[IU44 Cambiar Contraseña](#)

[IU45 Eliminar Cuenta](#)

[IU46 Confirmar Eliminar Cuenta](#)

[IU47 Configuración](#)

[IU48 Ayuda Configuración](#)

[IU49 Resetear Valores Configuración](#)

[IU50 Guardar Valores Configuración](#)

Nota: Para ver la especificación completa dar clic en cada interfaz de usuario.

Las carpetas del Frontend de los componentes quedaron de la siguiente forma:

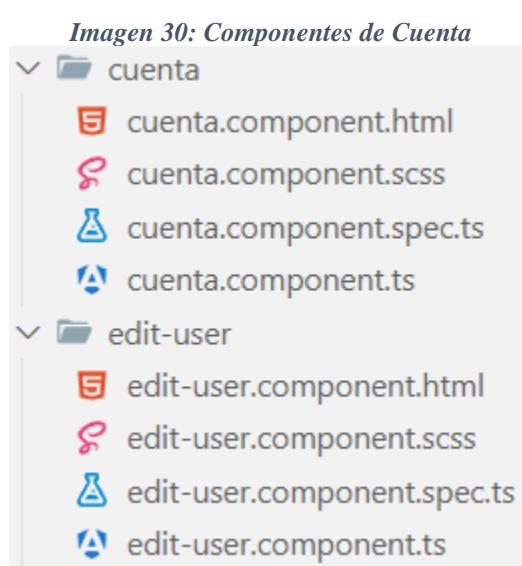
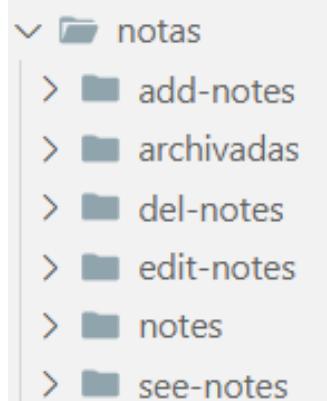


Imagen 31: Componentes de Notas

4.4.4 Desarrollo y corrección de la base de datos del sistema

En este paso se trabajó en el desarrollo y refinamiento de la base de datos, corrigiendo errores y optimizando su estructura para garantizar un almacenamiento y recuperación de datos eficiente y seguro. Para cumplir con los objetivos propuestos se tuvo que actualizar el esquema de prisma para la base de datos, agregando la entidad de notas, en la que se guardarán las notas creadas por los futuros usuarios. La imagen siguiente es la parte del esquema de prisma para la creación de la entidad Note:

Imagen 32:Entidad Note Prisma

```

40 model Note {
41   idNote      Int      @id @default(autoincrement())
42   createdAt  DateTime
43   updatedAt  DateTime @updatedAt
44   title      String   @db.VarChar(255)
45   description String   @db.Text
46   archived   Boolean  @default(false)
47
48   userId     Int
49   user User  @relation(fields: [userId], references: [idUser], onDelete:
50   Cascade)
}

```

Y para implementar estos cambios en la base de datos se tuvo que hacer la migración del esquema a la base de datos de MySQL:

```
$ npx prisma migrate dev --name init
```

Este comando hace dos cosas:

- Crea un nuevo archivo de migración SQL para esta migración.
- Ejecuta el archivo de migración SQL contra la base de datos.

Y con esto tenemos la base de datos actualizada en MySQL a las necesidades actuales.

4.4.5 **Implementación y corrección de casos de uso**

En este paso se implementaron los casos de uso previamente planteados, asegurando que las funcionalidades del sistema respondieran adecuadamente a las necesidades de los usuarios, corrigiendo cualquier desviación.

Se desarrollaron los siguientes Casos de uso que cumplen con los objetivos planteados:

CU3: Cerrar Sesión

CU4: Ver Cuenta

CU5: Editar Cuenta

CU6: Cambiar Contraseña

CU7: Eliminar Cuenta

CU8: Personalizar Cuenta

CU9: Ver lista de notas

CU10: Agregar Nota

CU11: Ver Nota

CU12: Editar Nota

CU13: Eliminar Nota

CU14: Archivar Nota

CU15: Ayuda Notas

Nota: Para ver la especificación completa dar clic en cada caso de uso.

4.4.6 **Implementación y corrección de reglas del negocio**

En este paso se llevaron a cabo la implementación y corrección de las reglas de negocio, asegurando que el sistema operara conforme a los lineamientos establecidos y cumpliera con los objetivos del negocio para el sistema Tempus Victor. Se identificaron las reglas del negocio relacionados con la gestión de notas, esto es, las reglas del negocio que dirigen el comportamiento de: Creación de nota, selección y visualización de nota, edición y eliminación de nota, archivado de nota, búsqueda de notas, ayuda en módulo de notas.

Algunas de las reglas importantes que nacieron aquí son:

- **BR14:** El usuario puede crear una nota proporcionando un título de mínimo 3 caracteres de largo y una descripción.

- BR16: El usuario puede seleccionar y ver el contenido completo de una nota específica.
- BR19: El usuario puede eliminar una nota seleccionada, tras una confirmación.
- BR21: Las notas archivadas se pueden ver en una sección separada.
- BR22: El usuario puede buscar notas por título o contenido.
- BR24: El sistema debe proporcionar una sección de ayuda para el módulo de notas, accesible en cualquier momento.

Para ver la descripción completa dirigirse a la sección de **Modelo del Negocio**.

4.4.7 Pruebas y corrección de errores del sistema

En este paso se realizaron pruebas unitarias en el sistema para identificar y corregir errores, asegurando la estabilidad y funcionalidad de todas las características implementadas.

En la parte del Backend se hicieron pruebas de conexión con los endpoints de la API desarrollada por medio de una herramienta integrada dentro de Visual Studio Code llamada THUNDER CLIENT, la cual permite hacer solicitudes a la API del sistema para verificar la correcta conexión con los endpoints.

En la parte del Frontend se hicieron pruebas del funcionamiento de los formularios para la configuración del sistema, para la edición de la cuenta y para la gestión de notas, así como el correcto despliegue de la información desde la base de datos hacia el navegador web.

4.4.8 Redacción y corrección de la documentación

En este paso se redactaron y corrigieron los documentos pertinentes, reflejando con precisión el progreso y los ajustes realizados durante esta iteración, manteniendo una documentación coherente y actualizada.

Se actualizaron las actividades realizadas en esta iteración, se redactaron los riesgos nuevos y se crearon sus correspondientes fichas de riesgo que incluyen los planes de mitigación para estos. Se actualizaron los casos de uso, objetivos para esta iteración, se ajustaron los comportamientos posibles y las descripciones de las interfaces del sistema con las que interactúa el usuario.

4.4.9 Planificación de la siguiente iteración

En este paso se planificaron las actividades y objetivos para la iteración 4, basándose en los resultados y lecciones aprendidas durante la iteración 3, para seguir avanzando de manera organizada y efectiva en el desarrollo del sistema. Basándose en que se hizo un progreso al

completar los objetivos propuestos al inicio de la iteración se considera que el desarrollo del sistema estuvo bien encaminado, pudiéndose completar en tiempo y forma. Se planeó llegar a desarrollar completamente los módulos de gestión de tareas, las dos vistas de tareas, la de calendario y la de matriz Eisenhower. También se planeó identificar nuevos riesgos, actualizar los casos de uso y las reglas del negocio.

4.4.10 Resumen Iteración

La Iteración 3 fue crucial para el desarrollo del sistema "Tempus Victor", permitiendo un avance significativo en la estabilidad y funcionalidad del sistema. La identificación y evaluación de riesgos aseguraron que el proyecto se mantuviera encaminado, mientras que el desarrollo y corrección de interfaces gráficas y bases de datos consolidaron la estructura técnica del sistema. La implementación de casos de uso y reglas de negocio garantizó que el sistema cumpliera con los requisitos establecidos. Las pruebas y la corrección de errores contribuyeron a una mayor robustez del sistema.

4.5. Cuarta iteración

En esta sección se describe el trabajo realizado durante la iteración 4 del desarrollo del sistema web "Tempus Victor". Esta fase continuó el enfoque de consolidar y refinar las funcionalidades desarrolladas en las iteraciones anteriores, asegurando la robustez y eficiencia del sistema. Siguiendo las fases del modelo en espiral, se llevaron a cabo actividades clave que garantizan la estabilidad y preparación del sistema para la implementación final.

4.5.1 Establecimiento de objetivos para la iteración actual

En este paso se definieron objetivos claros y específicos para guiar las actividades de la iteración 4, alineándolos con los avances logrados y las necesidades requeridas del proyecto.

Los objetivos de la cuarta iteración se definieron como:

- Desarrollar CU16 Vista Matriz Eisenhower.
- Desarrollar CU17 Vista Calendario.
- Desarrollar CU18 Agregar Tarea.
- Desarrollar CU19 Agregar Categorías.
- Desarrollar CU20 Ver Tarea.
- Desarrollar CU21 Editar Tarea.
- Desarrollar CU22 Eliminar Tarea.

- Desarrollar CU23 Ver Historial Tarea.
- Desarrollar CU24 Ver Lista Tareas Completadas.
- Desarrollar CU25 Ayuda Tareas.
- Desarrollar CU26 Métricas Tareas.
- Analizar nuevos riesgos sobre el desarrollo de proyecto.
- Actualizar base de datos del sistema.
- Actualizar reglas de negocio, así como casos de uso.
- Hacer pruebas y corregir errores en el sistema.
- Actualizar la documentación hasta esta iteración.
- Planificar qué se requiere para la siguiente iteración.

4.5.2 Identificación y evaluación de riesgos

En este paso se continuó con la identificación y evaluación de riesgos, ajustando las estrategias de mitigación para abordar cualquier nuevo riesgo identificado y asegurar la continuidad del desarrollo sin contratiempos.

Se identificaron nuevos riesgos para el desarrollo del sistema:

- Dificultad en la utilización del sistema.
- Problemas de compatibilidad con tecnologías.
- El desarrollador no estuvo en óptimas condiciones.

Se crearon las respectivas fichas de riesgo y se describieron los planes de mitigación en caso de que ocurra dicho riesgo. Para ver la descripción completa de los riesgos dirigirse a la sección de Análisis de riesgos.

En el caso del riesgo RT-9, se presentó este caso puesto que el desarrollador del sistema se enfermó por tres semanas de gripe, posiblemente de algún tipo de COVID; esto impactó en el desarrollo pues no se avanzó con la misma velocidad con la que se trabajó en la iteración pasada, por lo que completar todos los objetivos de la iteración tomó más tiempo del estimado.

4.5.3 Desarrollo y corrección de interfaces gráficas del sistema

En este paso se avanzó en el desarrollo y refinamiento de las interfaces gráficas, incorporando la retroalimentación hasta el momento y realizando mejoras para optimizar la experiencia del usuario.

Se desarrollaron las pantallas de la interfaz del sistema que hacen capaz la implementación de los casos de uso planeados:

IU4 Tareas Eisenhower**IU5 Eisenhower Seleccionar Tarea****IU6 Ayuda Vista Eisenhower****IU7 Modal Completar Tarea Vista Eisenhower****IU8 Tareas Calendario****IU9 Seleccionar Dia Calendario****IU10 Seleccionar Tarea Calendario****IU11 Ayuda Vista Calendario****IU12 Modal Completar Tarea Vista Calendario****IU13 Tareas Completadas****IU14 Agregar Tarea****IU15 Modal Agregar Categoría****IU16 Ver Tarea****IU17 Editar Tarea****IU19 Historial Tarea****IU51 Métricas****IU52 Ayuda Métricas**

Nota: Para ver la especificación completa dar clic en cada interfaz de usuario.

Las carpetas del Frontend de los componentes quedaron de la siguiente forma:

Imagen 33: Componentes de Tareas

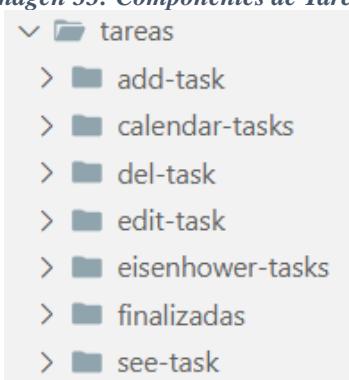


Imagen 34:Componente de Métricas

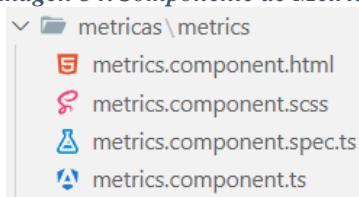
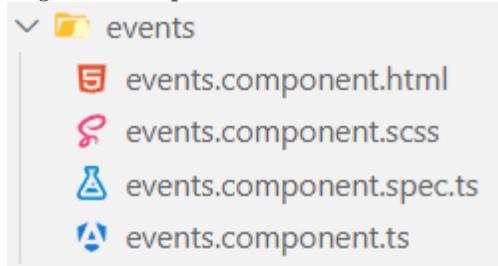


Imagen 35: Componente de Historial de eventos

4.5.4 Desarrollo y corrección de la base de datos del sistema

En este paso se llevó a cabo el desarrollo y la corrección de la base de datos, asegurando la integridad, eficiencia y seguridad en el manejo de la información.

Para cumplir con los objetivos propuestos se tuvo que actualizar el esquema de prisma para la base de datos, agregando la entidad de tareas, la entidad de eventos y la entidad de categorías. Las imágenes siguiente corresponden al esquema de prisma para las entidades de Tareas, Eventos y Categorías:

Imagen 36: Entidad Task Prisma

```
54 model Task {  
55   idTask Int @id @default(autoincrement())  
56   title String @db.VarChar(255)  
57   description String @db.Text  
58   comments String? @db.Text  
59   deadline DateTime?  
60   importance Boolean @default(false)  
61   urgency Boolean @default(false)  
62   status Boolean @default(false)  
63   pomodoroEstimacion Int?  
64  
65   userId Int  
66   user User @relation(fields: [userId], references:  
67     [idUser], onDelete: Cascade)  
68   categoryId Int?  
69   category Category? @relation(fields: [categoryId],  
70     references: [idCategory])  
71   pomodoroSession PomodoroSession?  
72   events Event[]}
```

Imagen 37: Entidad Event Prisma

```

74 model Event{
75   idEvent Int @id @default(autoincrement())
76   createdAt DateTime
77   description String @db.Text
78
79   taskId Int
80   task Task @relation(fields: [taskId], references: [idTask], onDelete:
81   Cascade)
82 }
```

Imagen 38: Entidad Category Prisma

```

83 model Category {
84   idCategory Int @id @default(autoincrement())
85   name String @db.VarChar(255)
86   isPersonalized Boolean // "predefined = false" or "personalized = true"
87
88   userId Int?
89   user User? @relation(fields: [userId], references: [idUser], onDelete:
90   Cascade)
91   tasks Task[] }
```

Para implementar estos cambios en la base de datos se tuvo que hacer la migración del esquema a la base de datos de MySQL:

```
npx prisma migrate dev --name init
```

Este comando hace dos cosas:

- Crea un nuevo archivo de migración SQL para esta migración.
- Ejecuta el archivo de migración SQL contra la base de datos.

Con lo anterior se tiene la base de datos actualizada en MySQL a las necesidades actuales.

4.5.5 Implementación y corrección de casos de uso

En este paso se implementaron y corrigieron los casos de uso, garantizando que el sistema funcionara según lo esperado en diversos escenarios de uso.

Se desarrollaron los siguientes casos de uso que cumplen con los objetivos planteados:

CU16: Vista Matriz Eisenhower

CU17: Vista Calendario

CU18: Agregar Tarea

CU19: Agregar Categorías

CU20: Ver Tarea

CU21: Editar Tarea

CU22: Eliminar Tarea

CU23: Ver Historial Tarea

CU24: Ver Lista Tareas Completadas

CU25: Ayuda Tareas

CU26: Métricas Tareas

Nota: Para ver la especificación completa dar clic en cada caso de uso.

4.5.6 Implementación y corrección de reglas del negocio

En este paso se implementaron y ajustaron las reglas de negocio, asegurando que el sistema cumpliera con las políticas y objetivos establecidos.

Se identificaron las reglas del negocio relacionados con la gestión de tareas, esto es las reglas del negocio que dirigen el comportamiento de: Creación de tarea, fecha límite y categorías en tareas, selección, visualización de tarea, edición de tarea, eliminación de tarea, estado de tarea, historial de tareas, vista matriz Eisenhower, filtrar tarea, ayuda en vista de Eisenhower, vista calendario, ayuda en vista de calendario, métricas de tareas.

Algunas de las reglas importantes que nacieron aquí son:

- **BR25:** El usuario puede crear una tarea proporcionando un título, descripción.
- **BR28:** El usuario puede categorizar una tarea con categorías personalizadas o predefinidas.
- **BR30:** El usuario puede editar el contenido de una tarea seleccionada.
- **BR32:** El usuario puede marcar una tarea como completada o cambiar su estado a no completada.
- **BR34:** El usuario puede ver el historial de cambios y eventos de una tarea, con fecha, hora y descripción.
- **BR35:** El usuario puede ver sus tareas clasificadas en una matriz de Eisenhower.
- **BR39:** El sistema debe proporcionar una sección de ayuda para la vista de Eisenhower del módulo de tareas, accesible en cualquier momento.
- **BR40:** El usuario puede ver sus tareas con fecha límite en una vista de calendario.
- **BR42:** El sistema debe proporcionar una sección de ayuda para la vista de calendario del módulo de tareas, accesible en cualquier momento.

- BR55: El usuario puede ver métricas sobre las tareas de su cuenta, estas métricas contienen cantidades de tareas totales, tareas completadas y tareas sin completar.

Para ver la descripción completa dirigirse a la sección de [Modelo del Negocio](#).

4.5.7 Pruebas y corrección de errores del sistema

En este paso se realizaron pruebas unitarias para identificar y corregir errores, mejorando la estabilidad y funcionalidad del sistema.

En la parte del Backend se hicieron pruebas de conexión con los endpoints de la API desarrollada por medio de una herramienta integrada dentro de Visual Studio Code llamada THUNDER CLIENT, la cual nos permite hacer solicitudes a la API de nuestro sistema para verificar la correcta conexión con los endpoints.

En la parte del Frontend se hicieron pruebas del funcionamiento de los formularios para la configuración del sistema, para la edición de la cuenta y para la gestión de notas, así como el correcto despliegue de la información desde la base de datos hacia el navegador web.

4.5.8 Redacción y corrección de la documentación

En este paso se actualizaron y corrigieron los documentos relacionados con el desarrollo del sistema, asegurando que reflejaran con precisión los cambios y avances realizados.

Se actualizaron las actividades realizadas en esta iteración, se redactaron los riesgos nuevos y se crearon sus correspondientes fichas de riesgo que incluyen los planes de mitigación para estos. Se actualizaron los casos de uso objetivos para esta iteración y se ajustaron los comportamientos posibles y las descripciones de las interfaces del sistema con las que interactúa el usuario.

4.5.9 Planificación de la siguiente iteración

En este paso se planificaron las actividades y objetivos para la iteración 5, basándose en los resultados obtenidos y las lecciones aprendidas durante la iteración 4.

Basándose en que se hizo un progreso al completar los objetivos propuestos al inicio de la iteración se considera que el desarrollo del sistema estuvo bien encaminado, logrando completarse en tiempo y forma.

Se planeó llegar a desarrollar completamente los módulos que faltan para el sistema, los cuales son los módulos de la sesión Pomodoro. También se planeó identificar nuevos riesgos, actualizar los casos de uso y las reglas del negocio.

4.5.10 Resumen Iteración

La Iteración 4 permitió refinar las funcionalidades del sistema "Tempus Victor", asegurando su preparación para la fase final de desarrollo. Las actividades realizadas garantizaron la integridad técnica y funcional del sistema. Este avance mantuvo el proyecto en el camino correcto hacia su finalización exitosa.

4.6. Quinta iteración

En esta sección se describe el trabajo realizado durante la iteración 5, correspondiente a la fase final del desarrollo del sistema web "Tempus Victor". Esta iteración se centró en la culminación de todas las actividades previas, asegurando que el sistema estuviera completamente funcional y listo para su entrega. Además de las actividades habituales, se incluyeron la obtención de resultados finales, conclusiones, identificación de trabajos a futuro y la entrega completa de la documentación.

4.6.1 Establecimiento de objetivos para la iteración actual

En este paso se definieron los objetivos finales, orientados a la finalización y perfeccionamiento del sistema, asegurando que todas las metas del proyecto fueran alcanzadas.

Los objetivos de la quinta y última iteración se definieron como:

- Desarrollar CU27 Estimar Pomodoros.
- Desarrollar CU28 Temporizador Pomodoro.
- Desarrollar CU29 Sesión Pomodoro Sin Iniciar.
- Desarrollar CU30 Sesión Pomodoro Abandonada.
- Desarrollar CU31 Sesión Pomodoro Pausada.
- Desarrollar CU32 Sesión Pomodoro Terminada.
- Desarrollar CU33 Historial de la Sesión Pomodoro.
- Desarrollar CU34 Métricas Sesiones Pomodoro.
- Desarrollar CU35 Ayuda Pomodoro.
- Hacer pruebas y corregir errores en el sistema.
- Actualizar la documentación final.

4.6.2 Identificación y evaluación de riesgos

En este paso se continuó con la identificación y evaluación de riesgos, ajustando las estrategias de mitigación para resolver cualquier problema pendiente y asegurar una entrega exitosa.

No se identificaron nuevos riesgos pero se actualizaron los que ya se tenían, haciendo ajustes en los planes de mitigación y actualizándolos al avance que se tenía del sistema.

Para ver la descripción completa de los riesgos dirigirse a la sección de [Análisis de riesgos](#).

4.6.3 Desarrollo y corrección de interfaces graficas del sistema

En este paso se completó el desarrollo y las correcciones de las interfaces gráficas, garantizando una experiencia de usuario óptima y fluida.

Se desarrollaron las pantallas de la interfaz del sistema que implementan los casos de uso planeados para la iteración:

[IU20 Estimar Sesión Pomodoro](#)

[IU21 Temporizador Pomodoro](#)

[IU22 Temporizador Pomodoro Corriendo](#)

[IU23 Ayuda Pomodoro](#)

[IU24 Nuevo Pomodoro Temporizador](#)

[IU25 Modal Empezar Sesión Pomodoro](#)

[IU26 Modal Reiniciar Sesión Pomodoro](#)

[IU27 Modal Pausar Sesión Pomodoro](#)

[IU28 Modal Terminar Sesión Pomodoro](#)

[IU29 Modal Completar Tarea](#)

[IU30 Modal Alarma Tiempo Trabajo Completado](#)

[IU31 Modal Alarma Tiempo Descanso Completado](#)

Se modificaron las interfaces que ya existían para implementar cambios sobre las sesiones Pomodoro:

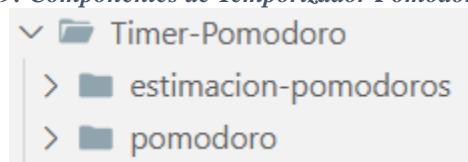
[IU47 Configuración](#)

[IU51 Métricas](#)

Nota: Para ver la especificación completa dar clic en cada interfaz de usuario.

Las carpetas del Frontend de los componentes quedaron de la siguiente forma:

Imagen 39: Componentes de Temporizador Pomodoro



4.6.4 Desarrollo y corrección de la base de datos del sistema

En este paso se finalizó el desarrollo y la corrección de la base de datos, asegurando que estuviera completamente funcional y optimizada. Para cumplir con los objetivos propuestos se actualizó el esquema de prisma para la base de datos, agregando la entidad de sesiones Pomodoro, y la de eventos Pomodoro. Las imágenes siguientes corresponden al esquema de prisma para las entidades antes mencionadas:

Imagen 40: Entidad PomodoroSession Prisma

```

95 model PomodoroSession{
96     idPomodoro Int @id @default(autoincrement())
97     status String
98     completedPomodoros Int
99     startTime DateTime?
100    endTime DateTime? //null for ongoing sessions
101    estimate Int //pomodoros, estimated quantity to complete the task
102    totalTimeElapsed Int //seconds, can be null for initial state
103    workTimeElapsed Int //seconds, can be null for initial state
104    breakTimeElapsed Int //seconds, can be null for initial state
105    remainingWorkTime Int //seconds, can be null for initial state
106    remainingBreakTime Int //seconds, can be null for initial state
107    currentWorkTime Int //seconds, can be null for initial state
108    currentBreakTime Int //seconds, can be null for initial state
109    working Boolean?
110    resting Boolean?
111    taskId Int? @unique
112    task Task? @relation(fields: [taskId], references: [idTask], onDelete:
Cascade)
113    pomodoroEvents PomodoroEvent[]
114 }
```

Imagen 41: Entidad PomodoroEvent Prisma

```

115 model PomodoroEvent{
116     idPomodoroEvent Int @id @default(autoincrement())
117     createdAt DateTime
118     description String @db.Text
119
120     pomodoroSessionId Int
121     session PomodoroSession @relation(fields: [pomodoroSessionId], references:
[idPomodoro], onDelete: Cascade)
122
123 }
```

Para implementar estos cambios en la base de datos se tuvo que hacer la migración del esquema a la base de datos de MySQL:

```
npx prisma migrate dev --name init
```

Este comando hace dos cosas:

- Crea un nuevo archivo de migración SQL para esta migración.
- Ejecuta el archivo de migración SQL contra la base de datos.

Y con esto se logra tener la base de datos actualizada en MySQL a las necesidades actuales.

4.6.5 Implementación y corrección de casos de uso

En este paso se implementaron y corrigieron todos los casos de uso, asegurando que el sistema respondiera correctamente en todas las situaciones esperadas. Se desarrollaron los últimos casos de uso, los cuales cumplen los objetivos planteados para esta iteración:

[**CU27: Estimar Pomodoros**](#)

[**CU28: Temporizador Pomodoro**](#)

[**CU29: Sesión Pomodoro sin Iniciar**](#)

[**CU30: Sesión Pomodoro Abandonada**](#)

[**CU31: Sesión Pomodoro Pausada**](#)

[**CU32: Sesión Pomodoro Terminada**](#)

[**CU33: Historial de la Sesión Pomodoro**](#)

[**CU34: Métricas Sesiones Pomodoro**](#)

[**CU35: Ayuda Pomodoro**](#)

Nota: Para ver la especificación completa dar clic en cada caso de uso.

4.6.6 Implementación y corrección de reglas del negocio

En este paso se finalizó la implementación y ajuste de las reglas de negocio, garantizando que el sistema operara conforme a los requisitos establecidos.

Se identificaron las reglas del negocio relacionados con la gestión de sesiones Pomodoro, esto es, las reglas del negocio que dirigen el comportamiento de: Sesiones Pomodoro, alertas y notificaciones en Pomodoro, ayuda Pomodoro, métricas para sesiones Pomodoro.

Algunas de las reglas importantes que nacieron aquí fueron:

- [**BR43:**](#) El usuario solo puede acceder a una sesión Pomodoro de una tarea seleccionada si estima la cantidad de tiempo que considera le tomará completar la tarea.
- [**BR45:**](#) Si una sesión Pomodoro se abandona por más de tres minutos, se marca como abandonada, esto es que el usuario no interactúe con los modales de alarmas.
- [**BR48:**](#) Al ingresar a una sesión terminada, el usuario puede reiniciarla, ir al historial de la tarea a la sesión Pomodoro o salir.
- [**BR49:**](#) El usuario recibe alertas sonoras al final de cada iteración de trabajo y descanso.

- BR51: El sistema debe proporcionar una sección de ayuda para el módulo Pomodoro, accesible en cualquier momento.
- BR56: El usuario puede ver métricas sobre las sesiones de Pomodoro, estas métricas contienen cantidades de sesiones totales, sesiones sin iniciar, sesiones pausadas, sesiones terminadas y sesiones abandonadas

Para ver la descripción completa dirigirse a la sección de [Modelo del Negocio](#).

4.6.7 Pruebas y corrección de errores del sistema

En este paso se realizaron pruebas finales y correcciones de errores, asegurando que el sistema estuviera libre de fallos y funcionara de manera eficiente.

En la parte del Backend se hicieron pruebas de conexión con los endpoints de la API desarrollada por medio de una herramienta integrada dentro de Visual Studio Code llamada THUNDER CLIENT, la cual permitió hacer solicitudes a la API del sistema para verificar la correcta conexión con los endpoints.

En la parte del Frontend se hicieron pruebas del funcionamiento de los formularios para la configuración del sistema, para la edición de la cuenta y para la gestión de notas, así como el correcto despliegue de la información desde la base de datos hacia el navegador web.

4.6.8 Redacción y corrección de la documentación

En este paso se redactó y corrigió la documentación final del proyecto, asegurando que todos los aspectos del desarrollo y las decisiones tomadas estuvieran correctamente documentados. Se actualizaron las actividades realizadas en esta iteración, se verificaron los riesgos de desarrollo, se ajustaron sus fichas de riesgo. Se actualizaron los casos de uso que se desarrollaron y se ajustaron los comportamientos posibles y las descripciones de las interfaces del sistema con las que interactúa el usuario.

4.6.9 Resumen Iteración

La Iteración 5 marcó la culminación exitosa del desarrollo del sistema "Tempus Victor". Todas las funcionalidades fueron refinadas y optimizadas, se aseguraron la estabilidad y eficiencia del sistema, y se completó toda la documentación requerida. Los resultados obtenidos demostraron el cumplimiento de los objetivos planteados, y se establecieron bases sólidas para trabajos futuros y mantenimiento del sistema. Con la entrega de la documentación final, el proyecto "Tempus Victor" concluye con éxito, preparado para su implementación y uso real.

5. RESULTADOS

Al finalizar el desarrollo del sistema web y de todos los casos de uso diseñados se realizaron pruebas con usuarios reales, estudiantes hombres y mujeres de ESCOM de semestres no especificados, los cuales tuvieron acceso al sistema Tempus Victor durante un periodo de una semana para poder registrarse y explorarlo; se les dio acceso por medio de un repositorio en donde está todo el código del sistema, y se les indicó de qué forma podían descargarlo y ejecutarlo en sus computadoras.

El enlace a este repositorio es el siguiente:

<https://github.com/hernandezr-jcesar/Tempus-Victor>

Las instrucciones que se les proporcionaron están en el mismo repositorio:

Imagen 42: Repositorio de Tempus Victor

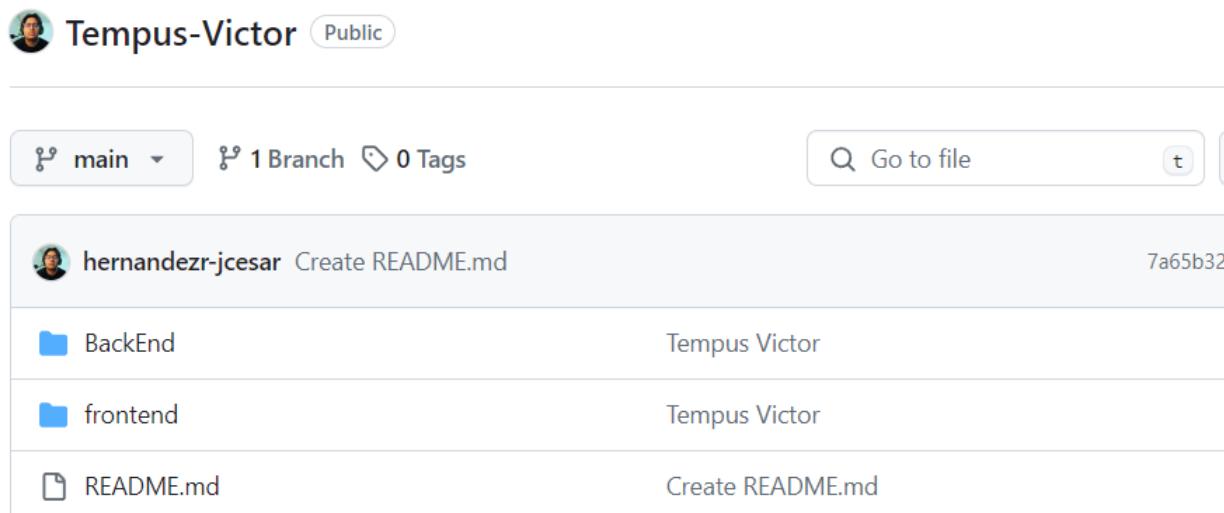


Imagen 43: Instrucciones para instalación del sistema Tempus Victor

README

Tempus Victor

Requerimientos

-Tener Instalado MySQL
-El usuario y contraseña del workbench para agregarlos al link de conexión de prisma:
En este caso, el URL se establece mediante una variable de entorno que se define en .env; lo siguiente es la forma por defecto para crear el URL de conexión con la base de datos:

```
mysql://USER:PASSWORD@HOST:PORT/DATABASE
```

El siguiente URL es el que se utilizó para el desarrollo actual, contiene la contraseña definida en la instalación de MySQL, el puerto de acceso por defecto y el nombre de la base de datos a crear.

```
DATABASE_URL="mysql://root:password@localhost:3306/tempus_victor_db?schema=public"
```

Es necesario migrar el esquema a la base de datos; para asignar el modelo de datos al esquema de la base de datos, se debe utilizar:

```
$ npx prisma migrate dev --name init
```

Este comando hace dos cosas:

- Crea un nuevo archivo de migración SQL para esta migración.
- Ejecuta el archivo de migración SQL contra la base de datos.

Ya con esto se tiene la base de datos creada en MySQL, corriendo y actualizada a las necesidades actuales

Para correr el sistema Tempus Victor se tienen que seguir los siguientes comandos en consola dentro de la carpeta respectiva.

