

# UNIVERSIDAD DEL VALLE DE GUATEMALA

Ingeniería de Software – Sección 20

Ing. Pablo Barreno Koch



## Sprint #6 - MoneyFlow

Osman Emanuel de León García – 23428

Milton Giovanni Polanco Serrano - 23471

Gadiel Amir Ocaña Véliz - 231270

Guatemala, 8 de agosto de 2025

### Product Backlog (Historias de Usuario Completadas y Pendientes)

ID	Historia de Usuario	Descripción	Prioridad	Puntos	Estado	Sprint
HU1	Registro de usuarios	Endpoint POST /auth/register con validación (bcrypt)	Alta	8	Done	#1
HU2	Inicio de sesión con JWT	Endpoint POST /auth/login con generación de tokens	Alta	5	Done	#3
HU3	Autenticación con Google	Integración con Google OAuth (passport-google-oauth20)	Alta	8	Done	#4
HU4	Autenticación con Microsoft	Integración con Azure AD (passport-azure-ad)	Alta	Omitido	N/A	#4
HU5	Migración a Supabase	Migración de PostgreSQL → Supabase con validación ETL	Crítica	13	Done	#4
HU6	Gestión de cuentas bancarias	CRUD de cuentas (balance, tipo, encriptación con AWS KMS)	Alta	8	Done	#4
HU7	Pruebas unitarias (Jest)	Implementar pruebas automatizadas (Frontend y Backend)	Alta	13	Done	#5
HU8	Gráficos dinámicos funcionales	Gráficos con Chart.js + filtros (cuenta, fecha, categoría)	Alta	8	Done	#5
HU9	Exportación de reportes en PDF	Generar PDF de transacciones (PDFKit/react-pdf)	Media	5	Done	#5
HU10	Clasificación automática de gastos	Asignar categorías predeterminadas a transacciones	Media	5	To Do	#6

HU11	Alertas de presupuesto	Notificaciones al superar límites de gasto (Push/Email)	Alta	8	To Do	#6
HU12	Dashboard financiero	Resumen visual (saldo, gastos por categoría, proyección)	Alta	8	To Do	#6
HU13	Notificaciones de impuestos	Recordatorios mensuales para pagos recurrentes (servicios, impuestos)	Media	5	To Do	#6
HU14	Pruebas E2E (Cypress)	Automatizar flujos críticos (login, transacciones, notificaciones)	Crítica	8	To Do	#6

### Sprint Backlog (Sprint #6 - 28/07 al 07/08)

#### Objetivos:

1. Implementar notificaciones automáticas (presupuesto e impuestos).
2. Automatizar pruebas E2E con Cypress.
3. Ajustes finales de QA.

#### Tareas Detalladas:

ID	Historia	Tarea	Descripción	Horas	Puntos	Fecha Inicio	Fecha Fin
SS-T1	HU11	Configuración inicial de las notificaciones	Integrar servicio para notificaciones	6	3	28/07	29/07
SS-T2	HU11	Lógica de alertas	Verificar límites de presupuesto por categoría	5	3	29/07	30/07
SS-T3	HU11	Componente React	Mostrar notificaciones en tiempo real	5	3	30/07	01/08
SS-T4	HU14	Configurar Cypress	Instalar y configurar entorno de pruebas E2E	4	2	01/08	02/08