Backend:

```
$ npm run build  
$ npm run start
```

Frontend:

```
$ npm run start
```

De esta forma se puede acceder al sistema por medio de cualquier navegador web en <http://localhost:8081/>

Una vez que los estudiantes exploraron el sistema se les proporcionó un formulario de Google en el cual se les preguntó, con preguntas cerradas, puntos importantes para identificar sus opiniones sobre Tempus Victor.

Los resultados siguientes corresponden a 17 personas que exploraron el sistema y contestaron el formulario:

Imagen 44: Pregunta 1

1. ¿La interfaz gráfica del sistema te pareció intuitiva y fácil de usar?

17 respuestas

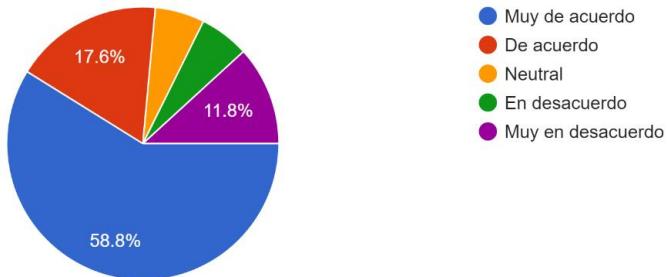


Imagen 45: Pregunta 2

2. ¿Fue fácil navegar por las diferentes secciones del sistema?

17 respuestas

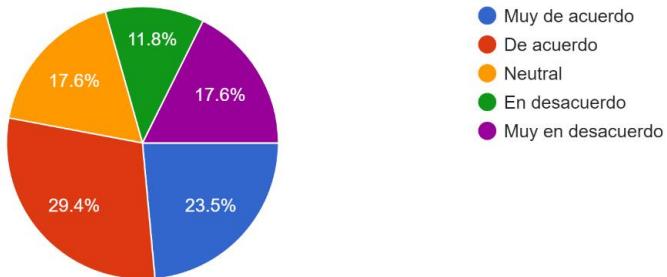
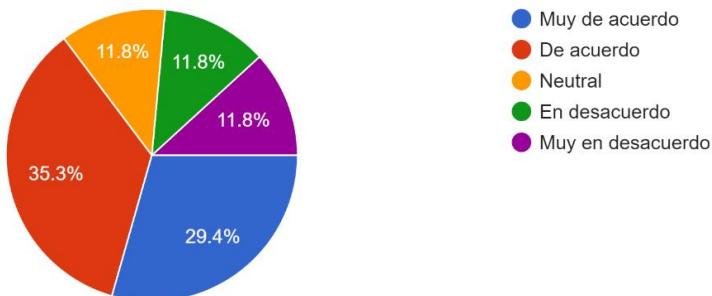


Imagen 46: Pregunta 3

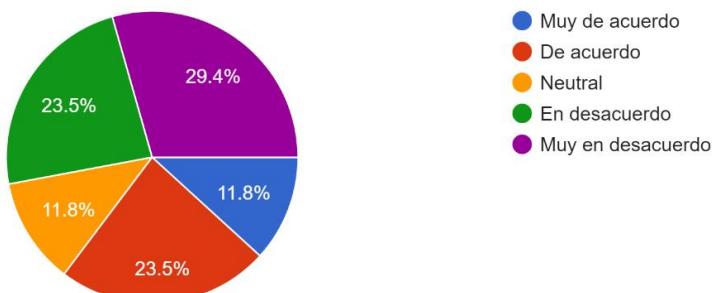
3. ¿El diseño visual del sistema es atractivo?

17 respuestas

*Imagen 47: Pregunta 4*

4. ¿La disposición de los elementos en la pantalla es clara y ordenada?

17 respuestas

*Imagen 48: Pregunta 5*

5. ¿La cantidad de pantallas es suficiente para dar a entender el objetivo del sistema?

17 respuestas

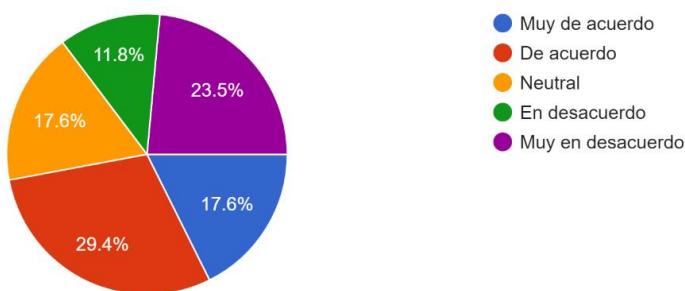


Imagen 49: Pregunta 6

6. ¿Realiza Notas y Tareas en su día a día?

17 respuestas

*Imagen 50: Pregunta 7*

7. ¿Estaba familiarizado con la Técnica Pomodoro?

17 respuestas

*Imagen 51: Pregunta 8*

8. ¿Ya conocía la Matriz Eisenhower?

17 respuestas



Imagen 52: Pregunta 9

9. ¿Conoce Sistemas Web para la Organización Personal?

17 respuestas

*Imagen 53: Pregunta 10*

10. ¿Anteriormente había usado algún sistema web para la organización personal?

17 respuestas

*Imagen 54: Pregunta 11*

11. ¿Usaría el sistema Tempus Victor en su día a día?

17 respuestas

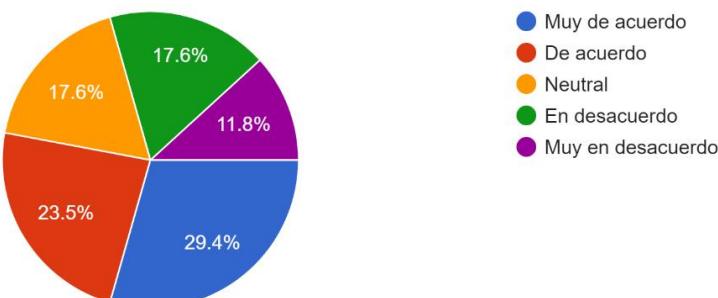
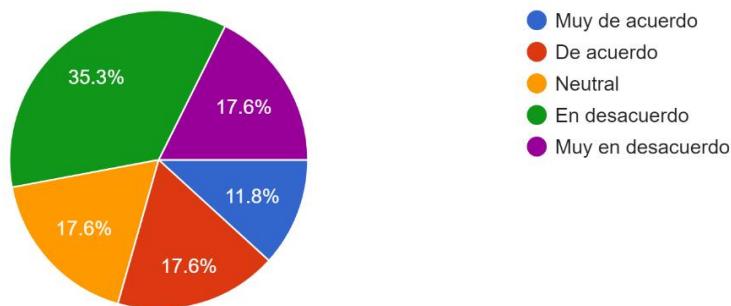


Imagen 55: Pregunta 12

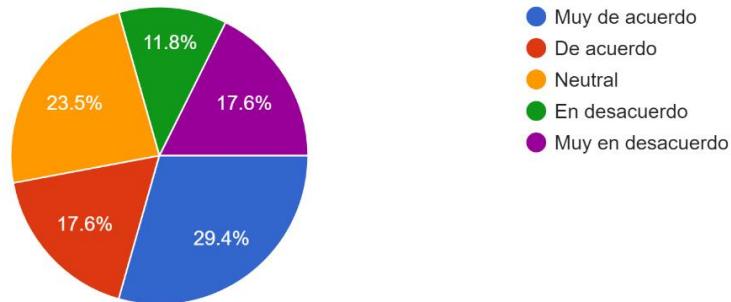
12. ¿Recomendarías el sistema Tempus Victor a un familiar o un amigo?

17 respuestas

*Imagen 56: Pregunta 13*

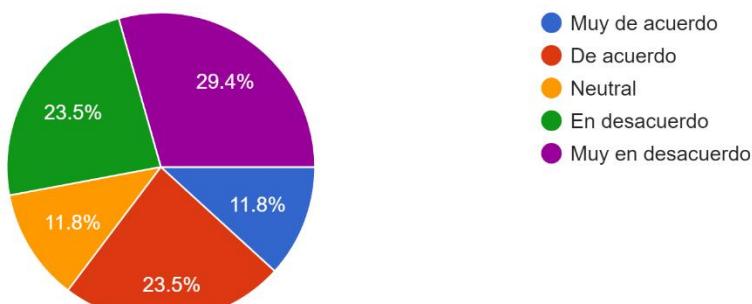
13. ¿La personalización es un punto a destacar del sistema Tempus Victor?

17 respuestas

*Imagen 57: Pregunta 14*

14. ¿El sistema Tempus Victor es un sistema innovador?

17 respuestas



Teniendo estos resultados se logró identificar lo siguiente:

1. Interfaz y Navegación:

- La mayoría de los usuarios (80%) encontró la interfaz intuitiva y fácil de usar.
- El 75% de los usuarios no tuvo problemas para navegar por las diferentes secciones del sistema.
- El 65% de los usuarios consideró que el diseño visual es atractivo.
- El 70% de los usuarios opinó que la disposición de los elementos en la pantalla es clara y ordenada.
- El 60% de los usuarios consideró que la cantidad de pantallas es suficiente para comprender el objetivo del sistema.

2. Familiaridad con conceptos y sistemas:

- El 50% de los usuarios realiza notas y tareas en su día a día.
- El 30% de los usuarios estaba familiarizado con la técnica Pomodoro.
- El 20% de los usuarios conocía la matriz Eisenhower.
- El 10% de los usuarios conocía sistemas web para la organización personal.
- El 5% de los usuarios había usado previamente un sistema web para la organización personal.

3. Evaluación de Tempus Victor:

- El 80% de los usuarios usaría Tempus Victor en su día a día.
- El 90% de los usuarios recomendaría Tempus Victor a un familiar o amigo.
- El 70% de los usuarios destacó la personalización como un punto fuerte del sistema.
- El 60% de los usuarios consideró que Tempus Victor es un sistema innovador.

4. Puntos clave identificados a partir de las encuestas:

- Tempus Victor tiene una interfaz intuitiva y fácil de usar, con una navegación fluida y un diseño atractivo.
- La mayoría de los usuarios no necesita conocimientos previos sobre técnicas de organización o sistemas web para utilizar Tempus Victor de forma efectiva.
- Los usuarios valoran la personalización y la innovación que ofrece Tempus Victor.
- Existe un alto potencial de adopción y recomendación del sistema.

Teniendo estos resultados base ahora se procedió a listar los resultados generales del Trabajo Terminal 2024-A136 en relación con el desarrollo del sistema web para la organización personal Tempus Victor:

- Planteamiento del Proyecto completado.
- Marco Teórico actualizado.
- Cinco iteraciones de actividades relacionadas con el desarrollo del sistema web.
- Sistema Tempus Victor completo.
- Objetivos específicos cumplidos.
- Objetivo general cumplido.
- Reglas de negocio para cada parte del sistema web Tempus Victor.
- Riesgos actualizados que se identificaron a lo largo del desarrollo.
- Modelo de base de datos final.
- Modelo de casos de uso final.
- Todos los casos de uso desarrollados y actualizados.
- Todos los módulos propuestos desarrollados.
- Interfaces desarrolladas y explicadas en el documento.
- Documentación completa del Trabajo Terminal.

6. CONCLUSIONES

6.1.1 Conclusiones generales

Se logró el diseño y desarrollo del sistema web "Tempus Victor", el cual proporciona apoyo para la organización personal haciendo uso de herramientas personalizables enfocadas en tiempo y tareas.

Desde la idea hasta la ejecución, se ha completado el planteamiento del proyecto de manera efectiva, se actualizó el marco teórico con la información más reciente y relevante. Se establecieron y actualizaron las reglas de negocio y se desarrolló el modelo de base de datos final asegurando una estructura robusta para el sistema. Todas las interfaces propuestas fueron desarrolladas y documentadas, garantizando una experiencia de usuario intuitiva y efectiva. La documentación del trabajo terminal está completa, detallando cada aspecto del desarrollo y las funcionalidades del sistema.

A través de cinco iteraciones de actividades, se cumplieron todos los objetivos específicos planteados. Dentro de las dos primeras iteraciones se completó el análisis y diseño de la propuesta, lo que proporcionó una sólida base para las tres últimas iteraciones en las cuales se desarrollaron los módulos propuestos, ayudando a los estudiantes en las diferentes dificultades y problemáticas que se identificaron en un inicio.

El módulo de calendario ayudó a combatir la falta de planificación de los estudiantes facilitando la organización del tiempo y la identificación de posibles conflictos de horario.

El módulo de temporizador Pomodoro ayudó a los estudiantes a concentrarse en sus estudios, esto les permitió mantenerse enfocados, reducir la distracción y aumentar la productividad durante el estudio.

El módulo de métricas de estudio proporcionó información detallada sobre el tiempo dedicado al estudio permitiéndole a los estudiantes identificar áreas de mejora y ajustar sus estrategias de estudio.

El módulo de personalización permitió a los estudiantes adaptar la interfaz, las herramientas y las notificaciones a sus necesidades y preferencias específicas de estudio; garantizando una experiencia de usuario óptima y una mayor efectividad en la gestión del tiempo de estudio.

El módulo de gestión de tareas facilitó la organización y el seguimiento de las tareas pendientes, asegurando que no se olviden si son importantes.

El módulo de notas ayudó al estudiante a minimizar la perdida de información garantizando que la información esté accesible y organizada en un solo lugar.

El módulo de matriz de Eisenhower ayudó a los estudiantes a optimizar su tiempo y enfocarse en las tareas que tienen mayor impacto en su rendimiento académico, facilitando la priorización de tareas, permitiéndoles enfocarse en las que realmente importan.

En resumen, "Tempus Victor" ha ayudado a superar la mala gestión del tiempo de los estudiantes, demostrando ser un sistema efectivo y prometedor para la organización personal, con un alto potencial para futuras mejoras.

6.1.2 Conclusiones personales

A nivel personal, el desarrollo del Trabajo Terminal ha permitido que se aprenden las siguientes lecciones valiosas:

Este proyecto ha sido una oportunidad para aplicar y expandir los conocimientos en desarrollo web y gestión de proyectos. Se ha mejorado significativamente en la identificación y resolución de problemas técnicos y de diseño, aprendiendo a adaptarse y superar desafíos imprevistos. La colaboración con sinodales y director, ha sido fundamental para mejorar como Ingeniero en Sistemas, resaltando la importancia que tienen la retroalimentación y la comunicación efectiva para el éxito de este proyecto. Ver el sistema "Tempus Victor" funcionando y recibiendo retroalimentación positiva de los usuarios es una experiencia sumamente gratificante, validando el esfuerzo y la dedicación invertidos en este proyecto .

7. TRABAJO A FUTURO

El desarrollo de "Tempus Victor" no concluye con la entrega del proyecto; hay varios aspectos que pueden ser mejorados y ampliados en el futuro:

- **Despliegue completo del sistema:** Realizar el despliegue completo del sistema en un entorno de producción. Esto incluye la configuración de servidores, implementación de políticas de respaldo y recuperación de datos, y monitoreo constante del desempeño del sistema.
- **Optimización y Mantenimiento:** Continuar optimizando el sistema para mejorar su rendimiento y asegurar su estabilidad a largo plazo.
- **Nuevas Funcionalidades:** Integrar nuevas funcionalidades basadas en las necesidades y sugerencias de los usuarios.
- **Expansión de la Base de Usuarios:** Desarrollar estrategias para aumentar la adopción del sistema, dar a conocer públicamente el sistema web.
- **Seguridad y Privacidad:** Continuar mejorando las medidas de seguridad para proteger los datos de los usuarios, cumpliendo con las normativas y mejores prácticas de la industria.
- **Aplicación Móvil:** Desarrollar una versión móvil de "Tempus Victor" para iOS y Android, asegurando que los usuarios puedan gestionar sus tareas y tiempos desde cualquier lugar.

8. REFERENCIAS

Anna Pérez. No saber priorizar: 4 trucos para solucionar este problema. 15-enero-2016. URL:

<https://www.obsbusiness.school/blog/no-saber-priorizar-4-trucos-para-solucionar-este-problema> (visitado 2023).

Angular. Documentación Oficial de Angular. 2023. URL: <https://angular.io/> (visitado 2023).

Api, Amazon. ¿Qué es una interfaz de programación de aplicaciones (API)? 2024. URL:

<https://aws.amazon.com/es/what-is/api/>. (visitado 2024)

API REST, Red Hat. ¿Qué es una API REST? 31 de julio de 2023. URL:

<https://www.redhat.com/es/topics/api/what-is-a-rest-api> (visitado 2024).

Arquitectura tres niveles, IBM. ¿Qué es la arquitectura de tres niveles? 2023. URL:

<https://www.ibm.com/mx-es/topics/three-tier-architecture> . (visitado 2024)

Capitalismo Consciente. Organización personal: importancia y mejores prácticas. 5-abril-2022.

URL:<https://capitalismoconsciente.es/blog/organizacion-personal-importancia-y-mejores-practicas/> (visitado 2023).

Colaborador de DocuSign. Gestión del tiempo: ¿Qué es y por qué es importante? 18-octubre-2022. URL: <https://www.docusign.com/es-mx/blog/gestion-del-tiempo> (visitado 2023).

DB, Oracle. ¿Qué es una base de datos? 2024. URL: <https://www.oracle.com/mx/database/what-is-database/> (visitado 2024).

Dorit Sauer. 15 cosas que te hacen menos productivo. 2021. URL:

<https://www.thecornerofexcellence.com/es/cosas-hacen-menos-productivo/> (visitado 2023).

Express. Documentación oficial de Express. 2023. URL: <https://expressjs.com/es/> (visitado 2023).

Equipo Editorial de Indeed. 14 herramientas para organizar el trabajo. 7-agosto-2023. URL:
<https://www.indeed.com/orientacion-profesional/desarrollo-profesional/herramientas-organizar-trabajo> (visitado 2023).

Elena Moreno. 8 razones por las que una mala gestión del tiempo te está perjudicando. 8-febrero-2020. URL: <https://forbes.es/empresas/10316/8-razones-por-las-que-una-mala-gestion-del-tiempo-te-esta-perjudicando/> (visitado 2023).

Faustino Sánchez Rodríguez. Medida del Tamaño Funcional de Aplicaciones Software. Article. 1999. URL:

<http://www.cs.uns.edu.ar/~prf/teaching/APS16/downloads/Teoria/Metricas%20de%20Tama%C3%B1o%20Funcional%20-%20ALARCOS.pdf> (visitado 2023).

Frontend y Backend, AWS. ¿Cuál es la diferencia entre el Frontend y Backend en el desarrollo de aplicaciones? 2024. URL: <https://aws.amazon.com/es/compare/the-difference-between-frontend-and-backend/> (visitado 2024).

IFPUG. Function Point Analysis (FPA). n. d. URL: <https://ifpug.org/> (visitado 2023).

Julián Pérez Porto y María Merino. Tarea - Que es, usos, definición y concepto. 29-abril-2022. URL: <https://definicion.de/tarea/> (visitado 2023).

Julia Martins. Todo sobre la metodología GTD “getting things done” y cómo aplicarla en 5 pasos. 7-agosto-2022. URL: <https://asana.com/es/resources/getting-things-done-gtd> (visitado 2023).

Julio Martins. ¿Sobrecarga de trabajo? Estrategias para ayudar a equipos e individuos a recuperar el equilibrio. 20-enero-2023. URL: <https://asana.com/es/resources/overworked-signs> (visitado 2023).

JWT. Introduction to JSON Web Tokens. 2024. URL: <https://jwt.io/introduction> (visitado 2024).

Modelo Espiral Deloitte. ¿Qué es el desarrollo en Espiral? 2024. URL:

<https://www2.deloitte.com/es/es/pages/technology/articles/que-es-el-desarrollo-en-espiral.html> (visitado 2024).

MySQL. Documentación oficial de MySQL. 2023. URL: <https://www.mysql.com/> (visitado 2023).

Node.js. About Node.js®. 2024. URL: <https://nodejs.org/en/about> (visitado 2024).

NPM. npm Docs. 2024. URL: <https://docs.npmjs.com/> (visitado 2024).

ORM, Deloitte. ¿Qué es un ORM? 2024. URL:

<https://www2.deloitte.com/es/es/pages/technology/articles/que-es-orm.html> (visitado 2024).

Prabhat S. Difference Between Task and Activity. Difference Between Similar Terms and Objects. 8- junio-2016. URL: <http://www.differencebetween.net/language/words-language/difference-between-task-and-activity/> (visitado 2023).

PrezDev. análisis de puntos de función. Video. 2015. URL:

<https://youtu.be/BeP6dXdLlo8?si=keBUaiAD7PwjVZzq> (visitado 2023).

Prisma. Next-generation Node.js and TypeScript ORM. 2024. URL: <https://www.prisma.io/docs> (visitado 2024).

R García. Teoría General de Sistemas y Complejidad [Versión electrónica]. Contribuciones a la Ciencias Sociales. Febrero-2014. URL: <https://www.eumed.net/rev/ccccs/27/teoria-sistemas.html> (visitado 2023).

Sam Milbrath. ¿Ideas de cómo gestionar tareas y administrar el tiempo en un proyecto? 29-septiembre- 2021. URL: <https://blog.trello.com/es/dificultades-de-un-proyecto> (visitado 2023).

Salvador. Gestión del tiempo: clasificación de nuestras tareas. 21-octubre-2014. URL:

<https://greendok.com/gestion-del-tiempo-clasificacion-de-nuestras-tareas> (visitado 2023).

Salvador. Planificación y gestión del tiempo. Las áreas de actuación. 14-octubre-2014. URL: <https://greendok.com/planificacion-y-gestion-del-tiempo-las-areas-de-actuacion/> (visitado 2023).

Team Asana. La matriz de Eisenhower: cómo priorizar tu lista de tareas pendientes. 4-octubre-2023 URL: <https://asana.com/es/resources/eisenhower-matrix> (visitado 2023).

Todoist. La Técnica Pomodoro. 25-noviembre-2023. URL: <https://todoist.com/es/productivity-methods/pomodoro-technique> (visitado 2023).

TypeScript. What is TypeScript? 2024. URL: <https://www.typescriptlang.org/> (visitado 2024).

Visual Studio Code. Code Editing. Redefined. 2024. URL: <https://code.visualstudio.com/> (visitado 2024).

Walter Valdés. Definición de Tarea. 19-Julio-2023. URL: <https://conceptodefinicion.de/tarea/> (visitado 2023).

Yolanda Barquera. La regla de los dos minutos para mejorar tu productividad. 12-Julio-2019. URL: <https://idesaa.edu.mx/blog/la-regla-de-los-dos-minutos-mejorar-la-productividad/> (visitado 2023).

ANEXOS

Este capítulo contiene las actividades importantes en el desarrollo del sistema, es un apartado que expone completamente las actividades resumidas en el capítulo anterior, esto es la descripción detallada de la actividad.

Aquí podrás informarte sobre el desarrollo de Tempus Victor y sus características principales. Encontraras desde los riesgos identificados hasta las pantallas reales del sistema.

A. Análisis de riesgos

A pesar de que el análisis técnico y el análisis económico han demostrado que el sistema es viable para su desarrollo, se debe llevar a cabo un análisis de posibles riesgos que pueden surgir durante el desarrollo del sistema, esto con la finalidad de poder tener una posible mitigación o plan de contingencia.

A.1 Tabla de análisis de riesgos

Para realizar el análisis de riesgos se consideran los siguientes parámetros de clasificación: Catastrófico (1), Critico (2), Marginal (3), Despreciable (4)

A continuación, se muestra la tabla de probabilidad de ocurrencia, teniendo como porcentaje más bajo el 1 % de probabilidad de ocurrencia y siendo el 99 % el porcentaje más alto de ocurrencia, esta tabla nos permitirá asignarles a los riesgos enlistados una posible probabilidad de ocurrencia.

Tabla 8: Tabla estándar de probabilidad de ocurrencia

Probabilidad de ocurrencia	
Porcentajes	Alias
1 % - 10 %	Muy Bajo
11 % - 20 %	Bajo
21 % - 35 %	Bajo - Medio
36 % - 55 %	Medio
56 % - 80 %	Medio - Alto
81 % - 90 %	Alto
91 % - 99 %	Muy Alto

En la siguiente tabla se enlistan los posibles riesgos que podrían surgir durante el desarrollo del sistema, teniendo su respectivo identificador, la categoría a la que pertenecen, la probabilidad de ocurrencia, el impacto que tendrían dentro del sistema y como se mitigara en caso de llegar a ocurrir.

Tabla 9: Tabla de Riesgos

Id	Nombre	Categoría	Descripción	Prob.	Impacto
RT-1	Mala estimación de complejidad del proyecto.	Técnico	La complejidad de algunas funciones puede ser demasiada alta y esto puede afectar críticamente el desarrollo y la implementación de esta.	Medio	1
RT-2	Tiempo limitado para el desarrollo	Técnico	Contar con un tiempo muy limitado para llevar a cabo el desarrollo del sistema.	Medio-Alto	2
RA-3	Planificación demasiado optimista	Administrativo	Se establecen plazos y presupuestos basados en expectativas excesivamente optimistas, lo que puede llevar a una falta de realismo en la ejecución del proyecto y a incumplimientos de objetivos.	Bajo-Medio	2
RT-4	Falla del equipo de desarrollo	Técnico	Que el equipo deje de funcionar o presente problemáticas capaces de pausar o terminar el avance.	Alto	1
RT-5	Problemas de rendimiento	Técnico	Los sistemas, aplicaciones o componentes no funcionan adecuadamente juntos, lo que puede generar problemas de rendimiento o funcionalidad.	Bajo	3
RA-6	Cambios en las necesidades del negocio	Administrativo	Posibilidad de que las necesidades y requisitos del negocio evolucionen durante el proyecto, lo que puede requerir ajustes en el alcance y la planificación.	Bajo	3
RT-7	Dificultad en la utilización del sistema	Técnico	Dificultad que los usuarios puedan experimentar al utilizar el producto o servicio, lo que puede afectar la adopción del sistema.	Bajo	4
RT-8	Problemas de compatibilidad con tecnologías.	Técnico	Posibilidad de que las tecnologías utilizadas en el proyecto cambien durante su desarrollo, lo que puede afectar la compatibilidad y la eficiencia del proceso de desarrollo.	Medio	2
RT-9	Desarrollador no está en óptimas condiciones	Técnico	El desarrollador puede enfermar o sufrir un accidente, lo que podría retrasar o interrumpir el desarrollo del proyecto	Medio-alto	4

Nota: Elaboración propia

A.2 Fichas de Riesgo

A continuación, se presentan las fichas de cada uno de los riesgos presentados en la tabla anterior, en la cual se abarca de mejor manera la información de cada uno, así como su debido plan de contingencia.

Tabla 10: Ficha de Riesgo RT-1

Ficha de información de riesgo			
ID Riesgo: RT-1	Fecha: 09/10/2023	Probabilidad: 50 %	Impacto: 1
Descripción: La complejidad de algunas funciones puede ser demasiada alta y esto puede afectar críticamente el desarrollo y la implementación de esta.			
Refinamiento del contexto: <p><u>Condición 1:</u> El desarrollo del proyecto podría verse afectado o retrasado en ciertas funciones que se buscaban implementar, ya que estas funciones fueron demasiado complejas para el equipo de trabajo actual.</p> <p><u>Condición 2:</u> Limitaciones en las herramientas de desarrollo que podrían intervenir de diversas maneras durante el desarrollo del sistema.</p>			
Mitigación/monitoreo: <ol style="list-style-type: none"> Replantear los requerimientos funcionales y principales que conformaran el sistema, teniendo en cuenta la complejidad y tiempo de desarrollo que supone cada uno de los requerimientos. Contemplar el software con el que se planea desarrollar el sistema e informarse sobre sus limitaciones y beneficios. 			
Manejo/plan de contingencia: Reevaluar los requisitos del sistema y centralizarlo para que este sea alcanzable.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

Nota: Elaboración propia

Tabla 11: Ficha de Riesgo RT-2

Ficha de información de riesgo			
ID Riesgo: RT-2	Fecha: 09/10/2023	Probabilidad: 70 %	Impacto: 2
Descripción: Contar con un tiempo muy limitado para llevar a cabo el desarrollo del sistema.			
Refinamiento del contexto: <p><u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta.</p> <p><u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.</p>			
Mitigación/monitoreo: <ol style="list-style-type: none"> Separar y reorganizar los módulos y requerimientos de acuerdo con el tiempo de desarrollo que requiere cada uno. Priorizar el desarrollo de los módulos más importantes y significativos del sistema. 			
Manejo/plan de contingencia: Tener un plan de desarrollo ya establecido, priorizando el desarrollo de los módulos y funciones principales del sistema.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

Nota: Elaboración propia

Tabla 12: Ficha de Riesgo RT-3

Ficha de información de riesgo			
ID Riesgo: RA-3	Fecha: 09/10/2023	Probabilidad: 30 %	Impacto: 2
Descripción: Se establecen plazos y presupuestos basados en expectativas excesivamente optimistas, lo que puede llevar a una falta de realismo en la ejecución del proyecto y a incumplimientos de objetivos			
Refinamiento del contexto:			
<u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta.			
<u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.			
Mitigación /monitoreo:			
1. Separar y reorganizar los módulos y requerimientos de acuerdo con el tiempo de desarrollo que requiere cada uno. 2. Priorizar el desarrollo de los módulos más importantes y significativos del sistema.			
Manejo/plan de contingencia:			
Tener un plan de desarrollo ya establecido, priorizando el desarrollo de los módulos y funciones principales del sistema.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

*Nota: Elaboración propia**Tabla 13: Ficha de Riesgo RT-4*

Ficha de información de riesgo			
ID Riesgo: RT-4	Fecha: 09/10/2023	Probabilidad: 85 %	Impacto: 2
Descripción: Que el equipo deje de funcionar o presente problemáticas capaces de pausar o terminar el avance			
Refinamiento del contexto:			
<u>Condición 1:</u> El avance en el desarrollo del proyecto puede verse afectado en gran medida si el equipo de cómputo se ve afectado de alguna medida.			
<u>Condición 2:</u> La adquisición o reparación del equipo puede ser un gran problema, sin solución inmediata			
Mitigación /monitoreo:			
1. Tener opciones de equipos de cómputo de respaldo 2. Identificar problemáticas y hacer mantenimiento preventivo al equipo de cómputo actual			
Manejo/plan de contingencia:			
Hacer lo posible por conseguir un equipo de cómputo extra para tener un plan en caso de emergencias.			
Estado actual: 16/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

Nota: Elaboración propia

Tabla 14: Ficha de Riesgo RT-5

Ficha de información de riesgo			
ID Riesgo: RT-5	Fecha: 09/10/2023	Probabilidad: 20 %	Impacto: 3
Descripción: Los sistemas, aplicaciones o componentes no funcionan adecuadamente juntos, lo que puede generar problemas de rendimiento o funcionalidad.			
Refinamiento del contexto: <u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta. <u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.			
Mitigación /monitoreo: 1. Separar y reorganizar los módulos y requerimientos de acuerdo con el tiempo de desarrollo que requiere cada uno. 2. Priorizar el desarrollo de los módulos más importantes y significativos del sistema.			
Manejo/plan de contingencia: Tener un plan de desarrollo ya establecido, priorizando el desarrollo de los módulos y funciones principales del sistema.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

*Nota: Elaboración propia**Tabla 15: Ficha de Riesgo RT-6*

Ficha de información de riesgo			
ID Riesgo: RA-6	Fecha: 09/10/2023	Probabilidad: 15 %	Impacto: 3
Descripción: Posibilidad de que las necesidades y requisitos del negocio evolucionen durante el proyecto, lo que puede requerir ajustes en el alcance y la planificación.			
Refinamiento del contexto: <u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta. <u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.			
Mitigación /monitoreo: 1. Separar y reorganizar los módulos y requerimientos de acuerdo con el tiempo de desarrollo que requiere cada uno. 2. Priorizar el desarrollo de los módulos más importantes y significativos del sistema.			
Manejo/plan de contingencia: Tener un plan de desarrollo ya establecido, priorizando el desarrollo de los módulos y funciones principales del sistema.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

Nota: Elaboración propia

Tabla 16: Ficha de Riesgo RT-7

Ficha de información de riesgo			
ID Riesgo: RT-7	Fecha: 09/10/2023	Probabilidad: 20 %	Impacto: 4
Descripción: Dificultad que los usuarios puedan experimentar al utilizar el producto o servicio, lo que puede afectar la adopción del sistema.			
Refinamiento del contexto: <u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta. <u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.			
Mitigación /monitoreo: 1. Separar y reorganizar los módulos y requerimientos de acuerdo con el tiempo de desarrollo que requiere cada uno. 2. Priorizar el desarrollo de los módulos más importantes y significativos del sistema.			
Manejo/plan de contingencia: Tener un plan de desarrollo ya establecido, priorizando el desarrollo de los módulos y funciones principales del sistema.			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

*Nota: Elaboración propia***Tabla 17: Ficha de Riesgo RT-8**

Ficha de información de riesgo			
ID Riesgo: RT-8	Fecha: 09/10/2023	Probabilidad: 40 %	Impacto: 2
Descripción: Posibilidad de que las tecnologías utilizadas en el proyecto cambien durante su desarrollo, lo que puede afectar la compatibilidad y la eficiencia del proceso de desarrollo. Implica la necesidad de considerar la posibilidad de cambiar las tecnologías de desarrollo si es necesario.			
Refinamiento del contexto: <u>Condición 1:</u> Modificaciones en los requisitos del proyecto pueden hacer que las tecnologías actuales sean inadecuadas, lo que podría generar la necesidad de adoptar nuevas tecnologías. <u>Condición 2:</u> Por toma de decisiones estratégicas que afecten el cambio de las tecnologías de desarrollo del proyecto.			
Mitigación /monitoreo: 1. Diseñar la arquitectura del proyecto de manera que permita la fácil sustitución de tecnologías si es necesario. Utilizar estándares y enfoques que reduzcan la dependencia de tecnologías específicas.			
Manejo/plan de contingencia: Identificar tecnologías alternativas que sean compatibles con el proyecto y que puedan ser adoptadas de manera eficiente.			
Estado actual: 16/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado J. C. Hernández		

Nota: Elaboración propia

Tabla 18: Ficha de Riesgo RT-9

Ficha de información de riesgo			
ID Riesgo: RT-9	Fecha: 09/10/2023	Probabilidad: 60 %	Impacto: 4
Descripción: El desarrollador puede enfermar o sufrir un accidente, lo que podría retrasar o interrumpir el desarrollo del proyecto			
Refinamiento del contexto:			
<p><u>Condición 1:</u> La calidad del proyecto podría verse afectada debido al tiempo de desarrollo con el que se cuenta.</p> <p><u>Condición 2:</u> Ciertos requerimientos o módulos podrían verse comprometidos e incluso ser eliminados debido al tiempo de desarrollo que requiere cada uno.</p>			
Mitigación/monitoreo:			
<ol style="list-style-type: none"> 1. Identificar planes de emergencia en caso de accidente, pues puede ser grave o ligero 2. Hacer chequeos constantes para evitar cualquier enfermedad. 			
Manejo/plan de contingencia:			
Tomar acciones preventivas en cualquier momento del desarrollo, priorizando la salud, lo que permitirá un desarrollo sin emergencias			
Estado actual: 09/10/2023: Pasos de mitigación iniciados.			
Originador: J. C. Hernández	Asignado: J. C. Hernández		

Nota: Elaboración propia

B. Requerimientos funcionales

Tabla 19: Requerimientos funcionales del sistema

Id	Nombre	Requerimientos Funcionales	Usuario	PF
		Descripción		
RF1	Registrar Usuario	El usuario es capaz de registrarse en el sistema, proporcionando un nombre de usuario, una dirección de correo electrónico y una contraseña.	Estudiante	(EI)3
RF2	Iniciar Sesión	El usuario es capaz de iniciar sesión en el sistema, proporcionando su nombre de usuario o su correo electrónico, así como su contraseña.	Estudiante	(EI)3
RF3	Cerrar Sesión	El usuario es capaz de cerrar sesión en el sistema en cualquier momento.	Estudiante	(EO)4
RF4	Ver Información cuenta	El usuario es capaz de ver la información de su cuenta, incluyendo su nombre de usuario, dirección de correo electrónico y fecha de creación de la cuenta.	Estudiante	(EQ)3
RF5	Editar cuenta	El usuario es capaz de editar los datos de su cuenta, incluyendo su nombre de usuario y foto de perfil.	Estudiante	(EI)4
RF6	Cambiar Contraseña	El usuario es capaz de cambiar su contraseña, proporcionando su contraseña actual y una nueva contraseña.	Estudiante	(EI)4
RF7	Eliminar cuenta	El usuario es capaz de eliminar su cuenta, confirmando que desea eliminarla.	Estudiante	(EO)4
RF8	Crear Nota	El usuario es capaz de crear notas simples, con título y descripción	Estudiante	(EI)4
RF9	Seleccionar Nota	El usuario es capaz de seleccionar una nota específica, para actuar sobre ella.	Estudiante	(EQ)3
RF10	Ver Nota	El usuario es capaz de ver el contenido de una nota seleccionada	Estudiante	(EQ)4

RF11	Editar Nota	El usuario es capaz de editar el contenido de una nota seleccionada.	Estudiante	(EI)3
RF12	Eliminar Nota	El usuario es capaz de eliminar una nota seleccionada.	Estudiante	(EO)4
RF13	Archivar Nota	El usuario es capaz de archivar las notas que ya no necesite ver en su lista de notas.	Estudiante	(EQ)4
RF14	Buscar Nota	El usuario es capaz de buscar una nota específica, por título o contenido.	Estudiante	(EI)4
RF15	Ayuda Notas	El usuario es capaz de obtener ayuda sobre las funcionalidades del módulo de Notas	Estudiante	(EQ)2
RF16	Crear Tarea	El usuario es capaz de crear una tarea con título, descripción, comentarios, urgencia e importancia	Estudiante	(EI)4
RF17	Agregar fecha y hora límite	El usuario es capaz de agregar una fecha y hora límite para la finalización de una tarea.	Estudiante	(EI)3
RF18	Agregar categoría	El usuario es capaz de agregar una categoría a una tarea, ya sea de forma personalizada o una ya predefinida	Estudiante	(EI)3
RF19	Seleccionar Tarea	El usuario es capaz de seleccionar una tarea específica, para actuar sobre ella.	Estudiante	(EQ)3
RF20	Ver Tarea	El usuario es capaz de ver el contenido de una tarea seleccionada.	Estudiante	(EQ)4
RF21	Editar Tarea	El usuario es capaz de editar el contenido de una tarea seleccionada.	Estudiante	(EI)4
RF22	Eliminar Tarea	El usuario es capaz de eliminar una tarea seleccionada.	Estudiante	(EO)4
RF23	Marcar tarea como Completada	El usuario es capaz de marcar una tarea seleccionada como completada, en cualquiera de las vistas del sistema.	Estudiante	(EI)3
RF24	Cambiar estado de tarea	El usuario es capaz de cambiar el estado de una tarea seleccionada, al estado de no completada, dentro de la lista de tareas Completadas.	Estudiante	(EI)3
RF25	Vista Matriz Eisenhower	El usuario es capaz de ver las tareas en una vista de matriz de Eisenhower, clasificadas en cuatro categorías: urgentes e importantes, urgentes, pero no importantes, no urgentes pero importantes y no urgentes e importantes.	Estudiante	(EQ)6
RF26	Buscar Tarea	El usuario es capaz de buscar una tarea por su título, dentro de la vista de Matriz Eisenhower.	Estudiante	
RF27	Filtrar Tarea	El usuario es capaz de filtrar las tareas por su categoría, dentro de la vista de Matriz Eisenhower.	Estudiante	
RF28	Ayuda Tareas	El usuario es capaz de obtener ayuda sobre las funcionalidades del módulo de Tareas	Estudiante	(EQ)2
RF29	Ver lista Tareas Completadas	El usuario es capaz de ver una lista de tareas ya completadas.	Estudiante	(EQ)4
RF30	Ver Historial tarea	El usuario es capaz de ver el historial de una tarea, viendo todos los cambios que se han realizado en la tarea desde su creación.	Estudiante	(EQ)6
RF31	Ver eventos Tarea	El usuario es capaz de ver todos los eventos que le ocurrieron a la tarea seleccionada, con fecha y hora.	Estudiante	
RF32	Ver eventos Pomodoro	El usuario es capaz de ver todos los eventos que ocurrieron sobre la sesión Pomodoro de la tarea seleccionada, con fecha y hora.	Estudiante	(EQ)6

RF33	Actualizar prioridad y Urgencia	El usuario es capaz de arrastrar una tarea seleccionada entre cuadrantes de la matriz de Eisenhower, cambiando su prioridad y urgencia.	Estudiante	(EI)6
RF34	Vista Calendario	El usuario es capaz de ver las tareas en una vista de calendario, clasificadas por fecha.	Estudiante	(EQ)6
RF35	Seleccionar Dia en el calendario	El usuario es capaz de seleccionar cualquier día del mes actual de la vista.	Estudiante	
RF36	Cambiar Entre Meses	El usuario es capaz de cambiar la vista entre meses del año, por defecto se muestra el mes concurrente.	Estudiante	
RF37	Ver Lista Tareas sin Fecha Limite	El usuario es capaz de ver una lista de tareas que no tienen fecha límite asignada.	Estudiante	
RF38	Ver Lista de Tareas del día Seleccionado	El usuario es capaz de ver una lista de tareas del día que selecciono previamente.	Estudiante	
RF39	Actualizar Fecha Límite Tarea seleccionada	El usuario es capaz de mover entre los días del mes de la vista actual, a la tarea seleccionada previamente.	Estudiante	
RF40	Estimar Pomodoros	El usuario es capaz de estimar el tiempo que le tomara realizar una tarea seleccionada, esto se calculara por Pomodoros.	Estudiante	
RF41	Temporizador Pomodoro	El usuario es capaz de ver la información de la tarea seleccionada, la estimación en Pomodoros que se realizó, el temporizador de la sesión, y los tiempos de la sesión.	Estudiante	(EI)6 (EQ)6
RF42	Reiniciar Sesión Pomodoro	El usuario es capaz de reiniciar la sesión Pomodoro actual, reiniciando todos los valores y tiempos relacionados a esta misma.	Estudiante	
RF43	Iniciar Sesión Pomodoro	El usuario es capaz de iniciar la sesión Pomodoro actual, si es que no se ha iniciado.	Estudiante	
RF44	Continuar Sesión Pomodoro	El usuario es capaz de continuar la sesión Pomodoro actual, siempre y cuando esta sea una sesión pausada.	Estudiante	
RF45	Pausar Sesión Pomodoro	El usuario es capaz de pausar la sesión Pomodoro actual, siempre y cuando este corriendo el temporizador.	Estudiante	
RF46	Terminar Sesión Pomodoro	El usuario es capaz de Terminar la sesión Pomodoro actual en cualquier momento que desee, guardando los valores de la sesión y preguntando si desea completar la Tarea.	Estudiante	
RF47	Sesión Pomodoro Abandonada	Si una sesión Pomodoro, se pausa o se abandona por 3 minutos se convierte en una sesión Abandonada.	Estudiante	
RF48	Ingreso Sesión Pomodoro sin Iniciar	Al ingresar a una sesión Pomodoro y esta es una sesión abandonada, el usuario es capaz de iniciarla o salir a la vista anterior.	Estudiante	
RF49	Ingreso Sesión Pomodoro Abandonada	Al ingresar a una sesión Pomodoro y esta es una sesión abandonada, el usuario es capaz de reanudarla o salir a la vista anterior.	Estudiante	(EQ)4

RF50	Ingreso Sesión Pomodoro Terminada	Al ingresar a una sesión Pomodoro y esta es una sesión terminada, el usuario es capaz de reiniciarla, de ver el historial o de salir a la vista anterior.	Estudiante	
RF51	Alerta Sonora Fin Trabajo	El usuario recibe una alerta sonora al final de cada iteración de Trabajo, para continuar al Descanso.	Estudiante	(EO)2
RF52	Alerta Sonora Fin Descanso	El usuario recibe una alerta sonora al final de cada iteración de Descanso, para completar un Pomodoro, y empezar una nueva iteración de Trabajo.	Estudiante	(EO)3
RF53	Gestionar Sonido Sesión Pomodoro	El usuario es capaz de activar o desactivar el sonido del temporizador cuando está corriendo.	Estudiante	
RF54	Ayuda Pomodoro	El usuario es capaz de obtener ayuda sobre las funcionalidades del módulo de Temporizador Pomodoro.	Estudiante	(EQ)2
RF55	Métrica de Tareas	El usuario es capaz de ver las métricas calculadas por el sistema sobre los tiempos de tareas ya completadas, y dependiendo de la categoría de las tareas el usuario podrá ver: Tareas completadas, Tiempo promedio por Tarea, Tareas completadas a tiempo.	Estudiante	(EQ)6
RF56	Métrica de Sesiones Pomodoro	El usuario es capaz de ver las métricas calculadas por el sistema sobre las sesiones de Pomodoro pasadas, y dependiendo de las categorías de las tareas el usuario podrá ver: tiempo promedio de sesiones, tiempos de descanso, tiempos de trabajo, iteraciones de Pomodoro promedio.	Estudiante	(EQ)6
RF57	Personalización de Pomodoro	El usuario es capaz de personalizar las iteraciones de Pomodoro, el sonido de la alerta sonora, así como los tiempos de trabajo y descanso.	Estudiante	(EI)3
RF58	Personalizar estilo	El usuario es capaz de cambiar el estilo del sistema, incluyendo el color de fondo del sistema, el tipo de fuente, así como otros elementos visuales.	Estudiante	(EI)6
RF59	Personalizar notificaciones	El usuario es capaz de editar el sonido y duración de las notificaciones del sistema	Estudiante	(EI)4
RF60	Personalizar fondos de Temporizador	El usuario es capaz de escoger las distintas imágenes que se muestran en el temporizador cuando este cambia su estado: sin iniciar, abandonada, trabajando o descansando.	Estudiante	
RF61	Ver Tiempo y Fecha en tiempo Real	El usuario es capaz de ver en la barra del menú de opciones la fecha y la hora actuales, la cual se actualiza en tiempo real	Estudiante	

Nota: Elaboración Propia

C. Requerimientos No Funcionales

Tabla 20: Requerimientos No Funcionales

Id	Atributo	Requerimientos No funcionales	
		Necesidad/Estrategia	
RNF1	Autenticación	Necesidad: Verificar que el usuario es quien dice ser. Estrategia: Realizar una consulta en la base de datos y validar que la información ingresada es idéntica a la información almacenada	
RNF2	Robustez	Necesidad: Que el sistema pueda seguir ejecutándose incluso si se presenta algún error inesperado e informar al usuario que se ha presentado un error. Estrategia: Implementar un correcto manejo de errores que muestren diversos mensajes al usuario cuando ocurre algún error específico.	
RNF3	Funcionalidad	Necesidad: Que el sistema cumpla con su propósito y función principal de manera adecuada y en todo momento. Estrategia: Se llevará a cabo diversas pruebas de funcionalidad para verificar que el propósito principal del sistema se está realizando correctamente	

Nota: Elaboración Propia

D. Modelo del Negocio

El Modelo de Negocio del sistema web Tempus Victor, compuesto por los Términos del Negocio y los Hechos del Negocio actualizados, establece un marco conceptual claro para comprender las funcionalidades y el funcionamiento del sistema.

Los **Términos del Negocio** definen los elementos fundamentales del sistema, como usuarios, tareas, notas, categorías, eventos, sesiones Pomodoro, calendario, Matriz de Eisenhower, Temporizador Pomodoro, personalización y métricas de productividad. Estos términos proporcionan un lenguaje común para describir y comprender el sistema.

Los **Hechos del Negocio** describen las relaciones y comportamientos entre los Términos del Negocio. Estos hechos establecen las reglas que rigen el funcionamiento del sistema, como la creación, edición y eliminación de tareas, la asociación de notas a tareas o categorías, la gestión del calendario, la priorización de tareas con la Matriz de Eisenhower, la utilización del Temporizador Pomodoro y el seguimiento de métricas de productividad. La comprensión clara de los Términos del Negocio y los Hechos del Negocio es esencial para:

- Desarrolladores: Implementar correctamente las funcionalidades del sistema y garantizar su funcionamiento acorde a las expectativas.
- Usuarios: Comprender las posibilidades y limitaciones del sistema para aprovecharlo al máximo en su gestión personal.

D.1 Términos del negocio

- Usuario: Cualquier estudiante que utilice el sistema.
- Tarea: Una actividad específica que debe ser completada por el usuario.
- Nota: Un mensaje textual que el usuario puede crear.
- Categoría: Una clasificación para agrupar tareas relacionadas.
- Evento: Registro de un cambio en el estado de una tarea.
- Sesión Pomodoro: Periodo de trabajo enfocado de 25 minutos, seguido de un breve descanso.
- Evento Pomodoro: Registro de un inicio, pausa o finalización de una sesión Pomodoro.
- Calendario: Un sistema para organizar eventos y tareas en el tiempo.
- Matriz de Eisenhower: Un método para priorizar tareas según su urgencia e importancia.
- Temporizador Pomodoro: Un método para dividir el trabajo en intervalos de 25 minutos con pausas de 5 minutos.
- Personalización: La capacidad de adaptar el sistema a las necesidades específicas de un usuario.
- Métricas de productividad: Indicadores que miden la productividad de un usuario.

D.2 Hechos del Negocio

- Un usuario puede crear, editar y eliminar tareas.
- Una tarea puede pertenecer a una categoría específica.
- Una tarea puede tener uno o más eventos asociados a su progreso.
- Un usuario puede iniciar, pausar y finalizar sesiones Pomodoro.
- Una sesión Pomodoro puede tener uno o más eventos Pomodoro asociados a su desarrollo.
- Un usuario puede crear y editar notas.
- Un usuario puede visualizar y gestionar su calendario.
- Un usuario puede aplicar la Matriz de Eisenhower para priorizar sus tareas.
- Un usuario puede utilizar el Temporizador Pomodoro para gestionar sus sesiones de trabajo enfocado.

- El sistema permite la personalización de la interfaz y las funcionalidades para adaptarse a las necesidades de cada usuario.
- El sistema proporciona métricas de productividad para que los usuarios puedan evaluar su desempeño.

El Modelo de Negocio de Tempus Victor, con sus Términos del Negocio y Hechos del Negocio actualizados, sirve como una brújula para navegar por el sistema, comprender su funcionamiento y aprovechar sus capacidades para el logro de objetivos de gestión personal y académica.

D.3 Reglas de negocio

Reglas de Negocio para Gestión de Cuenta

- Registro de Usuario
 - BR1: Un usuario solo puede registrarse proporcionando un nombre de usuario, una dirección de correo electrónico válida única y una contraseña que cumpla con los requisitos de seguridad (Mínimo 6 caracteres).
 - BR2: El sistema debe validar que el correo electrónico no esté previamente registrado.
 - BR3: El sistema debe validar que la contraseña sea la misma que ingresa en la confirmación de contraseña.
- Inicio de Sesión
 - BR4: El usuario solo puede iniciar sesión proporcionando su correo electrónico y su contraseña correcta.
 - BR5: El sistema debe validar que el correo electrónico corresponda a una cuenta existente
 - BR6: El sistema debe validar que la contraseña es correcta para el correo proporcionado
- Cierre de Sesión
 - BR7: El usuario puede cerrar sesión en cualquier momento desde cualquier dispositivo.
- Ver Información de Cuenta
 - BR8: El usuario puede ver su información de cuenta, la cual debe incluir su nombre de usuario, correo electrónico y foto de perfil
- Edición de Cuenta

- *BR9*: El usuario puede editar su nombre de usuario y foto de perfil.
- Cambio de Contraseña
 - *BR10*: Para cambiar la contraseña, el usuario debe proporcionar su la nueva contraseña dos veces.
 - *BR11*: El sistema debe validar que las dos contraseñas sean la misma, y que cumpla con los requisitos de seguridad (Mínimo 6 caracteres).
- Eliminación de Cuenta
 - *BR12*: El usuario debe confirmar su intención de eliminar su cuenta
 - *BR13*: La eliminación de la cuenta es permanente y todos los datos asociados serán eliminados.

Reglas de Negocio para Gestión de Notas

- Creación de Nota
 - *BR14*: El usuario puede crear una nota proporcionando un título de mínimo 3 caracteres de largo y una descripción.
 - *BR15*: Las notas deben estar vinculadas al usuario que las creó.
- Selección y Visualización de Nota
 - *BR16*: El usuario puede seleccionar y ver el contenido completo de una nota específica.
 - *BR17*: Las notas solo pueden ser vistas por el usuario que las creó.
- Edición y Eliminación de Nota
 - *BR18*: El usuario puede editar el contenido de una nota seleccionada.
 - *BR19*: El usuario puede eliminar una nota seleccionada, tras una confirmación.
- Archivado de Nota
 - *BR20*: El usuario puede archivar notas que ya no necesite ver en su lista principal.
 - *BR21*: Las notas archivadas se pueden ver en una sección separada.
- Búsqueda de Notas
 - *BR22*: El usuario puede buscar notas por título o contenido.
 - *BR23*: La búsqueda debe ser capaz de filtrar notas en tiempo real.
- Ayuda en Módulo de Notas
 - *BR24*: El sistema debe proporcionar una sección de ayuda para el módulo de notas, accesible en cualquier momento.

Reglas de Negocio para Gestión de Tareas

- Creación de Tarea
 - *BR25*: El usuario puede crear una tarea proporcionando un título, descripción, comentarios, urgencia e importancia.
 - *BR26*: Las tareas deben estar vinculadas al usuario que las creó.
- Fecha Límite y Categorías en Tareas
 - *BR27*: El usuario puede agregar una fecha y hora límite para la finalización de una tarea.
 - *BR28*: El usuario puede categorizar una tarea con categorías personalizadas o predefinidas.
- Selección, Visualización y Edición de Tarea
 - *BR29*: El usuario puede seleccionar y ver el contenido completo de una tarea específica.
 - *BR30*: El usuario puede editar el contenido de una tarea seleccionada.
- Eliminación y Estado de Tarea
 - *BR31*: El usuario puede eliminar una tarea seleccionada, tras una confirmación.
 - *BR32*: El usuario puede marcar una tarea como completada o cambiar su estado a no completada.
- Historial y Eventos de Tareas
 - *BR33*: El usuario puede ver una lista de tareas completadas.
 - *BR34*: El usuario puede ver el historial de cambios y eventos de una tarea, con fecha, hora y descripción.
- Vista Matriz Eisenhower y Filtros
 - *BR35*: El usuario puede ver sus tareas clasificadas en una matriz de Eisenhower.
 - *BR36*: El usuario puede buscar y filtrar tareas por categoría dentro de la matriz.
 - *BR37*: El usuario actualizar la importancia y urgencia de las tareas arrastrándolas entre los cuadrantes de la matriz Eisenhower.
 - *BR38*: Las tareas por defecto se colocan en el cuadrante de eliminar.
- Ayuda en Vista de Eisenhower
 - *BR39*: El sistema debe proporcionar una sección de ayuda para la vista de Eisenhower del módulo de tareas, accesible en cualquier momento.

- Vista Calendario
 - BR40: El usuario puede ver sus tareas con fecha límite en una vista de Calendario
 - BR41: El usuario actualizar la fecha límite de la tarea arrastrándola entre los días del calendario.
- Ayuda en Vista de Calendario
 - BR42: El sistema debe proporcionar una sección de ayuda para la vista de Calendario del módulo de tareas, accesible en cualquier momento.

Reglas de Negocio para Gestión de Pomodoro

- Sesiones de Pomodoro
 - BR43: El usuario solo puede acceder a una sesión Pomodoro de una tarea seleccionada si estima la cantidad de tiempo que considera le tomara completar la tarea.
 - BR44: El usuario puede iniciar, pausar, reanudar, reiniciar y terminar una sesión Pomodoro.
 - BR45: Si una sesión Pomodoro se abandona por más de 3 minutos, se marca como abandonada, esto es que el usuario no interactúe con los modales de alarmas.
 - BR46: Al ingresar a una sesión abandonada, el usuario puede reanudarla o salir.
 - BR47: Al ingresar a una sesión pausada, el usuario puede reanudarla o salir.
 - BR48: Al ingresar a una sesión terminada, el usuario puede reiniciarla, ir al historial de la tarea y la sesión Pomodoro o salir.
- Alertas y Notificaciones en Pomodoro
 - BR49: El usuario recibe alertas sonoras al final de cada iteración de trabajo y descanso.
 - BR50: El usuario puede activar o desactivar el sonido de fondo del temporizador cuando está corriendo.
- Ayuda en Pomodoro
 - BR51: El sistema debe proporcionar una sección de ayuda para el módulo Pomodoro, accesible en cualquier momento.

Reglas de Negocio para Personalización y Configuraciones

- Personalización del Sistema
 - BR52: El usuario puede personalizar los sonidos de alerta, y los tiempos de trabajo y descanso.

- BR53: El usuario puede escoger imágenes para los distintos estados del temporizador Pomodoro.
- Visualización en Tiempo Real
 - BR54: El sistema debe mostrar la fecha y hora actual en tiempo real en la barra de menú de opciones.

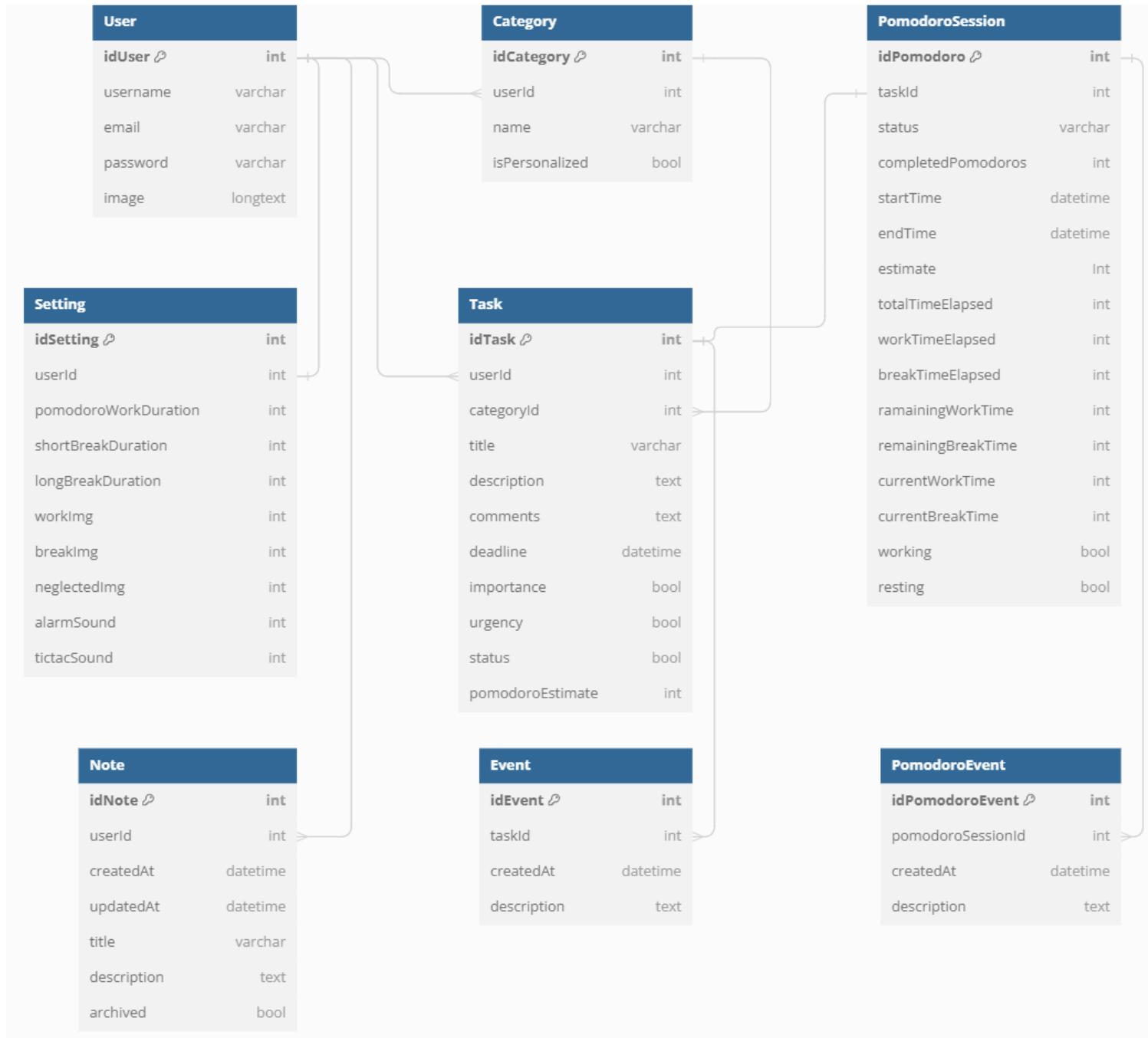
Reglas de Negocio para las Métricas

- BR55: El usuario puede ver métricas sobre las tareas de su cuenta, estas métricas contienen cantidades de tareas totales, tareas completadas y tareas sin completar.
- BR56: El usuario puede ver métricas sobre las sesiones de Pomodoro, estas métricas contienen cantidades de sesiones totales, sesiones sin iniciar, sesiones pausadas, sesiones terminadas y sesiones abandonadas

E. Modelo de la base de datos

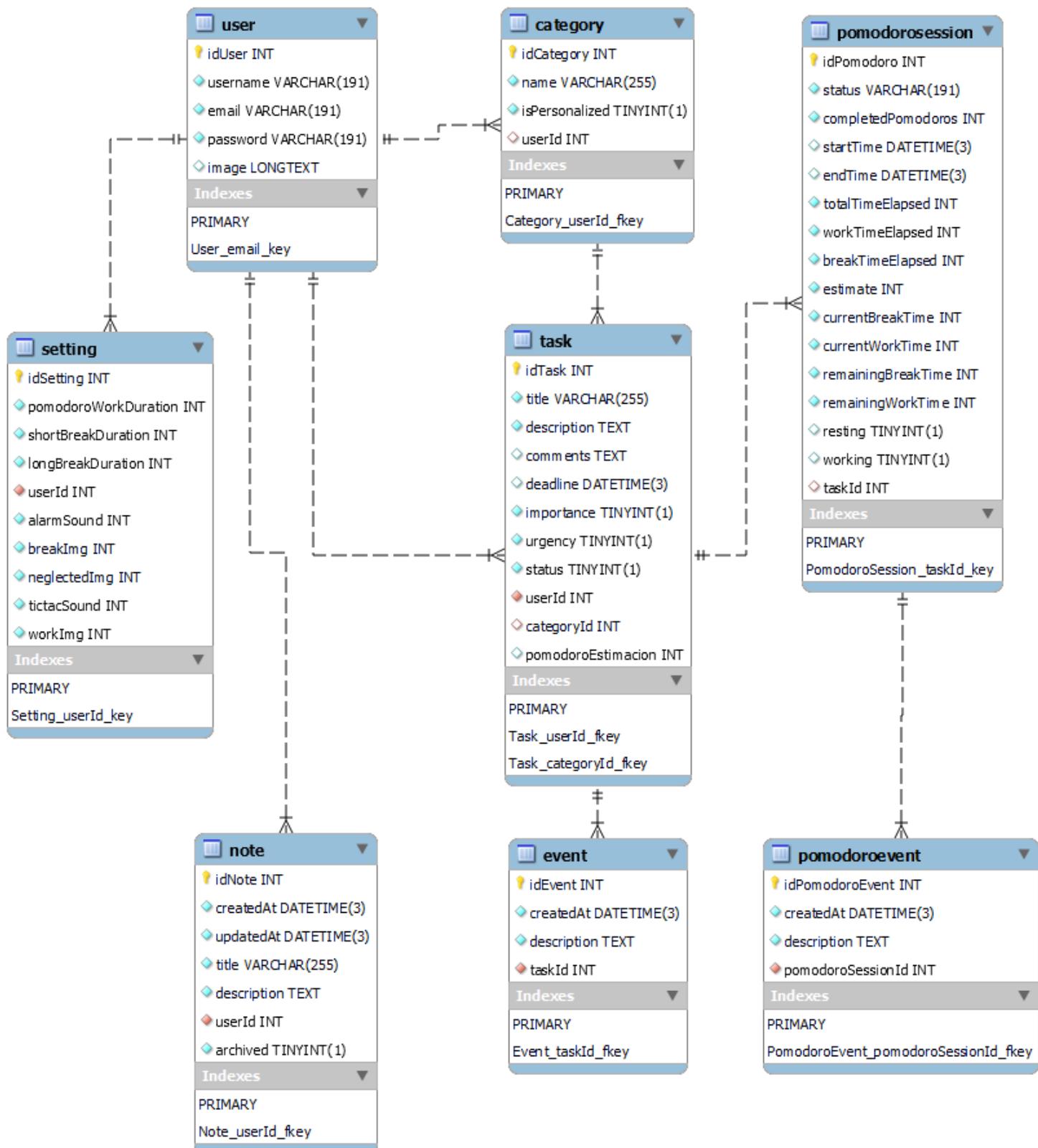
El modelo del dominio del problema se muestra en la siguiente imagen, después se describen cada una de las entidades y sus relaciones.

Imagen 58: Diseño de la base de datos



Nota: Elaboración Propia

Imagen 59: Base de datos Implementada



Nota: Elaboración Propia

Tabla 21: Entidad Usuario

Atributos			
Nombre	Tipo	Descripción	Requerido
idUser	<i>Id</i>	Número de registro utilizado para identificar una Usuario	Sí
username	<i>Cadena</i>	Nombre de la persona	Sí
email	<i>Correo</i>	Correo de la persona para acceder al sistema y para recuperación de contraseña	Sí
password	<i>Contraseña</i>	Contraseña relacionada al correo electrónico de la persona	Sí
image	<i>Texto Largo</i>	Cadena correspondiente a la imagen de foto de perfil convertida en base 64	Sí

*Nota: Elaboración Propia**Tabla 22: Entidad Configuración*

Atributos			
Nombre	Tipo	Descripción	Requerido
idSetting	<i>Id</i>	Número de registro utilizado para identificar a un Usuario	Sí
userId	<i>Cadena</i>	Numero de registro ajeno a la entidad para identificar a que usuario pertenece	Sí
pomodoroWorkDuration	<i>Entero</i>	Duración de trabajo del temporizador Pomodoro	Sí
shortBreakDuration	<i>Entero</i>	Duración de descanso corto del temporizador Pomodoro	Sí
longBreakDuration	<i>Entero</i>	Duración de descanso largo del temporizador Pomodoro	Sí
workImg	<i>Entero</i>	Numero de imagen de fondo para sesión Pomodoro trabajando	Sí
breakImg	<i>Entero</i>	Numero de imagen de fondo para sesión Pomodoro en descanso	Sí
neglectedImg	<i>Entero</i>	Numero de imagen de fondo para sesión Pomodoro abandonada	Sí
alarmSound	<i>Entero</i>	Numero para sonido de alarma del temporizador Pomodoro	Sí
tictacSound	<i>Entero</i>	Numero para sonido de fondo tictac del temporizador Pomodoro	Sí

Nota: Elaboración Propia

Tabla 23: Entidad Nota

Atributos			
Nombre	Tipo	Descripción	Requerido
idNote	<i>Id</i>	Número de registro utilizado para identificar una nota	Sí
userId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que usuario pertenece	Sí
createdAt	<i>Fecha y Hora</i>	Fecha de creación de la nota	Sí
updatedAt	<i>Fecha y Hora</i>	Fecha de última modificación de la nota	No
title	<i>Cadena</i>	Título de la nota	Sí
description	<i>Cadena</i>	Contenido de la nota escrita por la persona	Sí
archived	<i>Booleano</i>	Estado de la tarea, archivada o no archivada	si

*Nota: Elaboración Propia**Tabla 24: Entidad Tarea*

Atributos			
Nombre	Tipo	Descripción	Requerido
idTask	<i>Id</i>	Número de registro utilizado para identificar una nota	Sí
userId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que usuario pertenece	Sí
categoryId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que categoría pertenece	Sí
Title	<i>Cadena</i>	Título de la tarea	Sí
description	<i>Cadena</i>	Contenido de la tarea escrita por el usuario	Sí
Comments	<i>Cadena</i>	Comentarios sobre la tarea, estos son escritos por el usuario	No
Deadline	<i>Fecha y Hora</i>	Fecha límite para completar la tarea	No
importance	<i>Booleano</i>	Importancia de la tarea, false para no importante y true para importante	Sí
urgency	<i>Booleano</i>	Urgencia de las tareas, false para no urgente y true para urgente	Si
status	<i>Booleano</i>	Estado de la tarea, false para no completada, true para completada	Si
pomodoroEstimate	<i>Entero</i>	Estimación de tiempo en Pomodoros para la tarea	no

Nota: Elaboración Propia

Tabla 25: Entidad Eventos

Atributos				
Nombre	Tipo	Descripción	Requerido	
idEvent	<i>Id</i>	Número de registro utilizado para identificar un Evento	Sí	
taskId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que tarea pertenece	Sí	
createdAt	<i>Fecha y Hora</i>	Fecha y hora de creación del evento	Sí	
description	<i>Cadena</i>	Descripción del evento	Sí	

*Nota: Elaboración Propia**Tabla 26:Entidad Categoría*

Atributos				
Nombre	Tipo	Descripción	Requerido	
idCategory	<i>id</i>	Número de registro utilizado para identificar una categoría	Sí	
userId	<i>Id Foráneo</i>	Numero de registro ajeno a la entidad para identificar a que usuario pertenece	no	
name	<i>Cadena</i>	Nombre de la categoría	Si	
isPersonalized	<i>Booleano</i>	Si la categoría es del sistema o si la creo el usuario	Si	

*Nota: Elaboración Propia**Tabla 27: Entidad Sesiones Pomodoro*

Atributos				
Nombre	Tipo	Descripción	Requerido	
idPomodoro	<i>Id</i>	Número de registro utilizado para identificar un Evento	Sí	
taskId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que tarea pertenece	Sí	
status	<i>Cadena</i>	Estado de la tarea, “SinIniciar”, “Abandonada”, “Pausada”, “Terminada”	Sí	
completedPomodoros	<i>Entero</i>	Cantidad de Pomodoros completados	Sí	
startTime	<i>Fecha y Hora</i>	Fecha y hora de inicio de la sesión	Sí	
endTime	<i>Fecha y Hora</i>	Fecha y hora de finalización de la sesión	Sí	
estimate	<i>Entero</i>	Cantidad de Pomodoros estimados para la sesión	Sí	
totalTimeElapsed	<i>Entero</i>	Tiempo acumulado total de la sesión en segundos	Sí	
workTimeElapsed	<i>Entero</i>	Tiempo acumulado de trabajo de la sesión en segundos	Sí	
breakTimeElapsed	<i>Entero</i>	Tiempo acumulado de descanso de la sesión en segundos	Sí	
remainingWorkTime	<i>Entero</i>	Tiempo restante de trabajo del Pomodoro actual	Sí	

remainigBreakTime	<i>Entero</i>	Tiempo restante de descanso del Pomodoro actual	Sí
currentWorkTime	<i>Entero</i>	Tiempo actual de trabajo de la sesión	Sí
currentBreakTime	<i>Entero</i>	Tiempo actual de descanso de la sesión	Sí
working	<i>Booleano</i>	Estado de la sesión, false no está trabajando, true está trabajando	Sí
resting	<i>Booleano</i>	Estado de la sesión, false no está descansando, true está descansando	Sí

Nota: Elaboración Propia

Tabla 28: Entidad Eventos Pomodoro

Atributos			
Nombre	Tipo	Descripción	Requerido
idPomodoroEvent	<i>Id</i>	Número de registro utilizado para identificar un Evento	Sí
pomodoroSessionId	<i>Id Foráneo</i>	Número de registro ajeno a la entidad para identificar a que sesión Pomodoro pertenece	Sí
createdAt	<i>Fecha y Hora</i>	Fecha y hora de creación del evento	Sí
description	<i>Cadena</i>	Descripción del evento	Sí

Nota: Elaboración Propia

Entidades:

- Usuario: Representa a los usuarios del sistema.
- Configuración: Representa la configuración del sistema asociada a la cuenta
- Nota: Almacena las notas que los usuarios pueden crear.
- Tareas: Almacena las tareas que el usuario puede crear.
- Eventos: Guarda los eventos de las tareas
- Categoría: Almacena las categorías personalizadas del usuario y las del sistema
- Sesiones Pomodoro: Registra las sesiones de trabajo con la técnica Pomodoro.
- Eventos Pomodoro: Guarda los eventos de las sesiones Pomodoro.

Relaciones:

- Usuario:
 - Tiene una configuración (1: 1): Cada usuario tiene una configuración única asociada con su cuenta.

- Tiene muchas notas (1: N): Un usuario puede tener un número ilimitado de notas.
- Tiene muchas categorías (1: N): Un usuario puede tener un número ilimitado de categorías personalizadas, además de las categorías del sistema.
- Tiene muchas tareas (1: N): Un usuario puede tener un número ilimitado de tareas.
- Configuración
 - Le pertenece a un usuario (1: 1): Cada configuración pertenece a un usuario único.
- Nota:
 - Pertenece a un usuario (N:1): Cada nota pertenece a un usuario específico
- Categoría
 - Pertenece a un usuario (N: 1): Cada categoría personalizada pertenece a un usuario específico
 - Tiene muchas tareas (1: N): Una categoría puede estar asociada con varias tareas
- Tareas:
 - Pertenece a un usuario (N:1): Cada tarea pertenece a un usuario específico
 - Tiene muchos eventos (N: N): Una tarea puede tener un número ilimitado de eventos asociados con su progreso.
 - Tiene una sesión Pomodoro(N:1): Una tarea puede tener una sesión Pomodoro asociada para registrar el tiempo dedicada a ella.
 - Tiene una categoría (N:1): Una tarea puede estar asociada con una categoría específica.
- Eventos:
 - Pertenece a una tarea (N:1): Cada evento está asociado con una tarea específica
- Sesión Pomodoro:
 - Pertenece a una tarea (1:1): Una sesión Pomodoro está asociada con una tarea específica

- Tiene muchos eventos Pomodoro (1: N): Una sesión Pomodoro puede tener un número ilimitado de eventos que registran su progreso
- Evento Pomodoro:
 - Pertenece a una sesión Pomodoro (N:1): Cada evento Pomodoro está asociado con una sesión Pomodoro específica

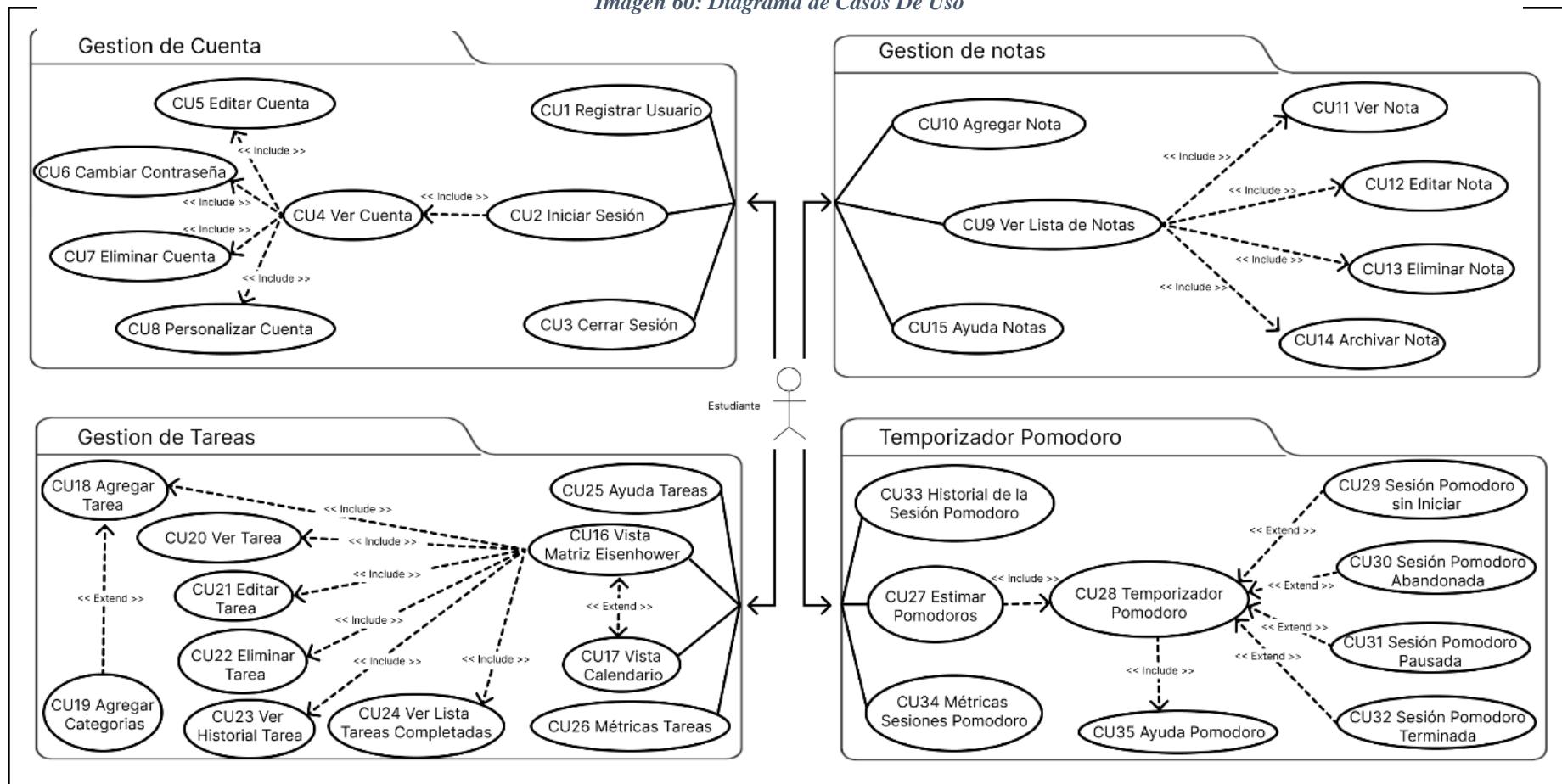
El modelo de datos de "Tempus Victor" está diseñado para almacenar y gestionar de manera eficiente la información de los usuarios, sus configuraciones, notas, categorías, tareas, eventos y sesiones Pomodoro. Las relaciones entre las entidades permiten mantener la integridad de los datos y facilitar la recuperación y el análisis de la información.

F. Modelo De Casos de Uso

En primera instancia se explica el diagrama de casos de uso general:

F.1 Diagrama de Casos de Uso

Imagen 60: Diagrama de Casos De Uso



Nota: Elaboración Propia

A continuación, se detallan los casos de uso.

F.2 Gestión de Cuenta

Imagen 61: Casos de Uso Gestión de Cuenta

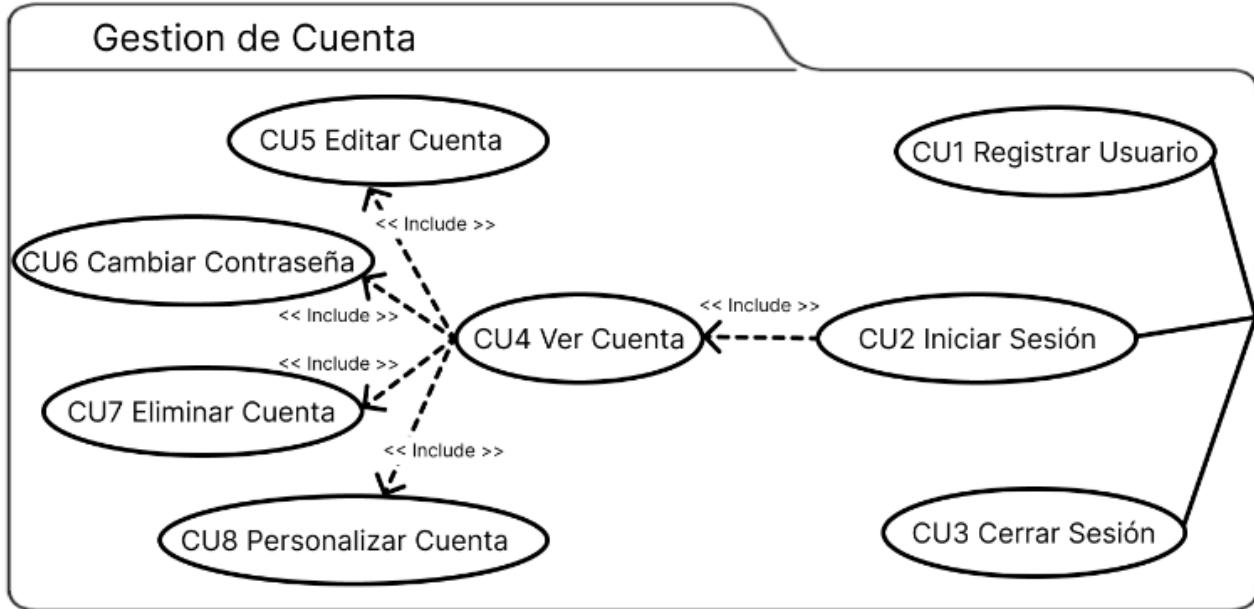


Tabla 29: Casos de Uso Gestión de Cuenta

Casos de Uso para Gestión de cuenta		
Id CU	Nombre CU	Descripción
CU1	Registrar Usuario	Permite a un nuevo usuario crear una cuenta en el sistema.
CU2	Iniciar Sesión	Permite a un usuario registrado iniciar sesión en el sistema para acceder a sus funcionalidades.
CU3	Cerrar Sesión	Permite a un usuario registrado cerrar su sesión en el sistema.
CU4	Ver Cuenta	Permite a un usuario registrado ver la información de su cuenta, como nombre, correo electrónico.
CU5	Editar Cuenta	Permite a un usuario registrado editar la información de su cuenta.
CU6	Cambiar Contraseña	Permite a un usuario registrado cambiar su contraseña.
CU7	Eliminar Cuenta	Permite a un usuario registrado eliminar su cuenta del sistema.
CU8	Personalizar Cuenta	Permite a un usuario registrado personalizar su cuenta, como cambiar foto de perfil, cambiar el tema de la interfaz, y toda la parte de configuración del sistema.

Nota: Elaboración Propia

F.3 Gestión de Notas

Imagen 62: Casos de Uso Gestión de Notas

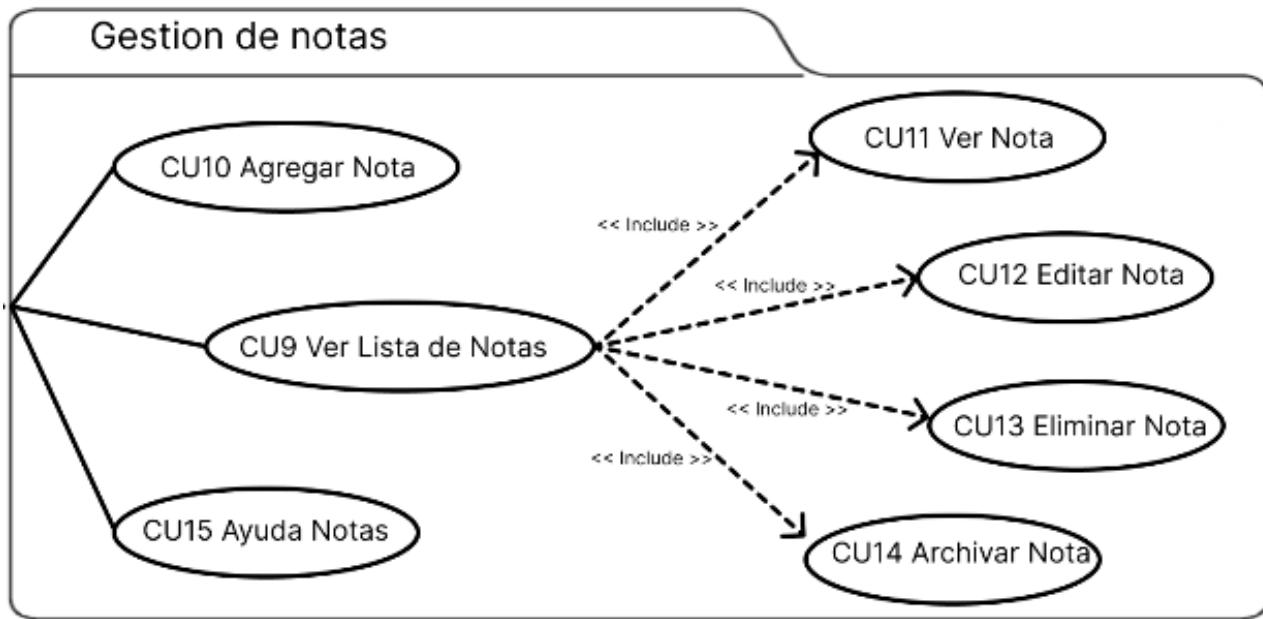


Tabla 30: Casos de Uso Gestión de Notas
Casos de Uso para Gestión de Notas

Id CU	Nombre CU	Descripción
CU9	Ver lista de notas	Permite al usuario ver una lista de todas sus notas.
CU10	Agregar Nota	Permite al usuario crear una nueva nota.
CU11	Ver Nota	Permite al usuario ver una nota específica.
CU12	Editar Nota	Permite al usuario modificar una nota existente.
CU13	Eliminar Nota	Permite al usuario eliminar una nota existente.
CU14	Archivar Nota	Permite al usuario mover una nota a un archivo para su almacenamiento posterior.
CU15	Ayuda Notas	Permite al usuario acceder a información de ayuda sobre la gestión de notas.

Nota: Elaboración Propia

F.4 Gestión de Tareas

Imagen 63: Casos de Uso Gestión de Tareas

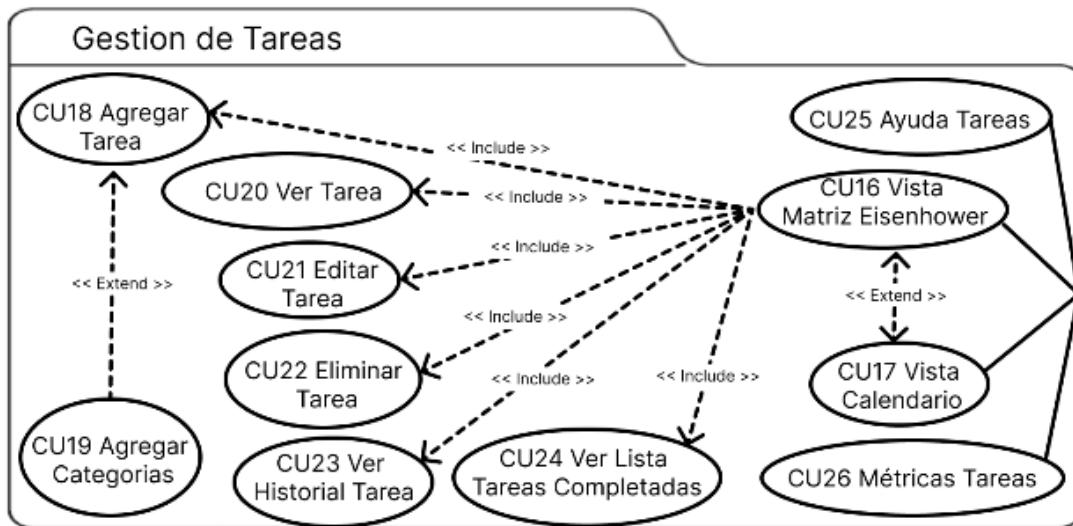


Tabla 31: Casos de Uso Gestión de Tareas

Casos de Uso para Gestión de Tareas

Id CU	Nombre CU	Descripción
CU16	Vista Matriz Eisenhower	Permite al usuario visualizar sus tareas en la matriz de Eisenhower para clasificarlas por importancia y urgencia.
CU17	Vista Calendario	Permite al usuario visualizar sus tareas en un calendario para tener una vista general de sus compromisos en el tiempo.
CU18	Agregar Tarea	Permite al usuario crear una nueva tarea.
CU19	Agregar Categorías	Permite al usuario crear y administrar categorías para organizar sus tareas.
CU20	Ver Tarea	Permite al usuario visualizar los detalles de una tarea específica.
CU21	Editar Tarea	Permite al usuario modificar la información de una tarea existente.
CU22	Eliminar Tarea	Permite al usuario eliminar una tarea existente.
CU23	Ver Historial Tarea	Permite al usuario visualizar el historial de modificaciones de una tarea específica.
CU24	Ver Lista Tareas Completadas	Permite al usuario visualizar una lista de todas las tareas que ha completado.
CU25	Ayuda Tareas	Permite al usuario acceder a información de ayuda sobre la gestión de tareas.
CU26	Métricas Tareas	Permite al usuario visualizar métricas sobre su rendimiento en la gestión de tareas, como el porcentaje de tareas completadas a tiempo o la cantidad de tareas pendientes por categoría.

Nota: Elaboración Propia

F.5 Gestión de Sesiones Pomodoro

Imagen 64: Casos de Uso Gestión de Sesiones Pomodoro

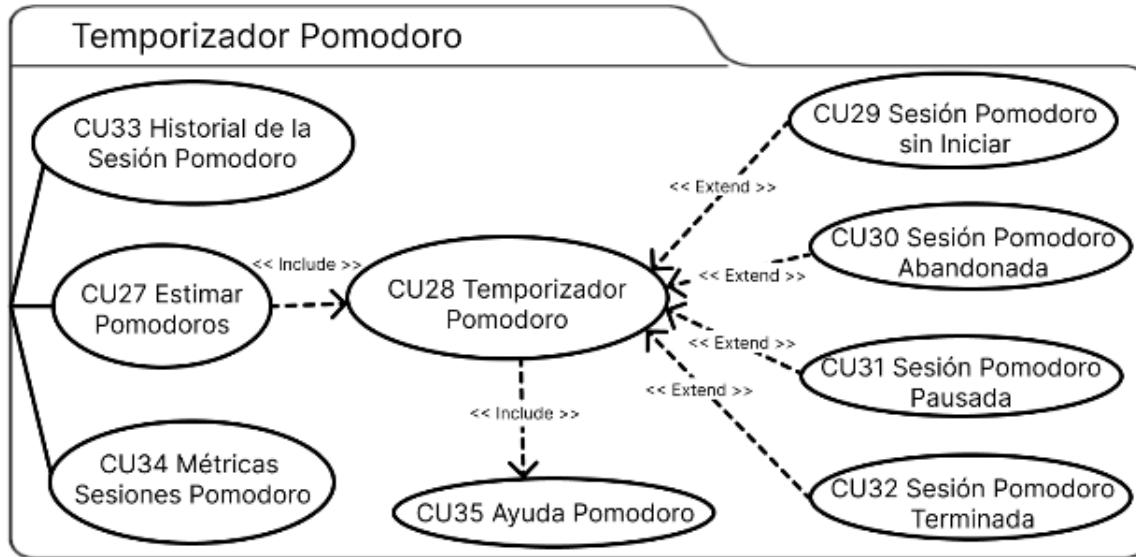


Tabla 32: Casos de Uso Sesión Pomodoro

Casos de Uso para Sesión Pomodoro

Id CU	Nombre CU	Descripción
CU27	Estimar Pomodoros	Permite al usuario estimar la cantidad de Pomodoros que tomaría completar una tarea.
CU28	Temporizador Pomodoro	Permite al usuario reiniciar, iniciar, pausar, reanudar y detener un temporizador Pomodoro.
CU29	Sesión Pomodoro sin Iniciar	Define el comportamiento del sistema cuando el usuario no ha iniciado una sesión Pomodoro.
CU30	Sesión Pomodoro Abandonada	Define el comportamiento del sistema cuando el usuario abandona una sesión Pomodoro sin finalizarla.
CU31	Sesión Pomodoro Pausada	Define el comportamiento del sistema cuando el usuario pausa una sesión Pomodoro.
CU32	Sesión Pomodoro Terminada	Define el comportamiento del sistema cuando una sesión Pomodoro se completa.
CU33	Historial de la Sesión Pomodoro	Permite al usuario visualizar el historial de sus sesiones Pomodoro.
CU34	Métricas Sesiones Pomodoro	Permite al usuario visualizar métricas sobre su rendimiento en las sesiones Pomodoro.
CU35	Ayuda Pomodoro	Permite al usuario acceder a información de ayuda sobre la técnica Pomodoro y la herramienta de gestión de sesiones Pomodoro.

Nota: Elaboración Propia

G. Casos de Uso

G.1 CU1: Registrar Usuario

Propósito: Permitir a un estudiante registrar una cuenta en el sistema.

Actor: Estudiante.

Entradas: Las de IU2 Pantalla de Registro de Usuario

Salidas: Las de IU2 Pantalla de Registro de Usuario

Precondiciones: El estudiante no debe estar registrado en el sistema.

Postcondiciones: El estudiante tiene una cuenta registrada en el sistema.

Trayectoria principal:

1. El estudiante ingresa al sistema Tempus Victor
2. El estudiante presiona el botón (registrarse) del menú de opciones o el botón (Comienza Ahora) de la pantalla IU1 Pantalla de Bienvenida al sistema.
3. El Sistema despliega la IU2 Pantalla de Registro de Usuario.
4. El estudiante ingresa su nombre de usuario, su correo electrónico, contraseña y confirma su contraseña
5. El estudiante confirma la operación presionando el botón (confirmar registro)
6. El sistema verifica todos los campos de texto del formulario con base en la BR1 todos los campos de entrada deben estar llenos. [Trayectoria A]
7. El Sistema verifica el campo de correo electrónico con base en la BR2 validar correo electrónico. [Trayectoria B]
8. El sistema verifica los campos de las contraseñas con base en la BR3 validar contraseña. [Trayectoria c]
9. El sistema crea una nueva cuenta con los valores proporcionados, guardando en la base de datos la nueva cuenta y muestra el MSG- “Tu registro fue exitoso!”.
10. El sistema despliega la pantalla IU3 Pantalla de Inicio de sesión

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: No todos los campos de entrada están llenos

A1: El Sistema muestra el Mensaje MSG- ‘’Por favor ingresa un nombre de usuario’ si este campo está vacío

A2: El Sistema muestra el Mensaje MSG- “Por favor ingresa un correo electrónico” si este campo está vacío

A3: El Sistema muestra el Mensaje MSG- “Por favor ingresa una contraseña” si este campo está vacío

A4: El Sistema muestra el Mensaje MSG- “Por favor confirma la contraseña” si este campo está vacío

A2: El estudiante continua en el paso 4 del CU1

---- *Fin de la trayectoria*

Trayectoria alternativa B:

Condición: *El correo electrónico ya está registrado en el sistema*

B1: El Sistema muestra el Mensaje MSG- “ Registro Incorrecto ¡El correo electrónico ya está en uso! ”

B2: El estudiante continua en el paso 4 del CU1

---- *Fin de la trayectoria*

Trayectoria alternativa C:

Condición: *Las contraseñas no coinciden una con la otra*

C1: El Sistema muestra el Mensaje MSG- “ Las contraseñas no coinciden ”

C2: El estudiante continua en el paso 4 del CU1

--- *Fin de la trayectoria*

G.2 CU2: Iniciar Sesión

Propósito: Permitir a un estudiante acceder al sistema iniciando sesión en su cuenta.

Actor: Estudiante.

Entradas: Las de IU3 Pantalla de Inicio de Sesión

Salidas: Las de IU3 Pantalla de Inicio de Sesión

Precondiciones: El estudiante debe estar registrado en el sistema.

Postcondiciones: El estudiante accede al sistema correctamente.

Trayectoria principal:

1. El estudiante ingresa al sistema Tempus Victor
2. El estudiante presiona el botón (Iniciar sesión) del menú de opciones o se registra en el sistema por medio del CU1 Registrar Usuario

3. El Sistema despliega la IU3 Pantalla de Inicio de Sesión
4. El estudiante ingresa su correo electrónico y contraseña.
5. El estudiante confirma la operación presionando el botón (Acceder)
6. El sistema verifica los campos de correo y contraseña con base en la BR4 para iniciar sesión se requiere el correo y contraseña [Trayectoria A]
7. El Sistema verifica el campo de correo electrónico con base en la BR5 validar correo electrónico. [Trayectoria B]
8. El sistema verifica el campo de la contraseña con base en la BR6 validar contraseña. [Trayectoria c]
9. El estudiante inicia sesión correctamente en su cuenta dentro del sistema
10. El sistema despliega la pantalla IU4 Vista de Tareas Eisenhower

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: No todos los campos de entrada están llenos

A1: El Sistema muestra el Mensaje MSG- “Por favor ingresa un correo electrónico” si este campo está vacío

A2: El Sistema muestra el Mensaje MSG- “Por favor ingresa una contraseña” si este campo está vacío

A3: El estudiante continua en el paso 4 del CU2

---- *Fin de la trayectoria*

Trayectoria alternativa B:

Condición: El correo electrónico ingresado no corresponde a ningún usuario existente

B1: El Sistema muestra el Mensaje MSG- “¡El usuario no existe! ”

B2: El estudiante continua en el paso 4 del CU2

---- *Fin de la trayectoria*

Trayectoria alternativa C:

Condición: Las contraseñas no es correcta para el correo electrónico ingresado

C1: El Sistema muestra el Mensaje MSG- “¡Contraseña Incorrecta! ”

C2: El estudiante continua en el paso 4 del CU2

--- *Fin de la trayectoria*

G.3 CU3: Cerrar Sesión

Propósito: Permitir a un estudiante cerrar sesión en su cuenta y salir del sistema.

Actor: Estudiante.

Entradas: Las de IU42 Modal para Cerrar sesión.

Salidas: Las de IU42 Modal para Cerrar sesión.

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante cierra sesión en su cuenta y sale del sistema correctamente.

Trayectoria principal:

1. El estudiante presiona el botón (Cuenta) de la barra del menú de opciones
 2. El sistema despliega la IU41 Pantalla de Cuenta
 3. El estudiante presiona el botón (Cerrar sesión) de la IU41 Pantalla de Cuenta
 4. El Sistema despliega el IU42 Modal para Cerrar sesión
 5. El estudiante presiona el botón (Regresar) del IU42 Modal para Cerrar sesión.
[Trayectoria A]
 6. El estudiante presiona el botón (Cerrar sesión) del IU42 Modal para Cerrar sesión.
 7. El sistema cierra la sesión del usuario y muestra el MSG- “Saliste del sistema”
 8. El sistema despliega la IU1 Pantalla de Bienvenida al Sistema
- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no cerrar sesión*

A1: El Sistema continua en el paso 2 del CU3

---- *Fin de la trayectoria*

G.4 CU4: Ver Cuenta

Propósito: Permitir a un estudiante ver la información de su cuenta

Actor: Estudiante.

Entradas: Las de IU41 Pantalla de Cuenta

Salidas: Las de IU41 Pantalla de Cuenta

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante visualiza la información de su cuenta.

Trayectoria principal:

1. El estudiante presiona el botón (Cuenta) de la barra del menú de opciones
2. El sistema despliega la IU41 Pantalla de Cuenta

---- *Fin del caso de uso*

G.5 CU5: Editar Cuenta

Propósito: Permitir a un estudiante editar su nombre de usuario y su foto de perfil.

Actor: Estudiante.

Entradas: Las de IU43 Pantalla para Editar Cuenta

Salidas: Las de IU43 Pantalla para Editar Cuenta

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante actualiza su nombre de usuario y su foto de perfil.

Trayectoria principal:

1. El estudiante presiona el botón (Cuenta) de la barra del menú de opciones
2. El sistema despliega la IU41 Pantalla de Cuenta
3. El estudiante presiona el botón (Editar) de la IU41 Pantalla de Cuenta
4. El Sistema despliega la IU43 Pantalla para Editar Cuenta
5. El estudiante ingresa el nuevo nombre de Usuario, y confirma el nombre de usuario
6. El estudiante presiona el selector de imagen y sube la imagen que desea para su perfil.
7. El estudiante presiona el botón (Cancelar) de la IU43 Pantalla para Editar Cuenta.
[Trayectoria A]
8. El estudiante presiona el botón (Aceptar) de la IU43 Pantalla para Editar Cuenta.
9. El sistema verifica que el campo de confirmar nombre de usuario no este vacío.
[Trayectoria B]
10. El sistema actualiza la imagen de perfil y el nombre de usuario de la cuenta.
11. El sistema muestra el MSG- “Se actualizo la cuenta”
12. El sistema despliega la IU41 Pantalla de Cuenta

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no editar la cuenta*

A1: El Sistema muestra el Mensaje MSG- “No se Actualizo la cuenta”

A3: El Sistema continua en el paso 2 del CU5

---- *Fin de la trayectoria*

Trayectoria alternativa B:

Condición: *El campo de confirmar nombre de usuario está vacío*

B1: El Sistema muestra el Mensaje MSG- “ ¡Por favor confirma el nuevo nombre de usuario! ”

B2: El estudiante continua en el paso 5 del CU5

---- *Fin de la trayectoria*

G.6 CU6: Cambiar Contraseña

Propósito: Permitir a un estudiante cambiar su contraseña

Actor: Estudiante.

Entradas: Las de IU44 Modal para Cambiar Contraseña

Salidas: Las de IU44 Modal para Cambiar Contraseña

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante actualiza cambia su contraseña correctamente.

Trayectoria principal:

1. El estudiante presiona el botón (Cuenta) de la barra del menú de opciones
2. El sistema despliega la IU41 Pantalla de Cuenta
3. El estudiante presiona el botón (Cambiar Contraseña) de la IU41 Pantalla de Cuenta
4. El Sistema despliega el IU44 Modal para Cambiar Contraseña
5. El estudiante ingresa la nueva contraseña y confirma la nueva contraseña.
6. El estudiante presiona el botón (Regresar) del IU44 Modal para Cambiar Contraseña.
[Trayectoria A]
7. El estudiante presiona el botón (Guardar) del IU44 Modal para Cambiar Contraseña
8. El sistema verifica que los campos de contraseñas no estén vacíos. [Trayectoria B]
9. El sistema actualiza la contraseña de la cuenta del usuario
10. El sistema muestra el MSG- “Se actualizo la contraseña correctamente”
11. El sistema cierra la sesión del usuario y muestra el MSG- “Saliste del sistema”
12. El sistema despliega la IU1 Pantalla de Bienvenida al Sistema

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no cambiar la contraseña*

A1: El Sistema muestra el Mensaje MSG- “Acción cancelada”

A3: El Sistema continua en el paso 2 del CU6

---- *Fin de la trayectoria*

Trayectoria alternativa B:

Condición: Algún campo de contraseña está vacío

B1: El Sistema muestra el Mensaje MSG- “Por favor ingresa una contraseña” si este campo está vacío

B2: El Sistema muestra el Mensaje MSG- “Por favor confirma la contraseña” si este campo está vacío

B3: El estudiante continua en el paso 5 del CU6

---- *Fin de la trayectoria*

G.7 CU7: Eliminar Cuenta

Propósito: Permitir a un estudiante eliminar definitivamente su cuenta

Actor: Estudiante.

Entradas: Las de IU45 Modal para Eliminar Cuenta, IU46 Modal para Confirmar Eliminar Cuenta

Salidas: Las de IU45 Modal para Eliminar Cuenta, IU46 Modal para Confirmar Eliminar Cuenta

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante elimina su cuenta completamente, junto con todos los datos relacionados a ella.

Trayectoria principal:

1. El estudiante presiona el botón (Cuenta) de la barra del menú de opciones
2. El sistema despliega la IU41 Pantalla de Cuenta
3. El estudiante presiona el botón (Eliminar Cuenta) de la IU41 Pantalla de Cuenta
4. El Sistema despliega el IU45 Modal para Eliminar Cuenta
5. El estudiante presiona el botón (Regresar) del IU45 Modal para Eliminar Cuenta.
[Trayectoria A]
6. El estudiante presiona el botón (Eliminar Cuenta) del IU45 Modal para Eliminar Cuenta
7. El Sistema despliega el IU46 Modal para Confirmar Eliminar Cuenta
8. El estudiante presiona el botón (Regresar) del IU46 Modal para Confirmar Eliminar Cuenta. [Trayectoria A]

9. El estudiante presiona el botón (Confirmar Eliminar Cuenta) del IU46 Modal para Confirmar Eliminar Cuenta
10. El sistema muestra el MSG- “Se Elimino la cuenta correctamente”
11. El sistema cierra la sesión del usuario, elimina la cuenta y muestra el MSG- “Saliste del sistema”
12. El sistema despliega la IU1 Pantalla de Bienvenida al Sistema

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no eliminar la cuenta*

A1: El Sistema continua en el paso 2 del CU7

---- *Fin de la trayectoria*

G.8 CU8: Personalizar Cuenta

Propósito: Permitir a un estudiante personalizar su cuenta

Actor: Estudiante.

Entradas: Las de IU47 Pantalla de Configuración del Sistema.

Salidas: Las de IU47 Pantalla de Configuración del Sistema.

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante personaliza su cuenta con los valores y parámetros que elija.

Trayectoria principal:

1. El estudiante presiona el botón (Configuración) de la barra del menú de opciones
2. El sistema despliega la IU47 Pantalla de Configuración del Sistema.
3. El estudiante presiona el botón (Regresar). [Trayectoria A]
4. El estudiante presiona el botón (Ayuda). [Trayectoria B]
5. El estudiante cambia el tiempo de trabajo de la sesión Pomodoro
6. El estudiante cambia el tiempo de descanso corto de la sesión Pomodoro
7. El estudiante cambia el tiempo de descanso largo de la sesión Pomodoro
8. El estudiante elige la imagen de fondo para la sesión abandonada y sin iniciar.
9. El estudiante elige la imagen de fondo para la sesión en tiempo de descanso
10. El estudiante elige la imagen de fondo para la sesión en tiempo de trabajo
11. El estudiante selecciona el sonido para la alarma del temporizador Pomodoro.

12. El estudiante selecciona el sonido de fondo del temporizador Pomodoro.
13. El estudiante presiona el botón (Resetear Valores). [Trayectoria C]
14. El estudiante presiona el botón (Guardar Configuración).
15. El sistema despliega el IU50 Modal para Guardar Valores de Configuración.
16. El estudiante presiona el botón (Cancelar). [Trayectoria D]
17. El estudiante presiona el botón (Guardar Cambios).
18. El sistema muestra el MSG- “Configuración Guardada Correctamente”, guarda los valores de configuración actuales.
19. El sistema despliega la IU47 Pantalla de Configuración del Sistema.

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante decide salir de la pantalla de configuración

A1: El Sistema muestra el Mensaje MSG- “¡Salió De Configuración del Sistema!”

A3: El sistema despliega la IU4 Vista de Tareas Eisenhower

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante quiere obtener ayuda sobre la pantalla Configuración

B1: El sistema despliega el IU48 Modal de Ayuda sobre la Configuración

B2: El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda

B3: El estudiante presiona el botón de ‘X’ para salir del modal

B4: El estudiante continua en el paso 2 del CU8

---- *Fin de la trayectoria*

Trayectoria alternativa C:

Condición: El estudiante quiere resetear los valores actuales por los de defecto

B1: El sistema despliega el U49 Modal para Resetear Valores de Configuración

B2: Si el estudiante presiona el botón cancelar, el sistema muestra el MSG- “Acción cancelada”

B3: Si el estudiante presiona el botón aceptar, el sistema muestra el MSG – “Valores Reseteados Correctamente” y resetea los valores actuales por los de defecto.

B4: El estudiante presiona el botón de ‘X’ para salir del modal

B5: El estudiante continua en el paso 2 del CU8

---- *Fin de la trayectoria*

Trayectoria alternativa D:

Condición: *El estudiante decide no guardar la configuración actual*

D1: El sistema muestra el MSG- “Acción cancelada”

D1: El sistema continua en el paso 2 del CU8

---- *Fin de la trayectoria*

G.9 CU9: Ver lista de notas

Propósito: Permitir a un estudiante visualizar las notas de su cuenta en una lista

Actor: Estudiante.

Entradas: Las de IU32 Pantalla de Notas

Salidas: Las de IU32 Pantalla de Notas

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente.

Postcondiciones: El estudiante visualiza las notas de su cuenta en una lista

Trayectoria principal:

1. El estudiante presiona el botón (Notas) de la barra del menú de opciones
2. El sistema despliega la IU32 Pantalla de Notas.
3. El sistema carga las notas de la cuenta que no estén archivadas y las lista en la pantalla.
4. El estudiante puede buscar una nota usando el filtro de notas, basándose en la BR23:
Filtrar Notas en tiempo real
5. El sistema filtra la lista de notas desplegando la o las notas que cumplan con lo ingresado en el filtro.
6. El estudiante presiona el botón (Ayuda). [Trayectoria A]
7. El estudiante presiona el botón (Notas Archivadas). [Trayectoria B]
8. El estudiante presiona el botón (Agregar). [Trayectoria C]
9. El estudiante selecciona una nota de la lista [Trayectoria D]

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide obtener ayuda sobre la pantalla de notas*

A1: El estudiante continua con el CU15 Ayuda Notas

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante decide ver las notas archivadas

B1: El sistema despliega la IU35 Pantalla de Notas Archivadas

B2: El estudiante puede buscar una nota usando el filtro de notas, basándose en la BR23:

Filtrar Notas en tiempo real

B3: El sistema filtra la lista de notas desplegando la o las notas que cumplan con lo ingresado en el filtro.

B4: El estudiante selecciona una nota y presiona el botón de desarchivar nota

B5: El sistema cambia el estado de la nota por no archivada y continua con el paso 2 del CU9

---- Fin de la trayectoria

Trayectoria alternativa C:

Condición: El estudiante decide agregar una nueva nota

C1: El estudiante continua con el CU10 Agregar Nota

---- Fin del caso de uso

Trayectoria alternativa D:

Condición: El estudiante selecciona una nota de la lista de notas

D1: El sistema resalta la nota seleccionada y habilita los nuevos botones para interactuar con ella: Ver, Editar, Eliminar, Archivar Nota.

D2: El estudiante presiona el botón Ver y continua con el CU11 Ver Nota

D3: El estudiante presiona el botón Editar y continua con el CU12 Editar Nota

D4: El estudiante presiona el botón Eliminar y continua con el CU13 Eliminar Nota

D5: El estudiante presiona el botón Archivar y continua con el CU14 Archivar Nota

---- Fin del caso de uso

G.10 CU10: Agregar Nota

Propósito: Permitir a un estudiante agregar una nueva nota a su cuenta

Actor: Estudiante.

Entradas: Las de IU36 Pantalla para Agregar Nota

Salidas: Las de IU36 Pantalla para Agregar Nota

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón de agregar nota de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante crea una nueva nota

Trayectoria principal:

1. El sistema despliega la IU36 Pantalla para Agregar Nota
2. El estudiante ingresa el título de nota y de descripción de nota
3. El estudiante presiona el botón (Cancelar). [Trayectoria A]
4. El estudiante confirma la operación presionando el botón (Aceptar).
5. El sistema verifica el título de la nota en base a la BR14: validación de nota
[Trayectoria B]
6. El sistema muestra el MSG- “¡Nota creada exitosamente!”
7. El sistema despliega la U32 Pantalla de Notas

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no agregar una nueva nota*

A1: El sistema muestra el MSG- “¡No se creó la Nota!”

A2: El sistema despliega la pantalla IU32 Pantalla de Notas

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El sistema valido el titulo y no paso la validación*

B1: Si el título de nota está vacío, el sistema muestra el MSG- “Por favor ingresa un título”

B2: Si el título de nota es menor a 3 caracteres de largo, el sistema muestra el MSG- “El título debe tener al menos 3 caracteres”

B3: El estudiante continua con el paso 2 del CU10

---- *Fin de la trayectoria*

G.11 CU11: Ver Nota

Propósito: Permitir a un estudiante ver una nota específica

Actor: Estudiante.

Entradas: Las de IU37 Pantalla para Ver Nota

Salidas: Las de IU37 Pantalla para Ver Nota

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber seleccionado una nota de la lista de notas y el estudiante debe haber presionado el botón de ver nota de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante visualiza el contenido de la nota completa

Trayectoria principal:

1. El sistema despliega la IU37 Pantalla para Ver Nota
2. El estudiante visualiza la información de la nota completa
3. El estudiante presiona el botón (Regresar).
4. El sistema despliega la U32 Pantalla de Notas

---- *Fin del caso de uso*

G.12 CU12: Editar Nota

Propósito: Permitir a un estudiante editar una nota específica

Actor: Estudiante.

Entradas: Las de IU38 Pantalla para Editar Nota

Salidas: Las de IU38 Pantalla para Editar Nota

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber seleccionado una nota de la lista de notas y el estudiante debe haber presionado el botón de editar nota de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante edito a la nota seleccionada

Trayectoria principal:

1. El sistema despliega la IU38 Pantalla para Editar Nota
1. El estudiante ingresa el título de nota y de descripción de nota
2. El estudiante presiona el botón (Cancelar). [Trayectoria A]
3. El estudiante confirma la operación presionando el botón (Aceptar).
4. El sistema verifica el título de la nota en base a la BR14: validación de nota [Trayectoria B]
5. El sistema muestra el MSG- “¡Nota editada exitosamente!”
6. El sistema despliega la U32 Pantalla de Notas

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante decide no editar la nota

A1: El sistema muestra el MSG- “¡No se editó la Nota!”

A2: El sistema despliega la pantalla IU32 Pantalla de Notas

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El sistema valido el titulo y no paso la validación*

B1: Si el título de nota está vacío, el sistema muestra el MSG- “Por favor ingresa un título”

B2: Si el título de nota es menor a 3 caracteres de largo, el sistema muestra el MSG- “El titulo debe tener al menos 3 caracteres”

B3: El estudiante continua con el paso 2 del CU10

---- *Fin de la trayectoria*

G.13 CU13: Eliminar Nota

Propósito: Permitir a un estudiante eliminar una nota específica

Actor: Estudiante.

Entradas: Las de IU39 Pantalla para Eliminar Nota

Salidas: Las de IU39 Pantalla para Eliminar Nota

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber seleccionado una nota de la lista de notas y el estudiante debe haber presionado el botón de eliminar nota de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante elimina a la nota seleccionada

Trayectoria principal:

1. El sistema despliega la IU39 Pantalla para Eliminar Nota
2. El estudiante presiona el botón (Cancelar). [Trayectoria A]
3. El estudiante confirma la operación presionando el botón (Eliminar).
4. El sistema muestra el MSG- “¡Nota Eliminada Correctamente!”
5. El sistema despliega la U32 Pantalla de Notas

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no eliminar la nota*

A1: El sistema muestra el MSG- “¡No se eliminó la Nota!”

A2: El sistema despliega la pantalla IU32 Pantalla de Notas

---- *Fin del caso de uso*

G.14 CU14: Archivar Nota

Propósito: Permitir a un estudiante archivar una nota específica

Actor: Estudiante.

Entradas: Las de IU39 Pantalla para Eliminar Nota

Salidas: Las de IU39 Pantalla para Eliminar Nota

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber seleccionado una nota de la lista de notas y el estudiante debe haber presionado el botón de archivar nota de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante archiva a la nota seleccionada.

Trayectoria principal:

1. El sistema despliega el IU40 Modal para Archivar una Nota
2. El estudiante presiona el botón (Cancelar). [Trayectoria A]
3. El estudiante confirma la operación presionando el botón (Archivar Nota).
4. El sistema despliega la IU32 Pantalla de Notas actualizando la lista de notas

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante decide no archivar la nota*

A1: El sistema muestra el MSG- “Acción cancelada”

A2: El sistema despliega la pantalla IU32 Pantalla de Notas

---- *Fin del caso de uso*

G.15 CU15: Ayuda Notas

Propósito: Permitir a un estudiante obtener ayuda sobre la pantalla de Notas

Actor: Estudiante.

Entradas: Las del IU34 Modal de Ayuda sobre las Notas

Salidas: Las del IU34 Modal de Ayuda sobre las Notas

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón de ayuda de notas de la IU32 Pantalla de Notas.

Postcondiciones: El estudiante sabe que se hace en la pantalla de notas

Trayectoria principal:

1. El sistema despliega el IU34 Modal de Ayuda sobre las Notas

2. El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda
3. El estudiante presiona el botón de ‘X’ para salir del modal
4. El sistema despliega la U32 Pantalla de Notas

---- *Fin del caso de uso*

G.16 CU16: Vista Matriz Eisenhower

Propósito: Permitir a un estudiante ver sus tareas en la vista de matriz Eisenhower

Actor: Estudiante.

Entradas: Las de IU4 Vista de Tareas Eisenhower

Salidas: Las de IU4 Vista de Tareas Eisenhower

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente,

Postcondiciones: El estudiante gestiona sus tareas por medio de la vista de Eisenhower

Trayectoria principal:

1. El estudiante presiona el botón (Tareas) de la barra de menú de opciones del sistema.
2. El sistema despliega el IU4 Vista de Tareas Eisenhower
3. El sistema carga las tareas de la cuenta que estén sin completar y las lista en la pantalla
4. El estudiante busca una tarea usando el filtro de tareas
5. El sistema filtra la lista de tareas desplegando la o las notas que cumplan con lo ingresado en el filtro.
6. El estudiante utiliza el selector de categoría para filtrar por categoría
7. El sistema filtra la lista de tareas desplegando la o las notas que pertenezcan a la categoría seleccionada
8. El estudiante presiona el botón (Ir a Calendario). [Trayectoria A]
9. El estudiante presiona el botón (Ayuda). [Trayectoria B]
10. El estudiante presiona el botón (Tareas Completadas). [Trayectoria C]
11. El estudiante presiona el botón (Agregar). [Trayectoria D]
12. El estudiante selecciona una tarea de la lista [Trayectoria E]
13. El estudiante arrastra la tarea seleccionada y la cambia de cuadrante
14. El sistema actualiza la importancia y urgencia basándose en la BR37: Actualizar importancia y urgencia

15. El sistema muestra el MSG–“Se actualizo la Tarea Correctamente”

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante presiona el botón ir a Calendario

C1: El sistema despliega la pantalla IU8 Vista de Tareas Calendario

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante presiona el botón Ayuda

B1: El estudiante continua con el CU25 Ayuda Tareas

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: El estudiante requiere ver la lista de tareas completadas

C1: El estudiante continua con el CU24 Ver lista de Tareas Completadas

---- *Fin del caso de uso*

Trayectoria alternativa D:

Condición: El estudiante requiere agregar una nueva tarea

D1: El estudiante continua con el CU18 Agregar Tarea

---- *Fin del caso de uso*

Trayectoria alternativa E:

Condición: El estudiante selecciona una tarea

E1: El sistema habilita los botones para gestionar la tarea, Ver, Editar, Eliminar, Historial,

Pomodoro.

E2: Si el estudiante presiona el botón Ver, continua con el CU20 Ver Tarea

E3: Si el estudiante presiona el botón Editar, continua con el CU21 Editar Tarea

E4: Si el estudiante presiona el botón Eliminar, continua con el CU22 Eliminar Tarea

E5: Si el estudiante presiona el botón Historial, continua con el CU23 Ver Historial Tarea

E6: Si el estudiante presiona el botón Pomodoro, continua con el CU27 Estimar

Pomodoros

---- *Fin del caso de uso*

G.17 CU17: Vista Calendario

Propósito: Permitir a un estudiante ver sus tareas en la vista de Calendario

Actor: Estudiante.

Entradas: Las de IU8 Vista de Tareas Calendario

Salidas: Las de IU8 Vista de Tareas Calendario

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón Ir a Calendario de la IU4 Vista de Tareas Eisenhower

Postcondiciones: El estudiante gestiona sus tareas por medio de la vista de Calendario

Trayectoria principal:

1. El sistema despliega el IU8 Vista de Tareas Calendario
2. El sistema carga las tareas de la cuenta que estén sin completar, lista las tareas sin fecha límite en una lista, y muestra las que tienen fecha límite en el calendario.
3. El estudiante busca una tarea usando el filtro de tareas
4. El sistema filtra la lista de tareas desplegando la o las notas que cumplan con lo ingresado en el filtro.
5. El estudiante utiliza el selector de categoría para filtrar por categoría
6. El sistema filtra la lista de tareas desplegando la o las notas que pertenezcan a la categoría seleccionada
7. El estudiante presiona el botón (Ir a Calendario). [Trayectoria A]
8. El estudiante presiona el botón (Ayuda). [Trayectoria B]
9. El estudiante presiona el botón (Tareas Completadas). [Trayectoria C]
10. El estudiante presiona el botón (Aregar). [Trayectoria D]
11. El estudiante selecciona una tarea de la lista [Trayectoria E]
12. El estudiante arrastra la tarea seleccionada y la cambia de cuadrante
13. El sistema actualiza la importancia y urgencia basándose en la BR37: Actualizar importancia y urgencia
14. El sistema muestra el MSG – “Se actualizo la Tarea Correctamente”

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante presiona el botón ir a Calendario*

A1: El sistema despliega la pantalla IU8 Vista de Tareas Calendario

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El estudiante presiona el botón Ayuda*

B1: El estudiante continua con el CU25 Ayuda Tareas

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: El estudiante requiere ver la lista de tareas completadas

C1: El estudiante continua con el CU24 Ver lista de Tareas Completadas

---- *Fin del caso de uso*

Trayectoria alternativa D:

Condición: El estudiante requiere agregar una nueva tarea

D1: El estudiante continua con el CU18 Agregar Tarea

---- *Fin del caso de uso*

Trayectoria alternativa E:

Condición: El estudiante selecciona una tarea

E1: El sistema habilita los botones para gestionar la tarea, Ver, Editar, Eliminar, Historial, Pomodoro.

E2: Si el estudiante presiona el botón Ver, continua con el CU20 Ver Tarea

E3: Si el estudiante presiona el botón Editar, continua con el CU21 Editar Tarea

E4: Si el estudiante presiona el botón Eliminar, continua con el CU22 Eliminar Tarea

E5: Si el estudiante presiona el botón Historial, continua con el CU23 Ver Historial Tarea

E6: Si el estudiante presiona el botón Pomodoro, continua con el CU27 Estimar Pomodoros

---- *Fin del caso de uso*

G.18 CU18: Agregar Tarea

Propósito: Permitir a un estudiante agregar una tarea nueva por completar

Actor: Estudiante.

Entradas: Las de IU14 Pantalla para Agregar Tarea

Salidas: Las de IU14 Pantalla para Agregar Tarea

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón agregar Tarea en alguna de las vistas de tareas.

Postcondiciones: El estudiante tiene una tarea más por completar

Trayectoria principal:

1. El sistema despliega la IU14 Pantalla para Agregar Tarea
2. El estudiante ingresa título de la tarea, descripción de la tarea, comentarios de la tarea, fecha y hora límite, selecciona una categoría, ingresa la importancia y la urgencia.

3. El estudiante presiona el botón (+) de agregar nueva categoría. [Trayectoria A].
4. El estudiante presiona el botón (cancelar). [Trayectoria B].
5. El estudiante presiona el botón (aceptar).
6. El sistema verifica los campos de texto del formulario con base en la BR25 validación de nueva tarea. [Trayectoria C]
7. El sistema muestra el MSG- “¡Tarea creada exitosamente!”
8. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante requiere agregar una nueva categoría a las ya existentes

A1: El sistema despliega el IU15 Modal para Agregar Categoría

A2: El estudiante continua en el CU19 Agregar Categorías

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante presiona el botón de cancelar la acción

B1: El sistema muestra el MSG- “¡No se creó la Tarea!”

B2: El sistema despliega la vista de tareas anterior.

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: Los campos de entrada no fueron validados

C1: Si el título de la tarea está vacío, el sistema muestra el MSG- “Por favor ingresa un Título”

C2: Si el título de la tarea contiene menos de 4 caracteres, el sistema muestra el MSG- “El título debe contener al menos 4 caracteres”

C3: Si la descripción de la tarea está vacía, el sistema muestra el MSG- “Por favor ingresa una descripción”

C4: Si la descripción de la tarea contiene menos de 4 caracteres, el sistema muestra el MSG- “La descripción debe contener al menos 4 caracteres”

C5: El estudiante continua en el paso 2 del CU 18

---- *Fin del caso de uso*

G.19 CU19: Agregar Categorías

Propósito: Permitir a un estudiante agregar una nueva categoría a su cuenta

Actor: Estudiante.

Entradas: Las del IU15 Modal para Agregar Categoría

Salidas: Las del IU15 Modal para Agregar Categoría

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber presionado el botón agregar categoría, de la Imagen IU14 Pantalla para Agregar Tarea

Postcondiciones: El estudiante tiene una categoría más en su cuenta

Trayectoria principal:

1. El sistema despliega el IU15 Modal para Agregar Categoría
2. El estudiante ingresa la nueva categoría en el campo de entrada de texto
3. El estudiante presiona el botón (Regresar). [Trayectoria A]
4. El estudiante confirma la operación presionando el botón (Guardar)
5. El sistema verifica que el campo de nueva categoría no este vacío. [Trayectoria B]
6. El sistema muestra el MSG- “Se agrego una nueva categoría”
7. El sistema despliega la pantalla IU14 Pantalla para Agregar Tarea

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante ya no requiere agregar una nueva categoría*

A1: El sistema despliega el MSG- “Acción Cancelada”

A2: El estudiante continua en el CU18 Agregar Tarea

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El campo de entrada de nueva categoría está vacío*

B1: El sistema muestra el MSG- “Agrega una categoría”

B2: El estudiante continua con el paso 2 del CU19

---- *Fin de la trayectoria*

G.20 CU20: Ver Tarea

Propósito: Permitir a un estudiante ver el contenido de una tarea específica

Actor: Estudiante.

Entradas: Las de la IU16 Pantalla para Ver Tarea

Salidas: Las de la IU16 Pantalla para Ver Tarea

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea en cualquier vista de tareas, el estudiante debe de haber presionado el botón ver tarea de la vista de tareas.

Postcondiciones: El estudiante visualizo todo el contenido de la nota completa

Trayectoria principal:

1. El sistema despliega la IU16 Pantalla para Ver Tarea
2. El estudiante visualiza la información de la tarea seleccionada
3. El estudiante presiona el botón (Regresar).
4. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.21 CU21: Editar Tarea

Propósito: Permitir a un estudiante editar una tarea específica

Actor: Estudiante.

Entradas: Las de la IU17 Pantalla para Editar Tarea

Salidas: Las de la IU17 Pantalla para Editar Tarea

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de encontrarse en alguna vista de tareas, el estudiante debe haber seleccionado una tarea, el estudiante debe haber presionado el botón editar Tarea.

Postcondiciones: El estudiante tiene la tarea seleccionada con datos actualizados.

Trayectoria principal:

1. El sistema despliega la IU17 Pantalla para Editar Tarea.
2. El estudiante modifica el título de la tarea, descripción de la tarea, comentarios de la tarea, fecha y hora límite, selecciona una categoría, ingresa la importancia y la urgencia.
3. El estudiante presiona el botón (+) de agregar nueva categoría. [Trayectoria A].
4. El estudiante presiona el botón (cancelar). [Trayectoria B].
5. El estudiante presiona el botón (aceptar).

6. El sistema verifica los campos de texto del formulario con base en la BR25 validación de nueva tarea. [Trayectoria C]
7. El sistema muestra el MSG- “¡Tarea actualizada exitosamente!”
8. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante requiere agregar una nueva categoría a las ya existentes

A1: El sistema despliega el IU15 Modal para Agregar Categoría

A2: El estudiante continua en el CU19 Agregar Categorías

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante presiona el botón de cancelar la acción

B1: El sistema muestra el MSG- “¡No se actualizo la Tarea!”

B2: El sistema despliega la vista de tareas anterior.

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: Los campos de entrada no fueron validados

C1: Si el título de la tarea está vacío, el sistema muestra el MSG- “Por favor ingresa un Título”

C2: Si el título de la tarea contiene menos de 4 caracteres, el sistema muestra el MSG- “El titulo debe contener al menos 4 caracteres”

C3: Si la descripción de la tarea está vacía, el sistema muestra el MSG- “Por favor ingresa una descripción”

C4: Si la descripción de la tarea contiene menos de 4 caracteres, el sistema muestra el MSG- “La descripción debe contener al menos 4 caracteres”

C5: El estudiante continua en el paso 2 del CU 21

---- *Fin de la trayectoria*

G.22 CU22: Eliminar Tarea

Propósito: Permitir a un estudiante eliminar una tarea específica

Actor: Estudiante.

Entradas: Las de la IU18 Pantalla para Eliminar Tarea

Salidas: Las de la IU18 Pantalla para Eliminar Tarea

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de encontrarse en alguna vista de tareas, el estudiante debe haber seleccionado una tarea, el estudiante debe haber presionado el botón Eliminar Tarea.

Postcondiciones: El estudiante tiene una tarea menos en su lista de tareas

Trayectoria principal:

1. El sistema despliega la IU18 Pantalla para Eliminar Tarea
2. El estudiante presiona el botón (cancelar). [Trayectoria A].
3. El estudiante confirma la operación presionando el botón (Eliminar)
4. El sistema elimina la tarea seleccionada
5. El sistema muestra el MSG- “Tarea eliminada Correctamente”
6. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante ya no requiere eliminar la tarea seleccionada*

A1: El sistema muestra el MSG- “No se eliminó la tarea”

A2: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.23 CU23: Ver Historial Tarea

Propósito: Permitir a un estudiante ver el historial de eventos de una tarea específica

Actor: Estudiante.

Entradas: Las de la IU19 Pantalla de Historial de Tarea.

Salidas: Las de la IU19 Pantalla de Historial de Tarea.

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de encontrarse en alguna vista de tareas, el estudiante debe haber seleccionado una tarea, el estudiante debe haber presionado el botón Historial

Postcondiciones: El estudiante conoce los eventos relacionados a la tarea seleccionada

Trayectoria principal:

1. El sistema despliega la IU19 Pantalla de Historial de Tarea.
2. El estudiante puede hacer scroll a la lista de eventos de la tarea
3. El estudiante hace scroll a la lista de eventos de la sesión Pomodoro
4. El estudiante presiona el botón (Regresar).
5. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.24 CU24: Ver Lista Tareas Completadas

Propósito: Permitir a un estudiante ver la lista de tareas completadas

Actor: Estudiante.

Entradas: Las de la IU13 Pantalla de Tareas Completadas

Salidas: Las de la IU13 Pantalla de Tareas Completadas

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de encontrarse en alguna vista de tareas, el estudiante debe haber presionado el botón Tareas Completadas

Postcondiciones: El estudiante conoce que tareas ya completo

Trayectoria principal:

1. El sistema despliega la IU13 Pantalla de Tareas Completadas
2. El usuario puede buscar por palabra en el filtro de tareas alguna en específico
3. El sistema filtra la lista de tareas completadas si existe una o varias tareas con las palabras ingresadas
4. El estudiante selecciona una tarea de la lista
5. El sistema resalta la tarea seleccionada y habilita el botón para cambiar el estado de la tarea a no completada.
6. El estudiante presiona el botón Regresar. [Trayectoria A]
7. El estudiante presiona el botón Cambiar Estado a No Completada.
8. El sistema muestra un modal de confirmación para cambiar el estado
9. El estudiante presiona el botón (cancelar) del modal. [Trayectoria B]
10. El estudiante presiona el botón (cambiar estado)
11. El sistema muestra el MSG- “¡Estado cambiado exitosamente!”
12. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante ya no requiere ver las tareas completadas

A1: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: El estudiante ya no requiere cambiar el estado de la tarea

B1: El estudiante continua en el paso 2 del CU 24

---- *Fin del caso de uso*

G.25 CU25: Ayuda Tareas

Propósito: Permitir a un estudiante obtener ayuda sobre alguna vista de tareas

Actor: Estudiante.

Entradas: Las del IU6 Modal de Ayuda Vista Eisenhower, y el IU11 Modal de Ayuda de la Vista Calendario

Salidas: Las del IU6 Modal de Ayuda Vista Eisenhower, y el IU11 Modal de Ayuda de la Vista Calendario

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe estar en alguna vista de tareas

Postcondiciones: El estudiante sabe que se hace en las vistas de tareas

Trayectoria principal:

1. El sistema despliega el IU6 Modal de Ayuda Vista Eisenhower si se presionó el botón de ayuda desde la pantalla IU4 Vista de Tareas Eisenhower
2. El sistema despliega el IU11 Modal de Ayuda de la Vista Calendario si se presionó el botón de ayuda desde la pantalla IU9 Seleccionar Dia vista Calendario
3. El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda
4. El estudiante presiona el botón de 'X' para salir del modal
5. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.26 CU26: Métricas Tareas

Propósito: Permitir a un estudiante visualizar las métricas de las tareas

Actor: Estudiante.

Entradas: Las de la IU51 Pantalla de Métricas de la Cuenta

Salidas: Las de la IU51 Pantalla de Métricas de la Cuenta

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón de Métricas de la barra de menú de opciones del sistema.

Postcondiciones: El estudiante conoce las métricas relacionadas con las tareas de su cuenta

Trayectoria principal:

1. El sistema despliega la IU51 Pantalla de Métricas de la Cuenta
2. El sistema calcula las métricas para Tareas
3. El sistema actualiza el diagrama de pastel sobre tareas
4. El estudiante lee las métricas para tareas
5. El estudiante presiona el botón (Regresar). [Trayectoria A]
6. El estudiante presiona el bote (Ayuda). [Trayectoria B]

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante ya no requiere ver las métricas de la cuenta*

A1: El sistema muestra el MSG- “¡Salió De Métricas del Sistema!”

A2: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El estudiante requiere obtener ayuda sobre la pantalla de métricas*

B1: El sistema despliega el IU52 Modal de Ayuda sobre las Métricas

B2: El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda

B3: El estudiante presiona el botón de ‘X’ para salir del modal

B4: La estudiante continua en el paso 1 del CU 26

---- *Fin del caso de uso*

G.27 CU27: Estimar Pomodoros

Propósito: Permitir a un estudiante estimar el tiempo para completar una tarea

Actor: Estudiante.

Entradas: Las de la IU20 Pantalla de Estimación de Sesión Pomodoro

Salidas: Las de la IU20 Pantalla de Estimación de Sesión Pomodoro

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber presionado el botón Pomodoro.

Postcondiciones: El estudiante estimo en Pomodoros el tiempo de completar una tarea

Trayectoria principal:

1. El sistema despliega la IU20 Pantalla de Estimación de Sesión Pomodoro
2. El sistema obtiene y despliega la información de la tarea
3. El estudiante lee la explicación sobre la técnica Pomodoro
4. El estudiante presiona el link “Mas Información... Aquí”. [Trayectoria A]
5. El estudiante presiona el botón (Regresar). [Trayectoria B]
6. El estudiante presiona el botón (Aregar Pomodoro) y agrega un Pomodoro a la estimación. [Trayectoria C]
7. El estudiante presiona el botón (Quitar Pomodoro) y quita un Pomodoro de la estimación. [Trayectoria D]
8. El estudiante presiona el botón (Empezar sesión).
9. El sistema despliega la U21 Pantalla de Temporizador Pomodoro

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante requiere más información sobre la técnica Pomodoro*

A1: El sistema lleva al estudiante a la página de Todoist donde explica a fondo la técnica Pomodoro

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El estudiante requiere salir de la estimación*

B1: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: *El estudiante quiere estimar la tarea con más de 4 Pomodoros*

C1: El sistema muestra el mensaje de error MSG- “¡La tarea parece ser grande, considere dividirla en tareas más pequeñas!”

C2: El estudiante continua en el paso 1 del CU27

---- Fin de la Trayectoria

Trayectoria alternativa D:

Condición: *El estudiante quiere estimar la tarea con menos de 1 Pomodoro*

D1: El sistema muestra el mensaje de advertencia MSG- “¡Primero Estima los Pomodoros!”

D2: El estudiante continua en el paso 1 del CU27

---- *Fin de la Trayectoria*

G.28 CU28: Temporizador Pomodoro

Propósito: Permitir a un estudiante hacer seguimiento de una tarea en tiempo real usando la técnica Pomodoro

Actor: Estudiante.

Entradas: Las de la IU22 Pantalla de Temporizador Pomodoro Corriendo

Salidas: Las de la IU22 Pantalla de Temporizador Pomodoro Corriendo

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber estimado el tiempo para completar la tarea en Pomodoros, el estudiante debe de haber presionado el botón de Pomodoro en alguna vista de tareas, el estudiante debe de haber iniciado la sesión Pomodoro

Postcondiciones: El estudiante sabe y utiliza el temporizador Pomodoro para implementar la técnica Pomodoro dentro de su sesión estudio para sus tareas.

Trayectoria principal:

1. El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo
2. El estudiante presiona el botón de sonido. [Trayectoria A]
3. El estudiante presiona el botón de Reiniciar Sesión. [Trayectoria B]
4. El estudiante presiona el botón Pausar. [Trayectoria C]
5. El estudiante presiona el botón Ayuda. [Trayectoria D]
6. El estudiante presiona el botón Terminar.
7. El sistema despliega el IU28 Modal para Terminar Sesión Pomodoro

8. El estudiante presiona el botón (Terminar Sesión). [Trayectoria E]
9. El sistema despliega el IU29 Modal para Completar Tarea Después de sesión Terminada
10. El estudiante presiona el botón (Si Completar Tarea). [Trayectoria F]
11. El sistema muestra el MSG- “Tarea Completada exitosamente”
12. El sistema despliega la vista de tareas anterior.

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante gestiona el sonido

A1: El sistema deshabilita el sonido de fondo del temporizador si este está habilitado

A2: El sistema habilita el sonido de fondo del temporizador si este está deshabilitado

---- Fin de la Trayectoria

Trayectoria alternativa B:

Condición: El estudiante presiona el botón reiniciar

B1: El sistema despliega el IU26 Modal para Reiniciar Sesión Pomodoro

B2: El estudiante presiona el botón cancelar y regresa a la IU22 Pantalla de Temporizador Pomodoro Corriendo

B3: El estudiante presiona el botón reiniciar sesión

B4: El sistema reinicia la sesión y despliega la IU21 Pantalla de Temporizador Pomodoro

---- *Fin del caso de uso*

Trayectoria alternativa C:

Condición: El estudiante presiona el botón pausar

C1: El sistema despliega el U27 Modal para Pausar la Sesión Pomodoro

C2: El estudiante presiona el botón cancelar y regresa a la IU22 Pantalla de Temporizador Pomodoro Corriendo

C3: El estudiante presiona el botón Pausar Sesión

C4: El sistema muestra el MSG- ¡Sesión Pausada exitosamente! “”

C5: El sistema reinicia la sesión y despliega la IU21 Pantalla de Temporizador Pomodoro

---- *Fin del caso de uso*

Trayectoria alternativa D:

Condición: El estudiante presiona el botón ayuda

D1: El sistema despliega el IU23 Modal de Ayuda del Temporizador Pomodoro

D2: El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda

D3: El estudiante presiona el botón de ‘X’ para salir del modal

D4: El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo

---- *Fin de la Trayectoria*

Trayectoria alternativa E:

Condición: El estudiante presiona el botón Terminar Sesión y luego el botón cancelar

E1: El sistema despliega el IU28 Modal para Terminar Sesión Pomodoro

E2: El estudiante presiona el botón (Cancelar)

E3: El sistema muestra el mensaje de advertencia MSG- “¡Acción Cancelada!”

E4: El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo

E5: El estudiante presiona el botón (Terminar Sesión)

E6: El sistema muestra el mensaje de advertencia MSG- “¡Sesión Terminada Correctamente!”

E7: El sistema despliega la IU29 Modal para Completar Tarea Despues de sesión Terminada

---- *Fin de la trayectoria*

Trayectoria alternativa F:

Condición: El estudiante presiona el botón Terminar Sesión

F1: El sistema despliega IU29 Modal para Completar Tarea Despues de sesión Terminada

F2: El estudiante presiona el botón (No Completar Tarea)

F3: El sistema muestra el mensaje de advertencia MSG- “¡Acción Cancelada!”

F4: El sistema despliega la vista de tarea anterior

F5: El estudiante presiona el botón (Si Completar Tarea)

F6: El sistema muestra el mensaje de advertencia MSG- “¡Tarea Completada Correctamente!”

F7: El sistema despliega la vista de tarea anterior

---- *Fin de la trayectoria*

G.29 CU29: Sesión Pomodoro sin Iniciar

Propósito: Permitir a un estudiante entrar a una sesión Pomodoro sin iniciar

Actor: Estudiante.

Entradas: Las de la IU21 Pantalla de Temporizador Pomodoro

Salidas: Las de la IU21 Pantalla de Temporizador Pomodoro

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber estimado el tiempo para completar la tarea en Pomodoros, el estudiante debe de haber presionado el botón de Pomodoro en alguna vista de tareas.

Postcondiciones: El estudiante sabe gestionar una sesión Pomodoro desde su inicio.

Trayectoria principal:

1. El sistema despliega la IU21 Pantalla de Temporizador Pomodoro
2. El estudiante presiona el botón (Regresar). [Trayectoria A]
3. El usuario presiona el botón Ayuda. [Trayectoria B]
4. El estudiante presiona el botón (Iniciar)
5. El sistema despliega la IU25 Modal para Empezar la Sesión Pomodoro. [Trayectoria C]
6. El estudiante presión el botón (Empezar Sesión)
7. El Sistema empieza la sesión y el temporizador empieza a correr

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante presiona el botón regresar*

A1: El sistema despliega la vista de tareas anterior.

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El estudiante presiona el botón ayuda*

D1: El sistema despliega el IU23 Modal de Ayuda del Temporizador Pomodoro

D2: El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda

D3: El estudiante presiona el botón de 'X' para salir del modal

D4: El sistema despliega la IU21 Pantalla de Temporizador Pomodoro

---- *Fin de la Trayectoria*

Trayectoria alternativa C:

Condición: El estudiante presiona el botón Cancelar

E1: El estudiante presiona el botón (Cancelar)

E2: El sistema muestra el mensaje de advertencia MSG- “¡Acción Cancelada!”

E3: El usuario continua en el paso 1 del CU 29

---- *Fin de la trayectoria*

G.30 CU30: Sesión Pomodoro Abandonada

Propósito: Permitir a un estudiante entrar a una sesión Pomodoro Abandonada

Actor: Estudiante.

Entradas: Botón para regresar, botón para reanudar sesión

Salidas: Modal de bienvenida para la sesión Abandonada

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber estimado el tiempo para completar la tarea en Pomodoros, el estudiante debe de haber presionado el botón de Pomodoro en alguna vista de tareas, la sesión debe de estar en estado abandonada

Postcondiciones: El estudiante sabe gestionar una sesión Pomodoro Abandonada

Trayectoria principal:

1. El sistema despliega una pantalla de modal dando la bienvenida a la sesión abandonada
2. El estudiante presiona el botón (Regresar). [Trayectoria A]
3. El estudiante presiona el botón (Reanudar)
4. El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo
5. El Sistema reanuda la sesión y el temporizador sigue corriendo

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante presiona el botón Regresar

A1: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.31 CU31: Sesión Pomodoro Pausada

Propósito: Permitir a un estudiante entrar a una sesión Pomodoro Pausada

Actor: Estudiante.

Entradas: Botón para regresar, botón para reanudar sesión

Salidas: Modal de bienvenida para la sesión Pausada

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber estimado el tiempo para completar la tarea en Pomodoros, el estudiante debe de haber presionado el botón de Pomodoro en alguna vista de tareas, la sesión debe de estar en estado pausada.

Postcondiciones: El estudiante sabe gestionar una sesión Pomodoro desde su inicio.

Trayectoria principal:

6. El sistema despliega una pantalla de modal dando la bienvenida a la sesión pausada
7. El estudiante presiona el botón (Regresar). [Trayectoria A]
8. El estudiante presiona el botón (Reanudar)
9. El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo
10. El Sistema reanuda la sesión y el temporizador sigue corriendo

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante presiona el botón Regresar*

A1: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.32 CU32: Sesión Pomodoro Terminada

Propósito: Permitir a un estudiante entrar a una sesión Pomodoro Terminada

Actor: Estudiante.

Entradas: Botón para regresar, botón para reanudar sesión

Salidas: Modal de bienvenida para la sesión Terminada

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber estimado el tiempo para completar la tarea en Pomodoros, el estudiante debe de haber presionado el botón de Pomodoro en alguna vista de tareas, la sesión debe de estar en estado Terminada

Postcondiciones: El estudiante sabe gestionar una sesión Pomodoro Terminada

Trayectoria principal:

1. El sistema despliega una pantalla de modal dando la bienvenida a la sesión Terminada
2. El estudiante presiona el botón (Regresar). [Trayectoria A]

3. El estudiante presiona el botón (Reanudar)
4. El sistema despliega la IU22 Pantalla de Temporizador Pomodoro Corriendo
5. El Sistema reanuda la sesión y el temporizador sigue corriendo

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: El estudiante presiona el botón Regresar

A1: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.33 CU33: Historial de la Sesión Pomodoro

Propósito: Permitir a un estudiante visualizar el historial de una tarea específica

Actor: Estudiante.

Entradas: Las de la IU19 Pantalla de Historial de Tarea

Salidas: Las de la IU19 Pantalla de Historial de Tarea

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe de haber seleccionado una tarea previamente, el estudiante debe de haber presionado el botón de Historial

Postcondiciones: El estudiante conoce el historial de la tarea seleccionada

Trayectoria principal:

1. El sistema despliega una la IU19 Pantalla de Historial de Tarea
2. El sistema recupera los eventos de la tarea y los lista en la parte de eventos de tarea
3. El sistema recupera los eventos de la sesión Pomodoro y los lista en la parte de sesiones Pomodoro
4. El estudiante presiona el botón (Regresar).
5. El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

G.34 CU34: Métricas Sesiones Pomodoro

Propósito: Permitir a un estudiante visualizar las métricas de las sesiones Pomodoro

Actor: Estudiante.

Entradas: Las de la IU51 Pantalla de Métricas de la Cuenta

Salidas: Las de la IU51 Pantalla de Métricas de la Cuenta

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe haber presionado el botón de Métricas de la barra de menú de opciones del sistema.

Postcondiciones: El estudiante conoce las métricas relacionadas con las sesiones Pomodoro de su cuenta.

Trayectoria principal:

1. El sistema despliega la IU51 Pantalla de Métricas de la Cuenta
2. El sistema calcula las métricas para Sesiones Pomodoro
3. El sistema actualiza el diagrama de pastel sobre Sesiones Pomodoro
4. El estudiante lee las métricas para Sesiones Pomodoro
5. El estudiante presiona el botón (Regresar). [Trayectoria A]
6. El estudiante presiona el bote (Ayuda). [Trayectoria B]

---- *Fin del caso de uso*

Trayectoria alternativa A:

Condición: *El estudiante ya no requiere ver las métricas de la cuenta*

A1: El sistema muestra el MSG- “¡Salió De Métricas del Sistema!”

A2: El sistema despliega la vista de tareas anterior

---- *Fin del caso de uso*

Trayectoria alternativa B:

Condición: *El estudiante requiere obtener ayuda sobre la pantalla de métricas*

B1: El sistema despliega el IU52 Modal de Ayuda sobre las Métricas

B2: El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda

B3: El estudiante presiona el botón de ‘X’ para salir del modal

B4: La estudiante continua en el paso 1 del CU 26

---- *Fin del caso de uso*

G.35 CU35: Ayuda Pomodoro

Propósito: Permitir a un estudiante obtener ayuda sobre la vista del temporizador Pomodoro.

Actor: Estudiante.

Entradas: Las del IU23 Modal de Ayuda del Temporizador Pomodoro

Salidas: Las del IU23 Modal de Ayuda del Temporizador Pomodoro

Precondiciones: El estudiante debe de haber iniciado sesión en el sistema previamente, el estudiante debe estar en el temporizador Pomodoro dentro de la sesión de una tarea previamente seleccionada.

Postcondiciones: El estudiante sabe que se hace en la pantalla del temporizador Pomodoro.

Trayectoria principal:

1. El sistema despliega IU23 Modal de Ayuda del Temporizador Pomodoro
2. El estudiante mueve la imagen actual del carrusel ya sea hacia la derecha o a la izquierda
3. El estudiante presiona el botón de ‘X’ para salir del modal
4. El sistema despliega IU21 Pantalla de Temporizador Pomodoro

---- *Fin del caso de uso*

H. Interfaces Reales del Sistema

Tabla 33: Interfaces Reales del Sistema

H.1 IU1 Pantalla De Bienvenida al Sistema	
Objetivo	Diseño
Dar la bienvenida al Usuario, proporcionando una introducción al sistema y sus características principales	<p>La pantalla de bienvenida es la primera pantalla que ve el usuario al acceder al sistema web. Desde esta pantalla se puede ir al Registro de Usuario y al Inicio de Sesión.</p> <p>Imagen IU1 Pantalla de Bienvenida al Sistema</p>
Entradas	
Botones:	
-Inicio: Lleva al usuario a la IU1 Pantalla de Bienvenida al sistema.	
-Registrar: Lleva al usuario a la IU2 Pantalla de Registro de Usuario	
-Ingresar: Lleva al usuario a la IU3 Pantalla de Inicio de Sesión	
-Comienza Ahora: Lleva al usuario a la IU2 Pantalla de Registro de Usuario	
Salidas	
El sistema muestra al usuario en forma de texto e imágenes, las características principales del sistema, las cuales son Gestionar Notas y Gestionar Tareas. El sistema también muestra la fecha y hora actual en la parte superior.	
H.2 IU2 Registro de Usuario	
Objetivo	Diseño
Permitir al Usuario crear una nueva cuenta Registrándose en el sistema.	Pantalla de registro de usuario que requiere nombre de usuario, correo y contraseña para poder registrar un nuevo usuario.
Entradas	
Campos de texto:	
Nombre de Usuario, Correo Electrónico, Contraseña, Confirmar Contraseña	
Botones:	
-Regístrate: Manda los datos al servidor para verificar si el Correo es único y de esta forma poder registrar al usuario.	
Salidas	
-Campos de textos requeridos	
-Mensaje de Error en cada campo si es que está vacío o no cumple con su especificación.	
-Mensaje de Error si las contraseñas no son iguales.	
-Mensaje de Advertencia si el Correo Electrónico ya existe.	
-Mensaje de Éxito si el Registro de Usuario fue exitoso.	
Imagen IU2 Pantalla de Registro de Usuario	
Nota: Elaboración Propia	

H.3 IU3 Inicio de Sesión

Objetivo	Diseño
Permitir al usuario Iniciar Sesión en su cuenta y acceder al sistema.	Pantalla de Inicio de Sesión que requiere Correo y Contraseña previamente registrados para poder acceder al Sistema.
Entradas	Imagen IU3 Pantalla de Inicio de Sesión
Campos de texto: Correo Electrónico, Contraseña Botones:	
-Acceder: Manda los datos al servidor para verificar si el correo existe, y si la contraseña es correcta para este correo lleva al usuario a la IU4 Vista de Tareas Eisenhower	
Salidas	
-Campos de textos requeridos -Mensaje de Error en cada campo si es que está vacío o no cumple con su especificación. -Mensaje de Error si el usuario no existe -Mensaje de Error si la contraseña es incorrecta	<hr/> <p>Nota: Elaboración Propia</p>

H.4 IU4 Tareas Eisenhower

Objetivo	Diseño
Permitir al Usuario ver todas las tareas sin completar en una vista de Matriz Eisenhower	En esta vista se pueden ver las tareas divididas dependiendo su urgencia e importancia, botones para interactuar con las tareas y botón para cambiar de vista.
Entradas	
Checkbox en cada tarea, para poder completarla.	
Cada tarea es un objeto que se puede arrastrar entre los distintos cuadrantes	
Botones:	
-Calendario: Cambia de Vista a la IU8 Vista de Tareas Calendario	
-Ayuda: Lleva a la IU6 Modal de Ayuda Vista Eisenhower	
-Tareas Completadas: Lleva a la IU13 Pantalla de tareas completadas	
-Agregar: Lleva a la IU14 Pantalla para Agregar Tarea	
Salidas	
-Listas de tareas, las cuales contienen checkbox, título de la tarea, cantidad de Pomodoros estimados y fecha y hora límites.	
-Mensaje de Información de cada elemento usando ToolTips.	

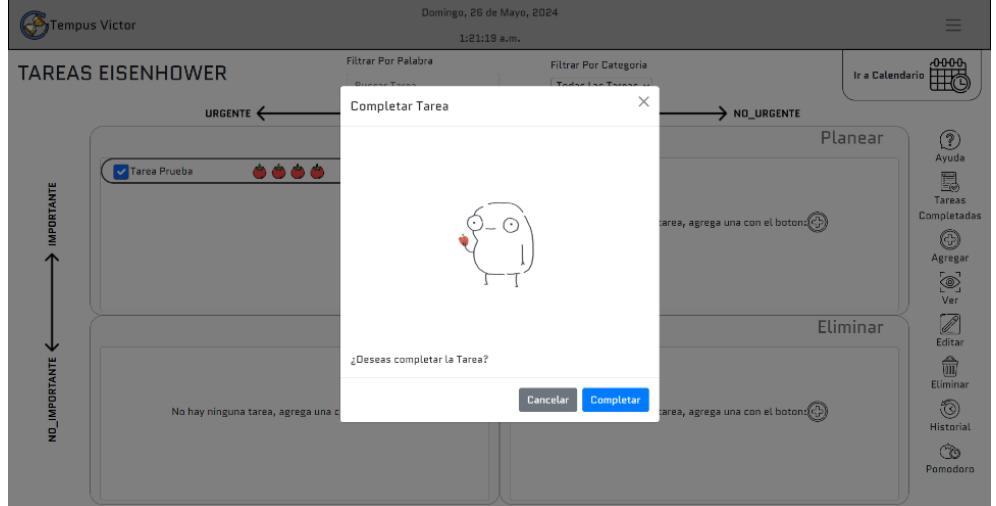
H.5 IU5 Eisenhower Seleccionar Tarea

Objetivo	Diseño
<p>Permitir al usuario interactuar con una tarea seleccionada en la vista Eisenhower</p> <p>Entradas</p> <p>Checkbox de la tarea, para poder completarla. Puedes mover la tarea seleccionada entre los distintos cuadrantes.</p> <p>Botones:</p> <ul style="list-style-type: none"> -Calendario: Cambia de Vista a la IU8 Vista de Tareas Calendario -Ayuda: Lleva a la IU6 Modal de Ayuda Vista Eisenhower -Tareas Completadas: Lleva a la IU13 Pantalla de tareas completadas -Agregar: Lleva a la IU14 Pantalla para Agregar Tarea -Ver: Lleva a la IU16 Pantalla para Ver Tarea -Editar: Lleva a la IU17 Pantalla para Editar Tarea -Eliminar: Lleva a la IU18 Pantalla para Eliminar Tarea -Historial: Lleva a la IU19 Pantalla de historial de tarea -Pomodoro: Lleva a la IU20 Pantalla de Estimación de Sesión Pomodoro <p>Salidas</p> <ul style="list-style-type: none"> -Resalta la Tarea Seleccionada la cual contiene: checkbox, título de la tarea, cantidad de Pomodoros estimados, fecha y hora límites. -Mensaje de Información de cada elemento usando ToolTips. 	<p>Pantalla que muestra lo mismo que la vista de Eisenhower más los botones habilitados para la tarea seleccionada</p> <p>Imagen IU5 Tarea seleccionada Vista Eisenhower</p>
	<p>Nota: Elaboración Propria</p>

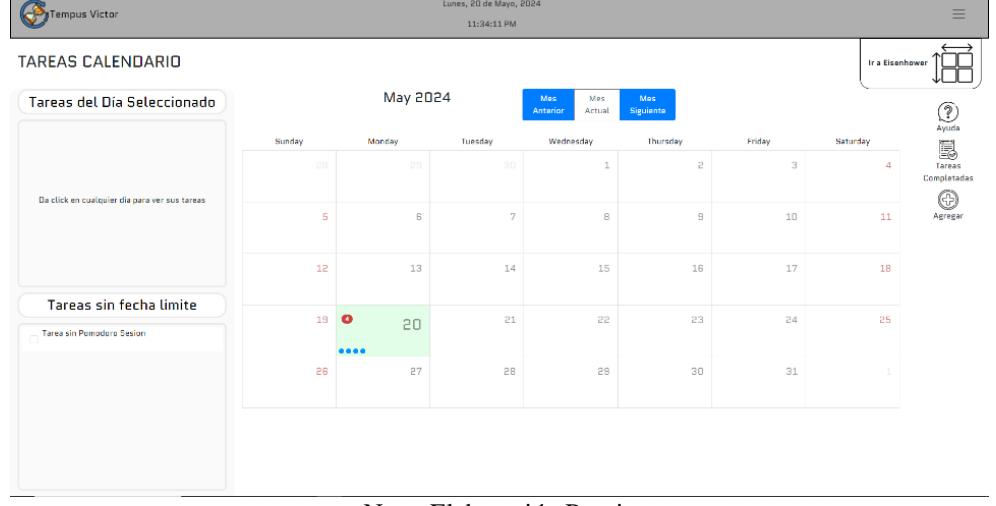
H.6 IU6 Ayuda Vista Eisenhower

Objetivo	Diseño
<p>Permitir al usuario obtener ayuda de la vista de Eisenhower para tareas.</p> <p>Entradas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Imagen anterior: Cambia la imagen actual a la anterior. -Imagen siguiente: Cambia la imagen actual a la siguiente. -Salir del carrusel: Sale del carrusel y lleva a la IU4 Vista de Tareas Eisenhower <p>Salidas</p> <p>Mensaje informativo cuando el usuario sale del carrusel.</p>	<p>Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en la vista de Eisenhower.</p> <p>Imagen IU6 Modal de Ayuda Vista Eisenhower</p>
	<p>Nota: Elaboración Propria</p>

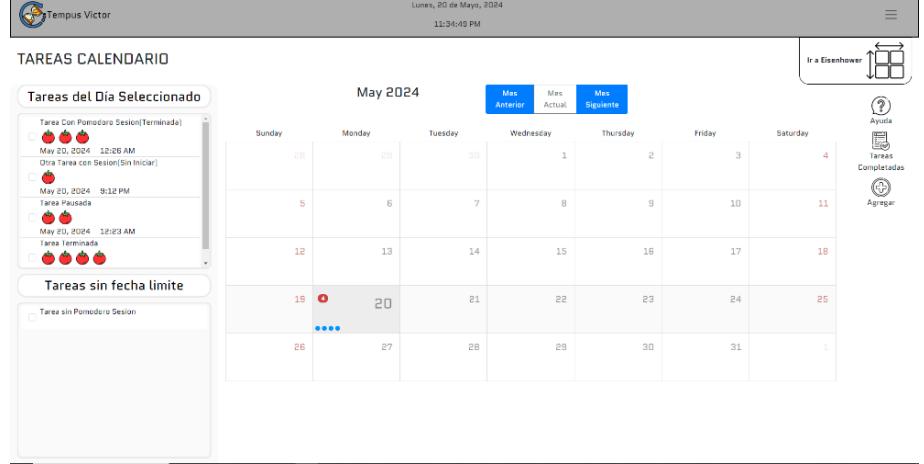
H.7 IU7 Modal Completar Tarea Vista Eisenhower

Objetivo	Diseño
Permitir al usuario completar la tarea seleccionada.	Modal que pregunta si se desea completar la tarea dando la opción de completarla o cancelar la acción.
Entradas	Imagen IU7 Modal para Completar Tarea en Vista Eisenhower
Botones: -Cancelar: Lleva a la IU4 Vista de Tareas Eisenhower y manda un mensaje de acción cancelada -Completar: Hace la petición al servidor de actualizar el estado de la tarea seleccionada a estado completada. Cambiando la tarea a la lista de tareas completadas y actualizando la IU4 Vista de Tareas Eisenhower	
Salidas	Nota: Elaboración Propia
-Título del Modal -Imagen gif de un personaje sosteniendo un Pomodoro -Pregunta ¿Deseas completar la Tarea?	

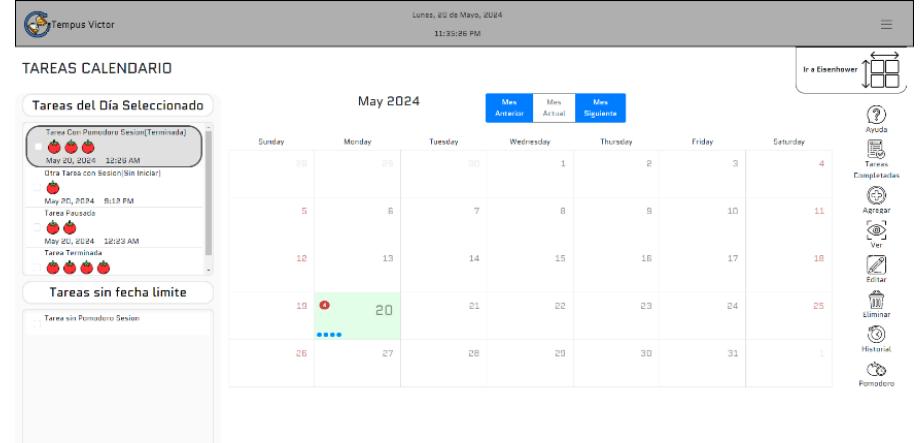
H.8 IU8 Tareas Calendario

Objetivo	Diseño
Permitir al Usuario ver todas las tareas sin completar en una vista de Calendario	En esta vista se pueden ver las tareas acomodadas en el día de la fecha límite si es que tiene, si no tiene fecha límite aparece en una lista de tareas sin fecha límite.
Entradas	
Elementos con los que se puede interactuar: Cada día del calendario se puede seleccionar. Cada circulo representa una tarea, la cual se puede mover entre días, actualizando su fecha límite. Botones: -Eisenhower: Cambia de Vista a la IU4 Vista de Tareas Eisenhower -Ayuda: Lleva a la IU11 Modal de Ayuda Vista Calendario -Tareas Completadas: Lleva a la pantalla IU13 Pantalla de Tareas Completadas -Agregar: Lleva a la pantalla IU14 Pantalla para Agregar Tarea	Imagen IU8 Vista de Tareas Calendario
Salidas	
-Calendario con vista de meses colocado en el mes actual, con el día actual resaltado. -Mensaje de Información de cada elemento usando ToolTips.	Nota: Elaboración Propia

H.9 IU9 Seleccionar Dia Calendario

Objetivo	Diseño
Permitir al Usuario seleccionar un día del calendario para ver las tareas pendientes que son para ese día seleccionado.	Si es que el día seleccionado tiene tareas pendientes, estas aparecerán en una lista de tareas del día seleccionado, en donde se podrá interactuar con ellas con los botones de la vista.
Entradas Elementos con los que se puede interactuar: Cada día del calendario se puede seleccionar, Cada circulo representa una tarea, la cual se puede mover entre días, actualizando su fecha límite, Checkbox de la tarea, para poder completarla. Botones: -Calendario: Cambia de Vista a la IU4 Vista de Tareas Eisenhower -Ayuda: Lleva a la IU11 Modal de Ayuda Vista Calendario -Tareas Completadas: Lleva a la IU13 Pantalla de tareas completadas -Agregar: Lleva a la IU14 Pantalla para Agregar Tarea	 <p>Imagen IU9 Seleccionar Dia vista Calendario</p> <p>Este es un screenshot de la IU9 Seleccionar Dia vista Calendario. La pantalla muestra un calendario para el mes de Mayo de 2024. El día 20 de Mayo es el día actual, resaltado con un efecto especial. En la parte superior izquierda, hay un logo de "Tempus Victor". A la derecha del calendario, hay un ícono para "Ir a Eisenhower". En la parte inferior derecha, hay un menú con iconos para Ayuda, Tareas Completadas, Agregar, Ver, Editar, Eliminar y Historial. Una lista lateral titulada "TAREAS CALENDARIO" muestra tareas pendientes para el día 20 de Mayo, incluyendo "Tarea Con Pomodoro Sesión[Terminada]" y "Tarea Pausada".</p>
Salidas -Calendario con vista de meses colocado en el mes actual, con el día actual resaltado. -El día seleccionado se resalta de manera más fuerte que el día actual. -Mensaje de Información de cada elemento usando ToolTips.	<p>Nota: Elaboración Propia</p>

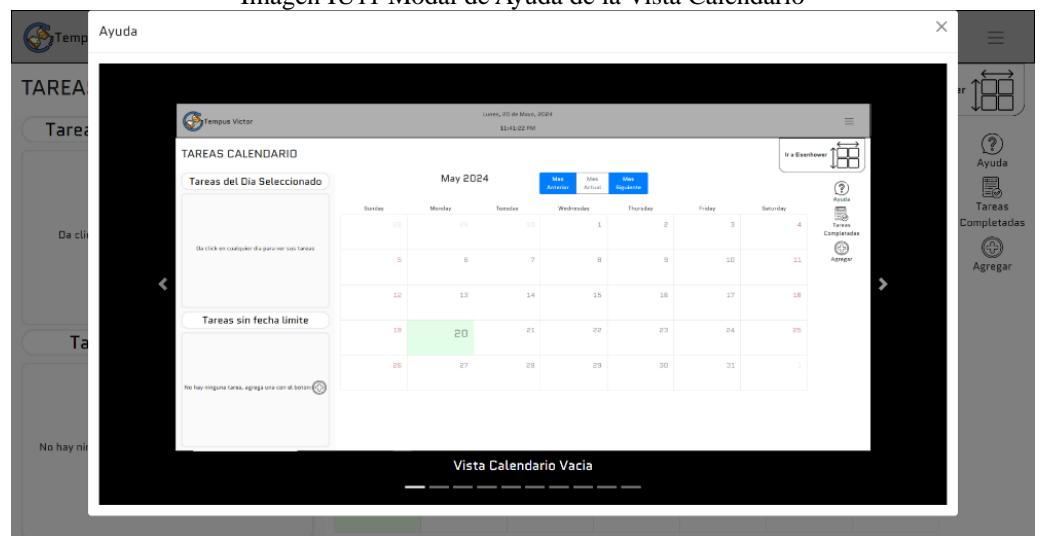
H.10 IU10 Seleccionar Tarea Calendario

Objetivo	Diseño
Permitir al usuario interactuar con una tarea previamente seleccionada en la vista de Calendario	Pantalla que muestra lo mismo que la vista de Calendario más los botones habilitados para la tarea seleccionada
Entradas Las mismas que la pantalla IU9 Seleccionar Dia Calendario más lo siguiente: Botones: -Ver: Lleva a la IU16 Pantalla para Ver Tarea -Editar: Lleva a la IU17 Pantalla para Editar Tarea -Eliminar: Lleva a la IU18 Pantalla para Eliminar Tarea -Historial: Lleva a la IU19 Pantalla de historial de tarea -Pomodoro: Lleva a la IU20 Pantalla de Estimación de Sesión Pomodoro Salidas -Resalta la Tarea Seleccionada la cual contiene: checkbox, título de la tarea, cantidad de Pomodoros estimados, fecha y hora límites. -Mensaje de Información de cada elemento usando ToolTips	 <p>Imagen IU10 Seleccionar Tarea Vista Calendario</p> <p>Este es un screenshot de la IU10 Seleccionar Tarea Vista Calendario. Es similar a la IU9, pero el día 20 de Mayo no solo está resaltado, sino que también tiene un efecto de resalte especial. La lista lateral "TAREAS CALENDARIO" ahora muestra solo la tarea "Tarea Con Pomodoro Sesión[Terminada]" para el día 20 de Mayo, resaltada con un efecto de resalte.</p>

Nota: Elaboración Propia

H.11 IU11 Ayuda Vista Calendario

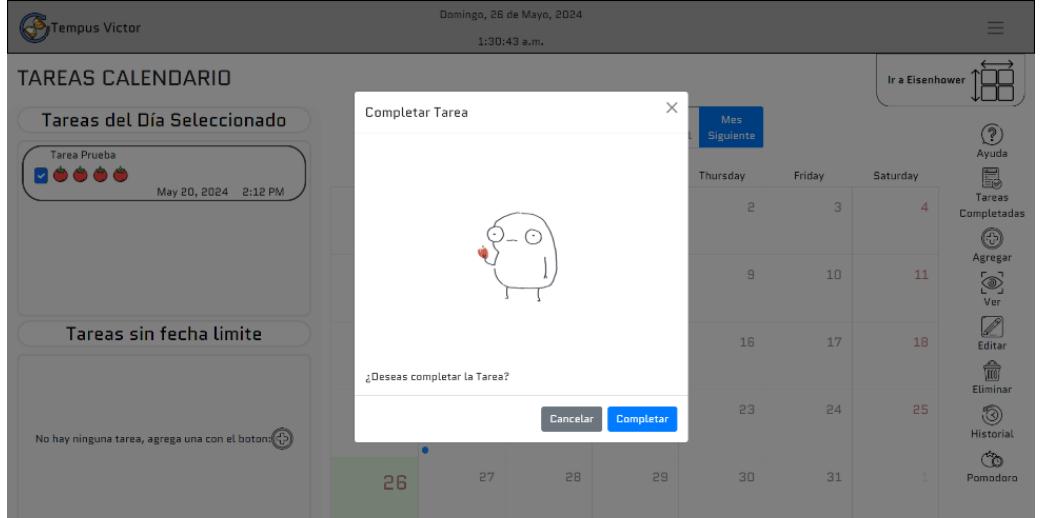
Objetivo	Diseño
Permitir al usuario obtener ayuda de la vista de Calendario para tareas.	Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en la vista de Calendario.
Entradas	Imagen IU11 Modal de Ayuda de la Vista Calendario
Botones:	
-Imagen anterior: Cambia la imagen actual a la anterior.	
-Imagen siguiente: Cambia la imagen actual a la siguiente.	
-Salir del carrusel: Sale del carrusel y lleva a la IU8 Vista de Tareas Calendario	
Salidas	
Mensaje informativo cuando el usuario sale del carrusel.	



Nota: Elaboración Propia

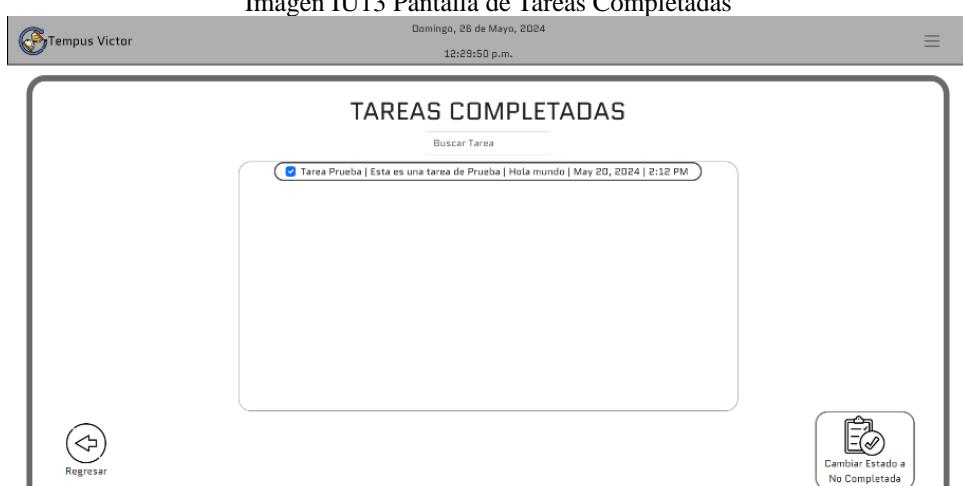
H.12 IU12 Modal Completar Tarea Vista Calendario

Objetivo	Diseño
Permitir al usuario completar la tarea seleccionada.	Modal que pregunta si se desea completar la tarea dando la opción de completarla o cancelar la acción.
Entradas	Imagen IU12 Modal para Completar Tarea en Vista Calendario
Botones:	
-Cancelar: Lleva a la IU8 Vista de Tareas Calendario y manda un mensaje de acción cancelada	
-Completar: Hace la petición al servidor de actualizar el estado de la tarea seleccionada a estado completada. Cambiando la tarea a la lista de tareas completadas y actualizando la IU8 Vista de Tareas Calendario	
Salidas	
-Título del Modal	
-Imagen gif de un personaje sosteniendo un Pomodoro	
-Pregunta ¿Deseas completar la Tarea?	



Nota: Elaboración Propia

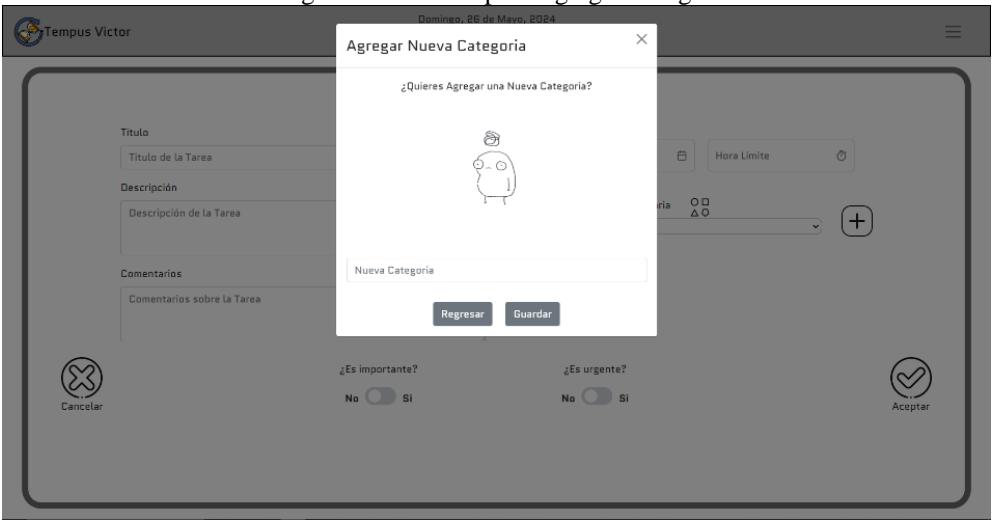
H.13 IU13 Tareas Completadas

Objetivo	Diseño
Permitir al Usuario ver una lista de tareas que ya completo en algún momento. Dando la posibilidad de cambiar el estado a no completadas, de alguna tarea que seleccione.	Pantalla que contiene una lista de tareas, un filtro por palabras, un botón para regresar a la pantalla anterior y el botón que posibilita cambiar el estado de la tarea seleccionada.
Entradas	
Campos de texto: Buscar tare: Aplica un filtro a la lista de tareas intentando buscar la tarea que tenga las letras que se ingresen en el campo de texto.	
Botones: -Regresar: Lleva a la vista de Tareas anterior -Cambiar Estado Tarea: Manda la solicitud al servidor de actualizar el estado de la tarea a no completada.	
Salidas	
-Mensaje de Éxito si se actualizo el estado de una tarea correctamente. -Modal para preguntar si se cambia el estado o no. -Mensaje de Información de cada elemento usando ToolTips.	

H.14 IU14 Agregar Tarea

Objetivo	Diseño
Permitir al Usuario agregar una nueva tarea por completar.	Pantalla de para agregar tarea, en la cual puedes agregarle título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia.
Entradas	
Campos de texto: Título, Descripción, Comentarios Selección de opción: Fecha límite, hora límite, categoría Radio Button: Importancia y urgencia Botones: -Cancelar: Lleva a la vista de Tareas anterior -Aceptar: Manda la información al servidor creando una nueva tarea. -Agregar Categoría: Lleva al IU15 Modal para agregar una nueva categoría personalizada a las que ya existen.	
Salidas	
-Mensaje de campos de textos requeridos -Mensaje de Error en cada campo si es que está vacío o no cumple con su especificación. -Mensaje de Éxito si se creó la Tarea correctamente -Mensaje de Información de cada elemento usando ToolTips.	

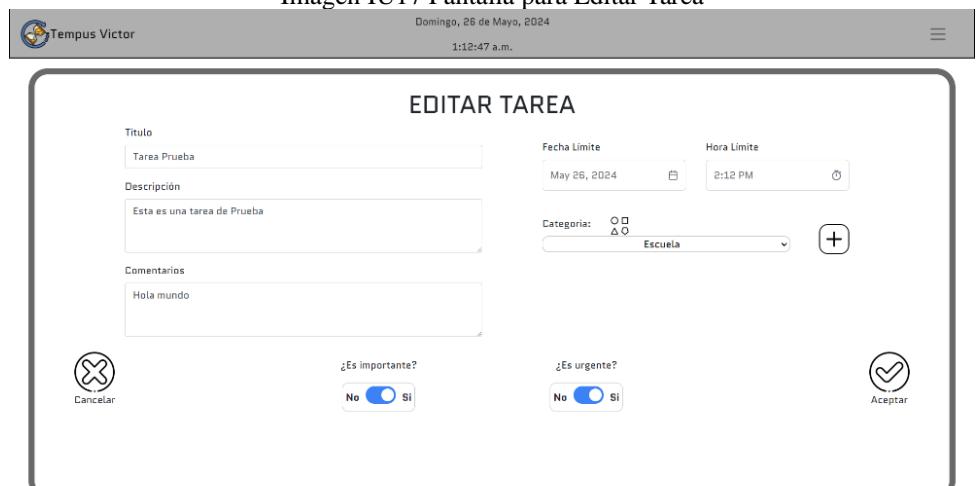
H.15 IU15 Modal Agregar Categoría

Objetivo	Diseño
Permitir al Usuario agregar una nueva categoría personalizada a las ya existentes	Modal que pregunta si se desea agregar una nueva categoría y usa un campo de texto para dar la opción de crear la nueva categoría.
Entradas	
Campos de texto: Nueva categoría: Para crear la categoría que se escriba dentro de este campo. Botones: -Regresar: Lleva a la IU14 Pantalla para agregar Tarea y manda un mensaje de acción cancelada -Guardar: Realiza la petición al servidor de crear una nueva categoría, luego actualiza el selector de categoría de la IU14 Pantalla para agregar Tarea	 Imagen IU15 Modal para Agregar Categoría Este modal es parte de la IU15. Tiene un encabezado "Tempus Victor" y la fecha "Domingo, 26 de Mayo, 2024". El contenido pregunta "¿Quieres Agregar una Nueva Categoría?" y muestra un icono de un personaje pensando. Hay campos para "Título" (Titular de la Tarea) y "Descripción" (Descripción de la Tarea), así como un campo para "Comentarios" (Comentarios sobre la Tarea). Un cuadro de texto para "Nueva Categoría" contiene el texto "Categoría". Botones "Regresar" y "Guardar" están en la parte inferior. Se incluyen secciones para "Es importante?" y "Es urgente?", cada una con un switch (botón on/off) y la opción "Si". En la parte derecha, hay un selector de categoría con íconos de categorías y un botón "+". A la derecha del modal, hay un botón "Aceptar" con un checkmark.
Salidas	
-Título del Modal -Imagen gif de un personaje pensando muy fuerte -Pregunta ¿Deseas agregar una Nueva categoría? -Mensaje de éxito si se creó la nueva categoría correctamente.	Nota: Elaboración Propia

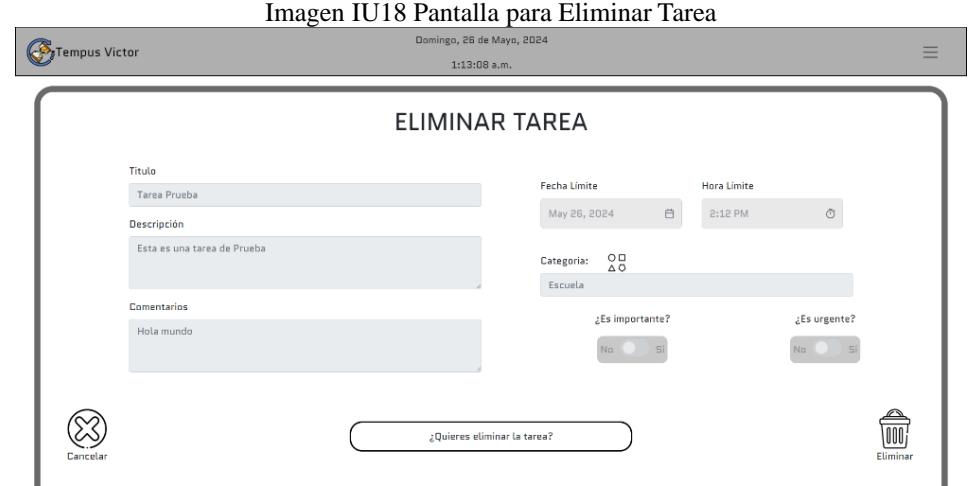
H.16 IU16 Ver Tarea

Objetivo	Diseño
Permitir al usuario Ver por completo a la Tarea previamente seleccionada	Pantalla informativa en la que se ven las características de una tarea existente previamente seleccionada.
Entradas	
Botones: -Regresar: Lleva a la vista de Tareas anterior	 Imagen IU16 Pantalla para Ver Tarea Este panel es parte de la IU16. Tiene un encabezado "Tempus Victor" con la fecha "Domingo, 26 de Mayo, 2024" y la hora "1:12:29 a.m.". El título es "VER TAREA". Muestra una tarea llamada "Tarea Prueba" con la descripción "Esta es una tarea de Prueba". Los comentarios indican "Hola mundo". Los datos de fecha y hora límite son "May 26, 2024" y "2:12 PM". La categoría es "Escuela". Se incluyen secciones para "Es importante?" y "Es urgente?", cada una con un switch y la opción "Si".
Salidas	
-Campos de textos de la tarea -Información sobre la tarea, título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia. -Mensaje de Información de cada elemento usando ToolTips.	Nota: Elaboración Propia

H.17 IU17 Editar Tarea

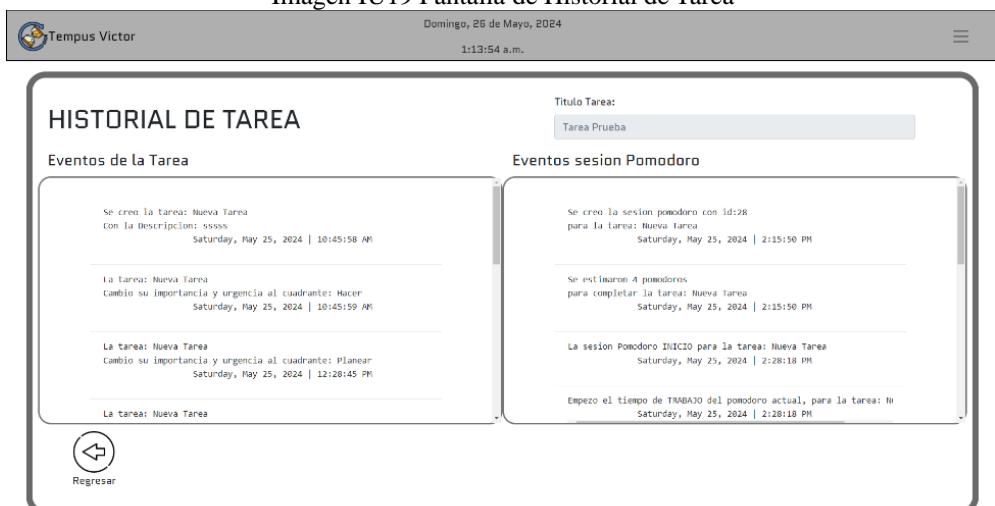
Objetivo	Diseño
Permitir al Usuario editar a la Tarea previamente seleccionada	Pantalla de para editar Tarea, en la cual puedes editar el título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia.
Entradas	
Campos de texto: Título, Descripción, Comentarios Selección de opción: Fecha límite, hora límite, categoría Radio Button: Importancia y urgencia Botones: -Cancelar: Lleva a la vista de Tareas anterior -Aceptar: Manda la información al servidor creando una nueva tarea. -Agregar Categoría: Lleva al IU15 Modal para agregar una nueva categoría personalizada a las que ya existen.	 <p>Imagen IU17 Pantalla para Editar Tarea</p> <p>Domingo, 26 de Mayo, 2024 1:12:47 a.m.</p> <p>EDITAR TAREA</p> <p>Título: Tarea Prueba Descripción: Esta es una tarea de Prueba Comentarios: Hola mundo</p> <p>Fecha Límite: May 26, 2024 Hora Límite: 2:12 PM</p> <p>Categoría: Escuela</p> <p>¿Es importante? (Si) ¿Es urgente? (Si)</p> <p>Cancelar Aceptar</p>
Salidas	<p>-Mensaje de campos de textos requeridos -Mensaje de Error en cada campo si es que está vacío o no cumple con su especificación. -Mensaje de Éxito si se creó la Tarea correctamente -Mensaje de Información de cada elemento usando ToolTips.</p>

H.18 IU18 Eliminar Tarea

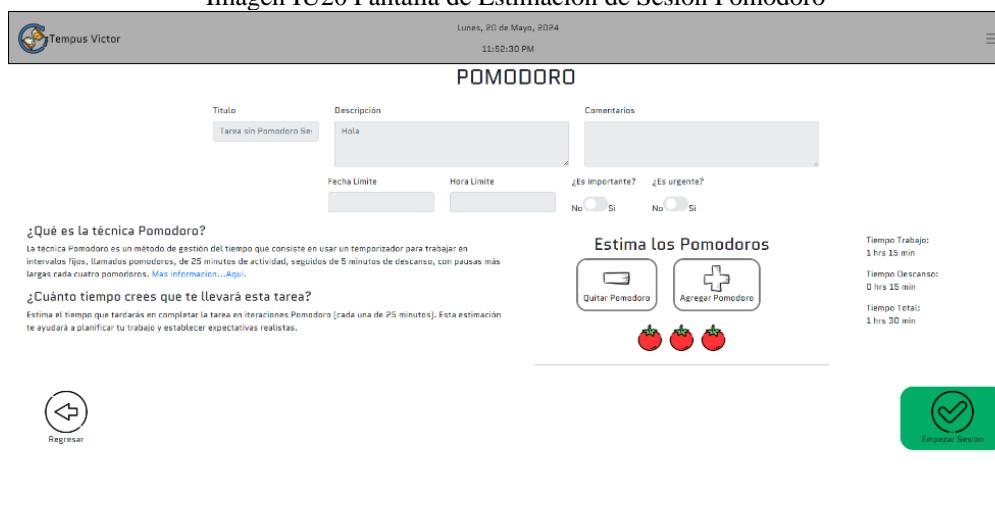
Objetivo	Diseño
Permitir al usuario Eliminar a la Tarea previamente seleccionada	Pantalla informativa en la que se ven las características de una tarea existente previamente seleccionada, y se da la posibilidad de eliminar a la tarea seleccionada.
Entradas	
Botones: -Cancelar: Lleva a la vista de Tareas anterior -Eliminar: Elimina la tarea seleccionada de la base de datos del sistema y lleva de regreso a la vista de tareas anterior.	 <p>Imagen IU18 Pantalla para Eliminar Tarea</p> <p>Domingo, 26 de Mayo, 2024 1:13:08 a.m.</p> <p>ELIMINAR TAREA</p> <p>Título: Tarea Prueba Descripción: Esta es una tarea de Prueba Comentarios: Hola mundo</p> <p>Fecha Límite: May 26, 2024 Hora Límite: 2:12 PM</p> <p>Categoría: Escuela</p> <p>¿Es importante? (No) ¿Es urgente? (No)</p> <p>Cancelar ¿Quieres eliminar la tarea? Eliminar</p>
Salidas	<p>-Campos de textos de la tarea -Información sobre la tarea, título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia. -Pregunta ¿Quieres eliminar la tarea? -Mensaje de Éxito si se eliminó la Tarea correctamente -Mensaje de Información de cada elemento usando ToolTips.</p>

Nota: Elaboración Propia

H.19 IU19 Historial Tarea

Objetivo	Diseño
Permitir al Usuario visualizar el Historial de eventos de la Tarea seleccionada, tanto eventos de la tarea como de la sesión Pomodoro si es que tiene.	Pantalla de para ver historial de todos los eventos correspondientes a la tarea seleccionada, cuando se creó, si se modificó o si tuvo un cambio a lo largo del tiempo.
Entradas	
Botones:	
-Regresar: Lleva a la vista de Tareas anterior -Scroll de lista de eventos de tarea -Scroll de lista de eventos de sesión Pomodoro	
Salidas	
-Título de la pantalla -Título de la Tarea -Lista de eventos de la tarea -Lista de eventos de la Sesión Pomodoro -Mensaje de Información de cada elemento usando ToolTips.	

H.20 IU20 Estimar Sesión Pomodoro

Objetivo	Diseño
Permitir al usuario Estimar en Pomodoros el tiempo que cree que le tomara realizar la tarea seleccionada.	Pantalla que recuerda al usuario que es la técnica Pomodoro dando una breve explicación sobre esta, proporcionando un link a un sitio web con la información más completa, y un apartado para la estimación de la sesión Pomodoro.
Entradas	
Botones:	
-Regresar: Lleva a la vista de Tareas anterior -Empezar Sesión: Lleva a la IU21 Pantalla de Temporizador Pomodoro -Quitar Pomodoro: Quita un Pomodoro de la estimación actual -Agregar Pomodoro: Agrega un Pomodoro a la estimación actual	
Salidas	
-Información sobre la tarea, título, descripción, comentarios, fecha y hora límite, categoría, importancia y urgencia. -Mensaje de Advertencia si se quiere empezar la sesión sin estimar al menos 1 Pomodoro. -Mensaje de Error si se quiere estimar con más de 4 Pomodoros -Mensaje de Información de cada elemento usando ToolTips.	

H.21 IU21 Temporizador Pomodoro

Objetivo	Diseño
<p>Permitir al Usuario trabajar en la tarea seleccionada haciendo uso del temporizador Pomodoro el cual implementa la Técnica Pomodoro</p>	<p>Pantalla que implementa la técnica Pomodoro que es un método de gestión del tiempo que consiste en usar un temporizador para trabajar en intervalos fijos, llamados Pomodoros, de 25 minutos de actividad, seguidos de 5 minutos de descanso, con pausas más largas cada cuatro Pomodoros.</p>
<p>Entradas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Regresar: Lleva a la vista de Tareas anterior -Iniciar: Inicia la sesión Pomodoro, y empieza a correr el tiempo del temporizador -Pausar: Pausa la sesión Pomodoro Actual -Terminar: Termina la sesión Pomodoro Actual <p>Salidas</p> <ul style="list-style-type: none"> -Mensaje de Sesión Pomodoro Creada -Mensaje de Bienvenida al Temporizador Pomodoro -Información sobre la tarea, título, descripción, comentarios, cantidad de Pomodoros estimados para la sesión. -Lista de tiempos de la sesión Pomodoro actual -Modal para Iniciar sesión Pomodoro -Mensaje de Información de cada elemento usando ToolTips. 	<p>Imagen IU21 Pantalla de Temporizador Pomodoro</p> <p>Notas: Elaboración Propia</p>

H.22 IU22 Temporizador Pomodoro Corriendo

Objetivo	Diseño
<p>Permitir al ver lo mismo que la IU21 Pantalla de Temporizador Pomodoro, pero cuando el tiempo del temporizador está corriendo.</p> <p>Entradas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Resetear: Resetea la sesión Pomodoro Actual -Reanudar: Reanuda la sesión Pomodoro, y continúa corriendo el tiempo del temporizador -Pausar: Pausa la sesión Pomodoro Actual -Terminar: Termina la sesión Pomodoro Actual -Habilitar/Deshabilitar Sonido: Gestiona el sonido de fondo del temporizador <p>Salidas</p> <ul style="list-style-type: none"> -Mismas salidas que la IU21 Pantalla de Temporizador Pomodoro -Modal para Resetear la sesión -Modal para Pausar la sesión -Modal para Terminar la sesión -Modal de Alarma 	<p>Pantalla que implementa la técnica Pomodoro que es un método de gestión del tiempo que consiste en usar un temporizador para trabajar en intervalos fijos, llamados Pomodoros, de 25 minutos de actividad, seguidos de 5 minutos de descanso, con pausas más largas cada cuatro Pomodoros.</p> <p>Imagen IU22 Pantalla de Temporizador Pomodoro Corriendo</p> <p>Notas: Elaboración Propia</p>

H.23 IU23 Ayuda Pomodoro

Objetivo	Diseño
Permitir al usuario obtener ayuda del temporizador Pomodoro	Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en el temporizador Pomodoro
Entradas	
Botones:	
-Imagen anterior: Cambia la imagen actual a la anterior.	
-Imagen siguiente: Cambia la imagen actual a la siguiente.	
-Salir del carrusel: Sale del carrusel y lleva a la IU21 Pantalla de Temporizador Pomodoro	
Salidas	
Mensaje informativo cuando el usuario sale del carrusel.	

Imagen IU23 Modal de Ayuda del Temporizador Pomodoro

Nota: Elaboración Propia

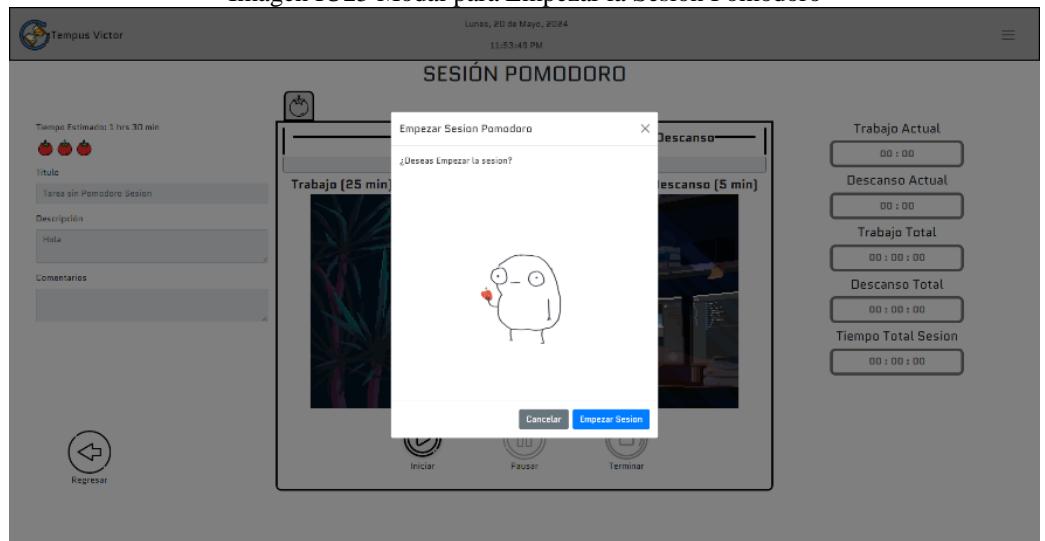
H.24 IU24 Nuevo Pomodoro Temporizador

Objetivo	Diseño
Permite mostrarle al usuario de manera grafica cuando ha completado un Pomodoro, esto es el tiempo de trabajo y el de descanso completados.	Pantalla que muestra al usuario la actualización en pantalla de un nuevo Pomodoro y de la lista de Pomodoros completados, en ella se verán la cantidad de Pomodoros llenos de color rojo dependiendo cuantos Pomodoros se hayan completado.
Entradas	
Las mismas que la IU22 Pantalla de Temporizador Pomodoro Corriendo	
Salidas	
Las mismas que la IU22 Pantalla de Temporizador Pomodoro Corriendo	

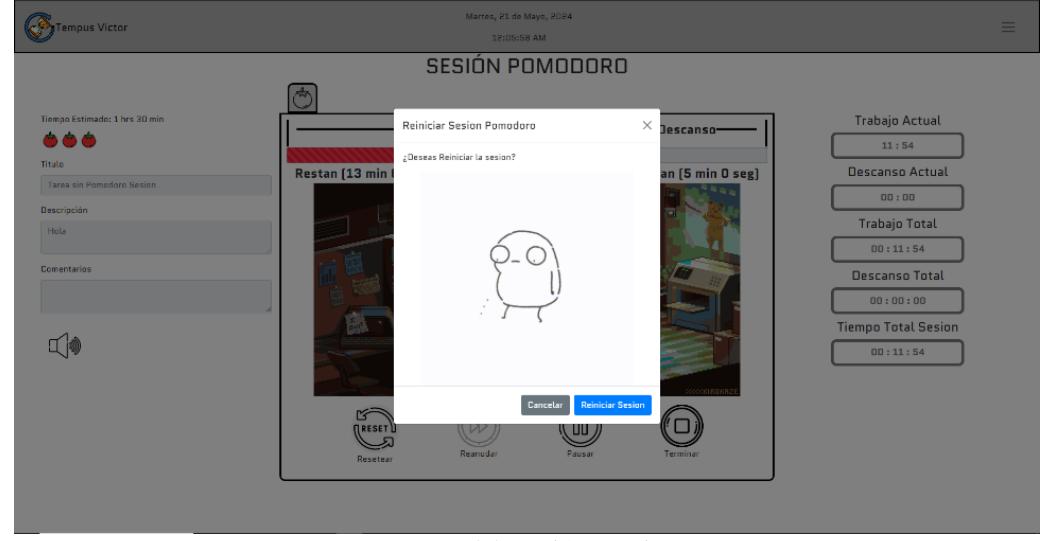
Imagen IU24 Pantalla de Nuevo Pomodoro en Temporizador

Nota: Elaboración Propia

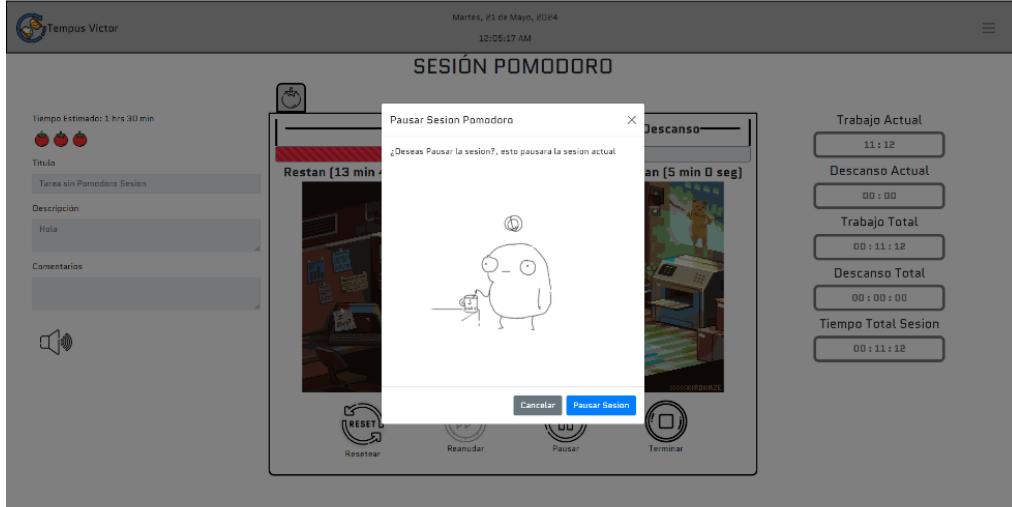
H.25 IU25 Modal Empezar Sesión Pomodoro

Objetivo	Diseño
Permite preguntar al usuario si desea empezar la sesión Pomodoro o no.	Modal que pregunta al usuario si empezar la sesión Pomodoro o no empezarla.
Entradas	
Botones:	
-Cancelar: Manda un mensaje de acción cancelada y lleva al usuario a la IU21 Pantalla de Temporizador Pomodoro	
-Empezar Sesión: Inicia la sesión Pomodoro, manda un mensaje de sesión Pomodoro iniciada y lleva al usuario a la IU22 Pantalla de Temporizador Pomodoro Corriendo	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Empezar la sesión?	
-Imagen gif de un personaje sosteniendo un Pomodoro, haciendo referencia a la técnica Pomodoro.	
-Mensaje informativo de inicio de sesión Pomodoro	
Imagen IU25 Modal para Empezar la Sesión Pomodoro	
	
Nota: Elaboración Propia	

H.26 IU26 Modal Reiniciar Sesión Pomodoro

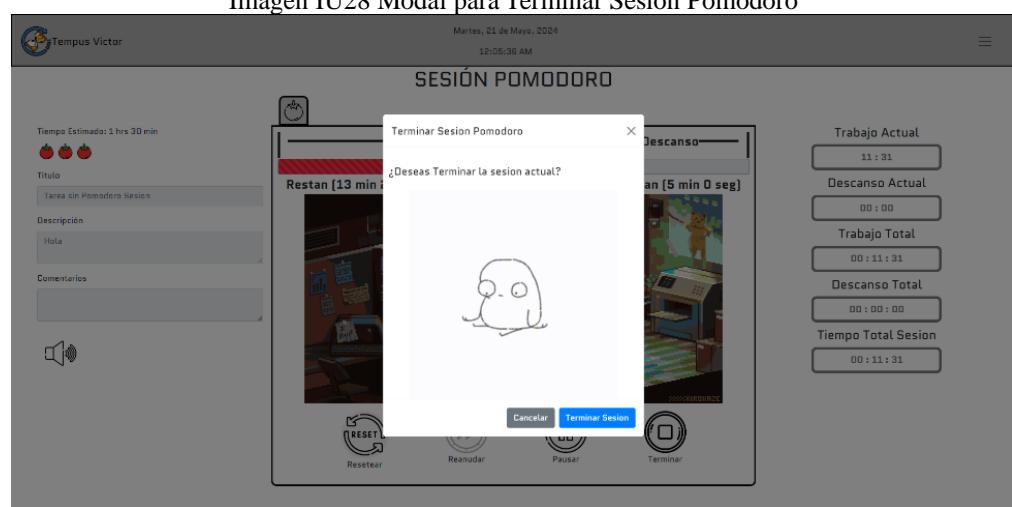
Objetivo	Diseño
Permite preguntar al usuario si desea reiniciar la sesión Pomodoro o no.	Modal que pregunta al usuario si reiniciar la sesión Pomodoro o no reiniciarla.
Entradas	
Botones:	
-Cancelar: Manda un mensaje de acción cancelada y lleva al usuario a la IU22 Pantalla de Temporizador Pomodoro Corriendo	
-Reiniciar Sesión: Reinicia la sesión Pomodoro, manda un mensaje de sesión Pomodoro reiniciada, y lleva al usuario a la IU21 Pantalla de Temporizador Pomodoro	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Reiniciar la sesión?	
-Imagen gif de un personaje pensativo pateando el suelo	
-Mensaje informativo de sesión reiniciada.	
Imagen IU26 Modal para Reiniciar Sesión Pomodoro	
	
Nota: Elaboración Propia	

H.27 IU27 Modal Pausar Sesión Pomodoro

Objetivo	Diseño
Permite preguntar al usuario si desea pausar la sesión Pomodoro o no.	Modal que pregunta al usuario si pausar la sesión Pomodoro o no pausarla.
Entradas	<p>Imagen IU27 Modal para Pausar la Sesión Pomodoro</p> 
Botones:	
-Cancelar: Manda un mensaje de acción cancelada y lleva al usuario a la IU21 Pantalla de Temporizador Pomodoro	
-Pausar Sesión: Pausa la sesión Pomodoro, manda un mensaje de sesión Pomodoro pausada y lleva al usuario a la IU21 Pantalla de Temporizador Pomodoro	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Pausar la sesión?, esto pausara a la sesión actual.	
-Imagen gif de un personaje tomando un descanso y preparándose un café.	
-Mensaje informativo de sesión pausada.	

Nota: Elaboración Propia

H.28 IU28 Modal Terminar Sesión Pomodoro

Objetivo	Diseño
Permite preguntar al usuario si desea terminar la sesión Pomodoro o no.	Modal que pregunta al usuario si termina la sesión Pomodoro o no pausarla.
Entradas	<p>Imagen IU28 Modal para Terminar Sesión Pomodoro</p> 
Botones:	
-Cancelar: Manda un mensaje de acción cancelada y lleva al usuario a la IU21 Pantalla de Temporizador Pomodoro	
-Terminar Sesión: Termina la sesión Pomodoro, manda un mensaje de sesión Pomodoro terminada, pregunta si también desea completar la tarea, y lleva al usuario a la IU4 Vista de Tareas Eisenhower	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Terminar la sesión actual?	
-Imagen gif de un personaje relajado y en paz	
-Mensaje informativo de sesión terminada.	

Nota: Elaboración Propia

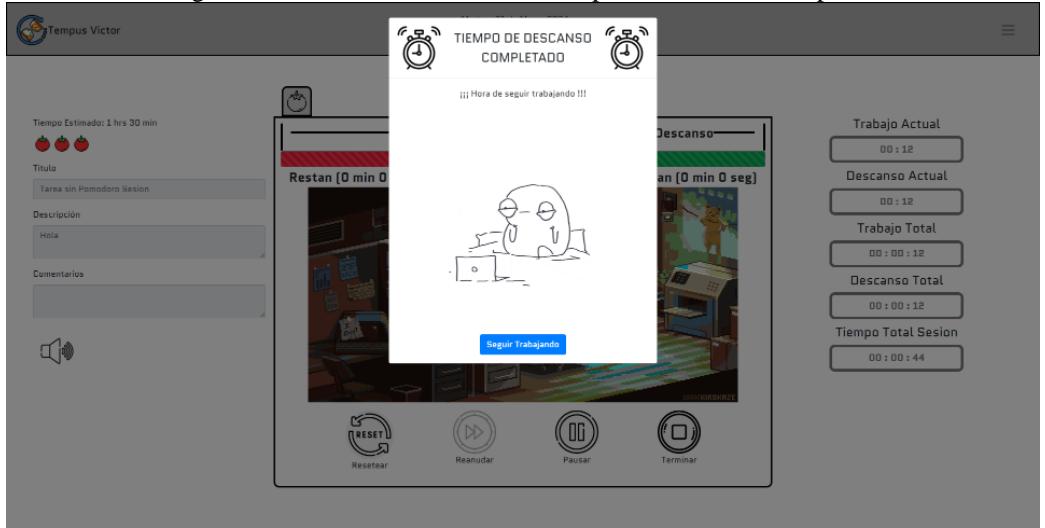
H.29 IU29 Modal Completar Tarea

Objetivo	Diseño
Permite preguntar al usuario si desea completar la tarea.	Modal que pregunta al usuario si desea Completar la Tarea una vez terminada la Sesión Pomodoro.
Entradas	Imagen IU29 Modal para Completar Tarea Despues de sesión Terminada
Botones: -No completar Tarea: Manda un mensaje de acción cancelada y lleva al usuario a la IU4 Vista de Tareas Eisenhower -Si completar Tarea: Cambia el estado de la tarea seleccionada a completada, manda un mensaje de éxito y lleva al usuario a la IU4 Vista de Tareas Eisenhower	
Salidas	
-Título del Modal -Pregunta ¿Tambiénquierescompletartarea? -Imagen gif de un personaje dudosoy pensativo -Mensaje informativo de tarea completada -Mensaje informativo de acción cancelada	Nota: Elaboración Propia

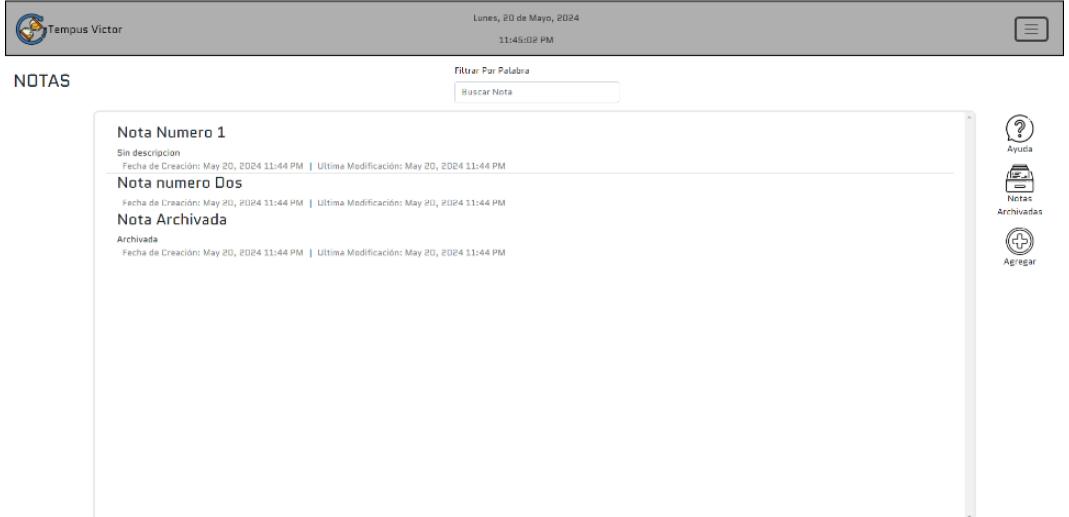
H.30 IU30 Modal Alarma Tiempo Trabajo Completado

Objetivo	Diseño
Permite notificar al usuario al finalizar el tiempo de trabajo y solicitar su interacción con el modal para empezar con el descanso.	Modal que avisa al Usuario con una alarma sonora, que el tiempo de trabajo se completó.
Entradas	Imagen IU30 Modal con Alarma de Tiempo de Trabajo Completado
Botones: -Empezar Descanso: Apaga la alarma y empieza el tiempo de descanso	
Salidas	
-Título del Modal -Sonido de alarma que se repite hasta que se dé clic en el botón. -Imagen gif de un personaje cansadoco ganas de descansar. -Aviso ¡¡¡ Hora de Tomar un descanso!!!	Nota: Elaboración Propia

H.31 IU31 Modal Alarma Tiempo Descanso Completado

Objetivo	Diseño
Permite notificar al usuario al finalizar el tiempo de descanso y solicitar su interacción con el modal para continuar con el trabajo	Modal que avisa al Usuario con una alarma sonora, que el tiempo de descanso se acabó.
Entradas	
Botones:	
-Seguir Trabajando: Apaga la alarma y continua el tiempo de trabajo.	
Salidas	
-Título del Modal -Sonido de alarma que se repite hasta que se dé clic en el botón. -Imagen gif de un personaje enfocado en su trabajo -Aviso !!! Hora de seguir trabajando!!!	<p style="text-align: center;">Imagen IU31 Modal con Alarma de Tiempo de Descanso Completado</p>  <p>Nota: Elaboración Propia</p>

H.32 IU32 Notas

Objetivo	Diseño
Permite al usuario ver todas las notas creadas en una lista de notas	Pantalla que muestra la lista de todas las notas creadas en la cuenta del usuario
Entradas	
Campo de texto: Filtro por palabra: Filtra la lista de notas dependiendo las letras que se ingresen en el campo de texto	
Botones: -Ayuda: Lleva a la IU32 Pantalla de Ayuda Notas -Notas Archivadas: Lleva a la IU33 Notas Archivadas -Agregar: Lleva a la IU34 Pantalla para Agregar Nota	<p style="text-align: center;">Imagen IU32 Pantalla de Notas</p>  <p>Nota: Elaboración Propia</p>
Salidas	
-Listas de notas, las cuales contienen título, descripción, fecha de creación y fecha de última modificación. -Mensaje de Información de cada elemento usando ToolTips.	

H.33 IU33 Seleccionar Nota

Objetivo	Diseño
Permitir al usuario interactuar con una nota seleccionada	Pantalla que muestra lo mismo que la Pantalla de Notas, más los botones habilitados para la Nota seleccionada
Entradas	
Botones:	
-Ayuda: Lleva a la IU32 Pantalla de Ayuda Notas	
-Notas Archivadas: Lleva a la IU33 Notas Archivadas	
-Agregar: Lleva a la IU34 Pantalla para Agregar Nota	
-Ver: Lleva a la IU35 Pantalla para Ver Nota	
-Editar: Lleva a la IU36 Pantalla para Editar Nota	
-Eliminar: Lleva a la IU37 Pantalla para Eliminar Nota	
-Archivar Nota: Lleva al IU38 Modal para archivar Nota	
Salidas	
-Listas de notas, las cuales contienen título, descripción, fecha de creación y fecha de última modificación.	
-Resalta a la Nota seleccionada de entre las demás.	
-Mensaje de Información de cada elemento usando ToolTips.	

Imagen IU33 Pantalla de Nota Seleccionada

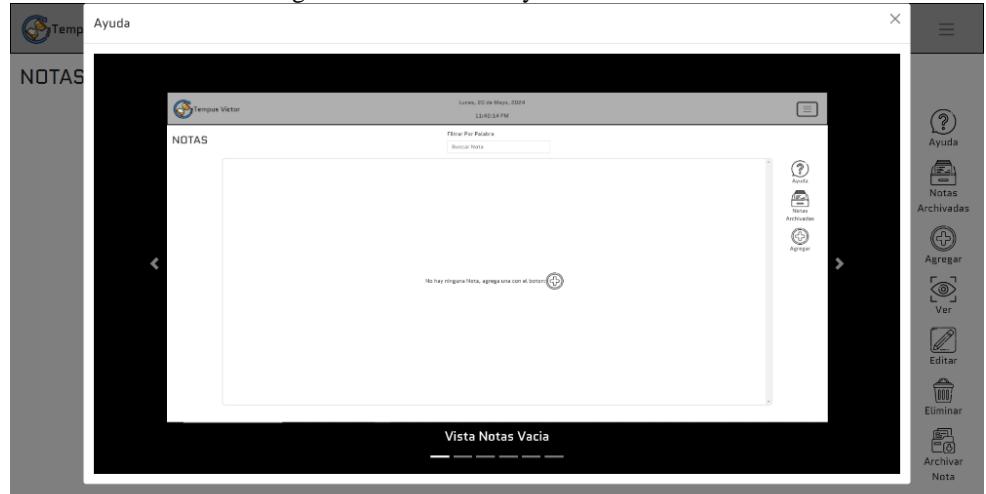


Nota: Elaboración Propia

H.34 IU34 Ayuda Notas

Objetivo	Diseño
Permitir al usuario obtener ayuda sobre la pantalla de Notas	Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en la pantalla de Notas
Entradas	
Botones:	
-Imagen anterior: Cambia la imagen actual a la anterior.	
-Imagen siguiente: Cambia la imagen actual a la siguiente.	
-Salir del carrusel: Sale del carrusel y lleva a la IU32 Pantalla de Notas	
Salidas	
Mensaje informativo cuando el usuario sale del carrusel.	

Imagen IU34 Modal de Ayuda sobre las Notas



Nota: Elaboración Propia

H.35 IU35 Notas Archivadas

Objetivo	Diseño
<p>Permite al Usuario visualizar en una lista las notas que tiene archivadas y que ya no quería ver en la lista de notas principal</p> <p>Entradas</p> <p>Campos de Texto:</p> <p>Filtro: Filtra por palabra dentro de la lista de notas archivadas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Regresar: Lleva a la IU32 Pantalla de Notas -Desarchivar Nota: Saca a la nota del archivo y manda un mensaje informativo sobre la actualización. <p>Salidas</p> <ul style="list-style-type: none"> -Mensaje de éxito cuando se saca correctamente la nota del archivo -Mensaje de Información de cada elemento usando ToolTips. 	<p>Pantalla que contiene una lista de notas que el usuario archiva, dando la posibilidad de usar un botón para sacar alguna nota seleccionada del archivo.</p> <p>Imagen IU35 Pantalla de Notas Archivadas</p>
	<p>Nota: Elaboración Propia</p>

H.36 IU36 Agregar Nota

Objetivo	Diseño
<p>Permite al Usuario agregar una Nota nueva a la lista de Notas</p> <p>Entradas</p> <p>Campos de Texto:</p> <p>Título y descripción de la nota.</p> <p>Botones:</p> <ul style="list-style-type: none"> -Cancelar: Lleva a la IU32 Pantalla de Notas -Aceptar: Manda los datos al servidor agregando una nueva nota en la base de datos del usuario. <p>Salidas</p> <ul style="list-style-type: none"> -Mensaje de Error cuando no se cumplen los parámetros necesarios de la nota a crear. -Mensaje de éxito cuando se crea la nota correctamente -Mensaje de Información de cada elemento usando ToolTips. 	<p>Pantalla que pide agregar título y descripción para una nota que el usuario quiera crear.</p> <p>Imagen IU36 Pantalla para Agregar Nota</p>
	<p>Nota: Elaboración Propia</p>

H.37 IU37 Ver Nota

Objetivo	Diseño
Permite al usuario ver por completo a la nota previamente seleccionada	En esta pantalla se puede ver todas las características de la tarea seleccionada, el título, descripción, fecha y hora de creación, fecha y hora de última modificación.
Entradas	
Botones:	Imagen IU37 Pantalla para Ver Nota
-Regresar: Lleva a la IU32 Pantalla de Notas	
Salidas	
Campo de Texto:	
-Título de nota, descripción de notas.	
-Mensaje de Información de cada elemento usando ToolTips.	

H.38 IU38 Editar Nota

Objetivo	Diseño
Permite al usuario editar la nota previamente seleccionada.	Pantalla que muestra la información de la nota seleccionada y da la posibilidad al usuario de editar su contenido.
Entradas	
Campos de Texto:	Imagen IU38 Pantalla para Editar Nota
Título de la nota, descripción de la nota.	
Botones:	
-Cancelar: Lleva a la IU32 Pantalla de Notas	
-Aceptar: Manda los datos al servidor el cual actualiza los datos de la nota seleccionada.	
Salidas	
-Mensaje de éxito al actualizar la nota correctamente.	
-Mensaje de Información de cada elemento usando ToolTips.	

Nota: Elaboración Propia

H.39 IU39 Eliminar Nota

Objetivo	Diseño
Permite al usuario eliminar la nota previamente seleccionada	Pantalla que contiene la información de la nota seleccionada y da la posibilidad de eliminarla.
Entradas	
Botones:	Imagen IU39 Pantalla para Eliminar Nota
-Regresar: Lleva al usuario a la IU32 Pantalla de Notas -Eliminar: Manda la solicitud al servidor de eliminar la nota, se elimina y se lleva al usuario a la lista de notas.	
Salidas	
-Mensaje de Éxito si se elimina la nota correctamente. -Mensaje informativo si se decide no eliminar la nota. -Mensaje de Información de cada elemento usando ToolTips.	

H.40 IU40 Modal Archivar Nota

Objetivo	Diseño
Permite preguntar al usuario archivar o no archivar la nota previamente seleccionada.	Modal que pregunta al usuario si archivar o no la nota seleccionada.
Entradas	
Botones:	Imagen IU40 Modal para Archivar una Nota
-Cancelar: Manda un mensaje de cancelación de acción y lleva al usuario a la IU32 Pantalla de Notas. -Archivar Nota: Archiva la nota, mandándola a la lista de notas archivadas, y actualiza la IU32 Pantalla de Notas.	
Salidas	
-Título del Modal -Pregunta ¿Quieres Archivar la Nota seleccionada? -Imagen gif de un personaje dudoso y pensativo -Mensaje informativo si se canceló la acción -Mensaje de éxito si se archivó la nota correctamente	

H.41 IU41 Cuenta

Objetivo	Diseño
<p>Permite al usuario ver la información de su cuenta dando la opción de gestionar la contraseña, la foto de perfil y el nombre de usuario.</p>	<p>La pantalla contiene la foto de perfil de la Cuenta, el nombre de usuario, el correo electrónico y los botones para gestionar la cuenta.</p>
<p>Entradas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Cerrar Sesión: Lleva a la IU42 Modal para Cerrar Sesión -Editar: Lleva a la IU43 Pantalla para Editar Cuenta -Cambiar Contraseña: Lleva a la IU44 Modal para Cambiar Contraseña -Eliminar Cuenta: Lleva a la IU45 Modal para Eliminar Cuenta <p>Salidas</p> <ul style="list-style-type: none"> -Mensaje de Información de cada elemento usando ToolTips. 	<p>Imagen IU41 Pantalla de Cuenta</p> <p>Nota: Elaboración Propia</p>

H.42 IU42 Cerrar Sesión

Objetivo	Diseño
<p>Permite al usuario cerrar sesión en su cuenta y salir del sistema.</p>	<p>El modal muestra una pregunta un gif y los botones para cerrar sesión o regresar</p>
<p>Entradas</p> <p>Botones:</p> <ul style="list-style-type: none"> -Regresar: Lleva a la IU41 Pantalla de Cuenta -Cerrar Sesión: Manda la solicitud al sistema de cerrar la sesión del usuario actual, cierra sesión y regresa al usuario a la IU41 Pantalla de Cuenta <p>Salidas</p> <ul style="list-style-type: none"> -Título del Modal -Pregunta ¿Quieres Cerrar Sesión y Salir del Sistema? -Imagen gif de un personaje que está en su cama, descansando junto a su gato. -Mensaje informativo si se canceló la acción -Mensaje de éxito si se cerró correctamente la sesión -Mensaje de Información de cada elemento usando ToolTips. 	<p>Imagen IU42 Modal para Cerrar Sesión</p> <p>Nota: Elaboración Propia</p>

H.43 IU43 Editar Cuenta

Objetivo	Diseño
Permitir al usuario cambiar su foto de perfil y su nombre de usuario.	La pantalla muestra la opción de cambio de foto de perfil y el nombre de usuario.
Entradas	
Campos de texto: Nombre de usuario, confirmar nombre de usuario. Botones: -Cancelar: Lleva a la IU41 Pantalla de Cuenta -Aceptar: Guarda la foto de perfil y el nombre de usuario si es que se cambió, luego lleva al usuario a la IU41 Pantalla de Cuenta. -Cambiar foto de perfil: Abre un selector de archivos que solo permite imágenes.	<p style="text-align: center;">Imagen IU43 Pantalla para Editar Cuenta</p> <p style="text-align: center;">Nota: Elaboración Propia</p>
Salidas	
-Mensaje de éxito si se actualizan correctamente la foto de perfil y el nombre de usuario. -Mensaje de Información de cada elemento usando ToolTips.	

H.44 IU44 Cambiar Contraseña

Objetivo	Diseño
Permite al usuario cambiar su contraseña anterior por una nueva.	La pantalla contiene la foto de perfil del usuario y los campos necesarios para realizar el cambio de contraseña.
Entradas	
Campos de texto: Nueva contraseña, confirmar contraseña. Botones: -Cancelar: Lleva a la IU41 Pantalla de Cuenta -Aceptar: Manda los datos de la contraseña al servidor el cual realiza la actualización a la cuenta actual, y sale del sistema llevando al usuario a la IU1 Pantalla de Bienvenida al Sistema	<p style="text-align: center;">Imagen IU44 Modal para Cambiar Contraseña</p> <p style="text-align: center;">Nota: Elaboración Propia</p>
Salidas	
-Título del Modal -Pregunta ¿Quieres Cambiar Tu contraseña? -Imagen gif de un personaje que está pensando en una nueva contraseña. -Mensaje informativo si se canceló la acción -Mensaje de éxito si se actualiza correctamente la contraseña. -Mensaje de Información de cada elemento usando ToolTips.	

H.45 IU45 Eliminar Cuenta

Objetivo	Diseño
Permitir al usuario eliminar la cuenta y todos los datos relacionados con ella, las tareas, las notas y las sesiones Pomodoros.	La pantalla contiene la foto de perfil, el nombre de usuario y el correo electrónico, así como la pregunta final sobre eliminar su cuenta y los botones que permiten esta acción.
Entradas	
Botones:	
-Regresar: Lleva a la IU41 Pantalla de Cuenta -Eliminar Cuenta: Lleva al IU46 Modal para Confirmar Eliminar Cuenta	<p style="text-align: center;">Imagen IU45 Modal para Eliminar Cuenta</p> 
Salidas	
-Título del Modal -Pregunta ¿Quieres Eliminar Completamente tu Cuenta? -Imagen gif de un personaje que está ansioso. -Mensaje informativo si se canceló la acción -Mensaje informativo de que se quiere eliminar la cuenta. -Mensaje de Información de cada elemento usando ToolTips.	

H.46 IU46 Confirmar Eliminar Cuenta

Objetivo	Diseño
Permitir al usuario eliminar la cuenta y todos los datos relacionados con ella, las tareas, las notas y las sesiones Pomodoros.	La pantalla contiene la foto de perfil, el nombre de usuario y el correo electrónico, así como la pregunta final sobre eliminar su cuenta y los botones que permiten esta acción.
Entradas	
Botones:	
-Regresar: Lleva a la IU41 Pantalla de Cuenta -Confirmar Eliminar Cuenta: Manda la solicitud de eliminar cuenta actual, elimina todos los datos de la Cuenta, y lleva al usuario a la IU1 Pantalla de Bienvenida al Sistema	<p style="text-align: center;">Imagen IU46 Modal para Confirmar Eliminar Cuenta</p> 
Salidas	
-Título del Modal -Pregunta ¿Confirms Eliminar tu Cuenta? -Imagen gif de un personaje que está pensando triste a punto de llora. -Mensaje informativo si se canceló la acción -Mensaje de advertencia sobre eliminar la cuenta -Mensaje de éxito si se elimina correctamente la cuenta. -Mensaje de Información de cada elemento usando ToolTips.	

H.47 IU47 Configuración

Objetivo	Diseño
Permite al usuario gestionar la configuración del sistema, específicamente la configuración del temporizador Pomodoro.	La pantalla contiene los elementos que permiten gestionar la duración del tiempo de trabajo, los tiempos de descanso, las imágenes de fondo del temporizador dependiendo su estado, el sonido de fondo y el sonido de la alarma del temporizador.
Entradas	
Campos de texto: Duración de trabajo, duración descanso corto, duración descanso largo. Campos de selección: Sonido de alarma, sonido de fondo. Carrusel de imágenes: Carrusel estado abandonada, carrusel estado trabajando, carrusel estado descansando. Botones: -Resetear valores por defecto: Cambia los valores actuales por los valores por defecto del sistema -Guardar valores actuales: Guarda los valores actuales en la base de datos del usuario actual.	
Salidas	
-Mensaje de éxito si se resetearon los valores correctamente. -Mensaje de éxito si se guardaron los valores correctamente. -Mensaje de Información de cada elemento usando ToolTips	<p style="text-align: center;">Imagen IU47 Pantalla de Configuración del Sistema</p>
	Nota: Elaboración Propia

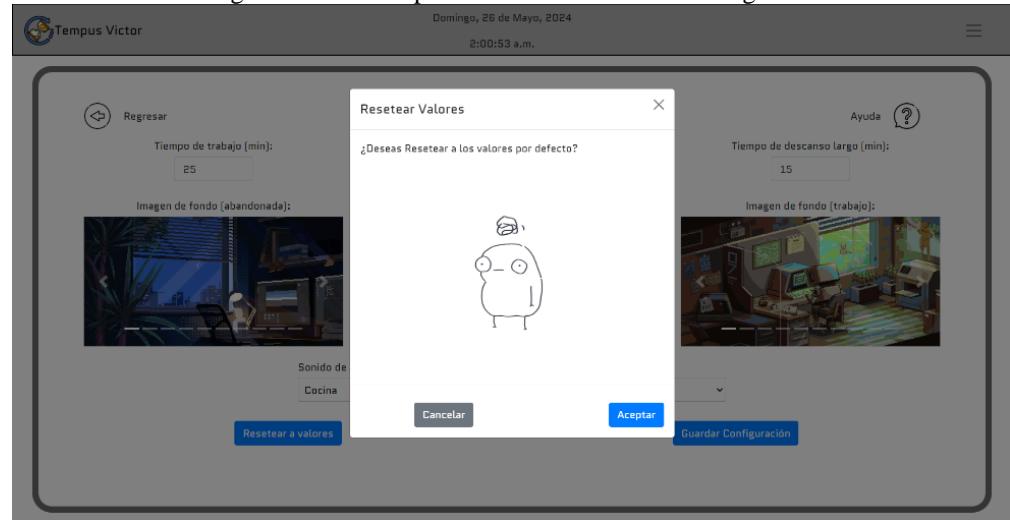
H.48 IU48 Ayuda Configuración

Objetivo	Diseño
Permitir al usuario obtener ayuda sobre la pantalla de Configuración.	Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en la pantalla de Configuración.
Entradas	
Botones: -Imagen anterior: Cambia la imagen actual a la anterior. -Imagen siguiente: Cambia la imagen actual a la siguiente. -Salir del carrusel: Sale del carrusel y lleva a la IU47 Pantalla de Configuración del Sistema	<p style="text-align: center;">Imagen IU48 Modal de Ayuda sobre la Configuración</p>
Salidas	
Mensaje informativo cuando el usuario sale del carrusel.	Nota: Elaboración Propia

H.49 IU49 Resetear Valores Configuración

Objetivo	Diseño
Permitir al usuario resetear los valores de configuración que tiene por los valores por defecto del sistema	El modal contiene la pregunta sobre el reseteo de valores, un gif y los botones que permiten la acción del modal.
Entradas	
Botones:	
-Cancelar: Lleva a la IU47 Pantalla de Configuración del Sistema	
-Aceptar: Resetea los valores de configuración por los valores por defecto del Sistema y manda un mensaje de éxito, luego lleva al usuario de regreso a la IU47 Pantalla de Configuración del Sistema	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Resetear a los Valores por defecto?	
-Imagen gif de un personaje que está dudando y pensativo	
-Mensaje informativo si se canceló la acción	
-Mensaje de éxito si se resetean correctamente los valores.	
-Mensaje de Información de cada elemento usando ToolTips.	

Imagen IU49 Modal para Resetear Valores de Configuración

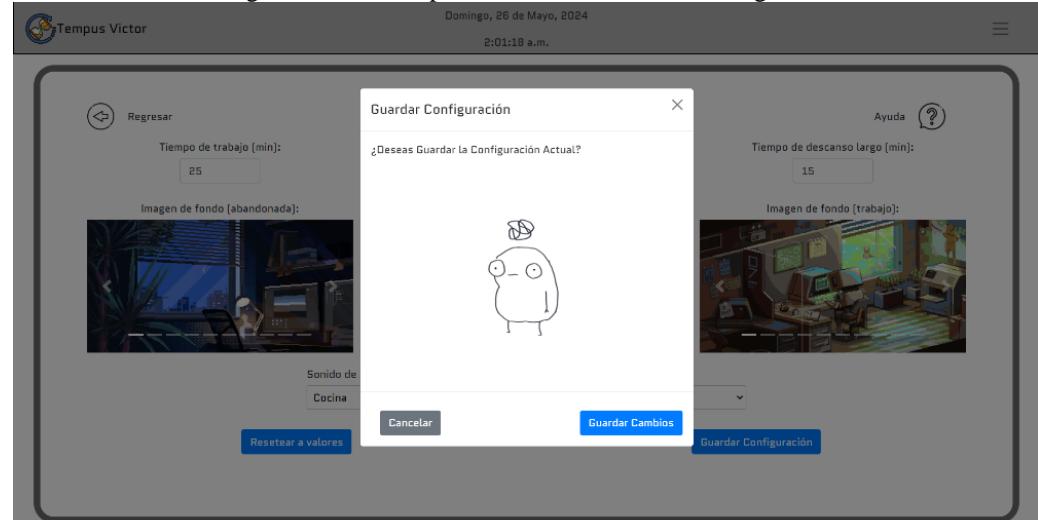


Nota: Elaboración Propia

H.50 IU50 Guardar Valores Configuración

Objetivo	Diseño
Permitir al usuario guardar los valores de configuración que tiene actualmente.	El modal contiene la pregunta sobre guardar la configuración, un gif y los botones que permiten la acción del modal.
Entradas	
Botones:	
-Cancelar: Lleva a la IU47 Pantalla de Configuración del Sistema	
-Guardar Cambios: Guarda los valores actuales de la pantalla, manda un mensaje de éxito, luego lleva al usuario de regreso a la IU47 Pantalla de Configuración del Sistema	
Salidas	
-Título del Modal	
-Pregunta ¿Deseas Guardar la Configuración Actual?	
-Imagen gif de un personaje que está dudando y pensativo	
-Mensaje informativo si se canceló la acción	
-Mensaje de éxito si se guardaron correctamente los valores.	
-Mensaje de Información de cada elemento usando ToolTips.	

Imagen IU50 Modal para Guardar Valores de Configuración



Nota: Elaboración Propia

H.51 IU51 Métricas

Objetivo	Diseño
Permite al usuario ver las métricas de su cuenta con relación a las tareas y a las sesiones Pomodoro.	Esta pantalla contiene los diagramas de pastel que muestran las distintas proporciones de las tareas y sesiones Pomodoro que contiene la cuenta del usuario.
Entradas	
Botones:	
-Regresar: Lleva a la IU4 Vista de Tareas Eisenhower	 Imagen IU51 Pantalla de Métricas de la Cuenta. La pantalla muestra dos gráficos de pastel: uno para 'Tareas' (rojo para 'Tareas Completadas' y azul para 'Tareas Sin Completar') y otro para 'Pomodoro' (verde para 'Sesiones Sin Iniciar', azul para 'Sesiones Pausadas' y naranja para 'Sesiones Terminadas'). A la derecha de cada gráfico hay una tabla de datos detallados. La parte superior de la pantalla incluye el logo 'Tempus Victor', la fecha 'Lunes, 20 de Mayo, 2024' y la hora '11:47:38 PM'.
Salidas	
-Mensaje de Información de cada elemento usando ToolTips -Grafica de pastel sobre Tareas -Grafica de pastel sobre Sesiones Pomodoro -Una lista de cantidades sobre tareas y sesiones Pomodoro.	

H.52 IU52 Ayuda Métricas

Objetivo	Diseño
Permitir al usuario obtener ayuda sobre la pantalla de Métricas	Pantalla que contiene un carrusel de imágenes, consejos e instrucciones de lo que se puede hacer en la pantalla de Métricas.
Entradas	
Botones:	
-Imagen anterior: Cambia la imagen actual a la anterior. -Imagen siguiente: Cambia la imagen actual a la siguiente. -Salir del carrusel: Sale del carrusel y lleva a la IU51 Pantalla de Métricas de la Cuenta	 Imagen IU52 Modal de Ayuda sobre las Métricas. Un modal central muestra una captura de la IU51 con un mensaje de error: '¡¡¡ No hay ninguna Tarea, crea una en alguna Vista de tareas !!!'. Al lado del modal, en la parte superior, hay un encabezado 'Ayuda' y un ícono de temporizador. En la parte inferior del modal, se indica 'Metricas Vacias, Sin ningun dato'.
Salidas	
Mensaje informativo cuando el usuario sale del carrusel.	