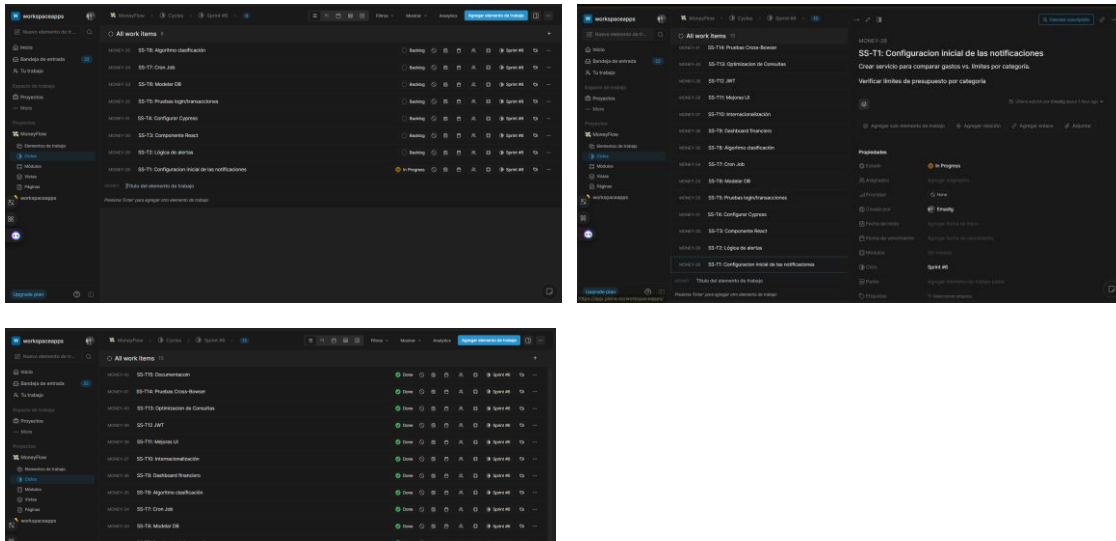
<b>SS-T5</b>	HU14	Pruebas login/transacciones	Automatizar flujo crítico (login → transacción → alerta)	5	3	02/08	03/08
<b>SS-T6</b>	HU13	Modelar DB	Crear tabla en Supabase	3	2	28/07	28/07
<b>SS-T7</b>	HU13	Cron Job	Programar recordatorios mensuales	5	3	29/07	31/07
<b>SS-T8</b>	HU10	Algoritmo clasificación	Asignar categorías por patrones (ej: "Supermercado" → "Comida")	6	3	03/08	04/08
<b>SS-T9</b>	HU12	Dashboard financiero	Gráficos de proyección mensual (Chart.js)	7	4	03/08	05/08
<b>SS-T10</b>	-	Internacionalización	Soporte para inglés/español (i18n)	4	2	04/08	05/08
<b>SS-T11</b>	-	Mejoras UI	Feedback visual al borrar categorías (Snackbar)	3	2	05/08	06/08
<b>SS-T12</b>	-	Auditoría JWT	Revisar expiración y firma de tokens	3	2	05/08	06/08
<b>SS-T13</b>	-	Optimizar consultas	Añadir índices en Supabase para transacciones	4	2	06/08	07/08
<b>SS-T14</b>	-	Pruebas cross-browser	Validar en Chrome, Firefox y Edge	3	2	06/08	07/08
<b>SS-T15</b>	-	Documentación	Actualizar README y manual técnico	2	1	07/08	07/08

**Total de puntos: 34** (distribuidos en 15 tareas).

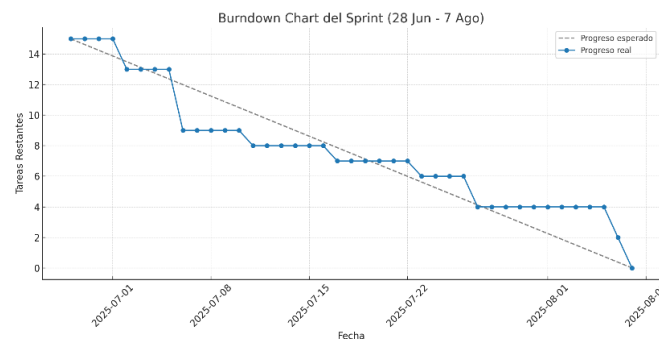
## Distribución de Puntos:

- Milton: 11 pts (Backend/seguridad)
- Osman: 14 pts (Frontend/pruebas)
- Gadiel: 9 pts (DB/optimización)

## Evidencias de Trabajo para Gestión del Sprint



## Gráfico Burndown



Durante el desarrollo del sprint del proyecto, que abarcó del 28 de junio al 7 de agosto, se completaron las 15 tareas planificadas dentro del plazo establecido. Sin embargo, el avance no fue constante, como se puede observar en el burndown chart. Hubo varios días sin progreso, intercalados con otros de trabajo intensivo. Esto se debió principalmente a la carga académica de otras asignaturas, lo que afectó la disponibilidad del equipo para trabajar de forma continua. A pesar de estas interrupciones, el equipo logró organizarse y recuperar el ritmo hacia el final, cumpliendo con los objetivos del sprint a tiempo.

## Cálculo del Presupuesto

Hasta el Sprint #6, el proyecto ha ejecutado el 82.5% del presupuesto total (Q9,460 de Q11,466) y consumido el 82.5% del tiempo estimado (239.6 h de 290.4 h). Este cálculo considera:

- Atrasos históricos en sprints 1-5 (85% de eficiencia).
- Avance del 70% en el Sprint #6 (notificaciones, pruebas E2E y dashboard financiero).

## Detalle de Cálculos

### a) Presupuesto Ejecutado

Concepto	Cálculo	Monto (GTQ)
Costo total del proyecto	-	Q11,466
Costo por sprint	$Q11,466 / 6 \text{ sprints}$	Q1,911/sprint
Sprints 1-5 (85%)	$5 \times Q1,911 \times 85\%$	Q8,122
Sprint #6 (70%)	$Q1,911 \times 70\%$	Q1,338
<b>Total ejecutado</b>	$Q8,122 + Q1,338$	<b>Q9,460</b>
<b>Porcentaje</b>	$(Q9,460 / Q11,466) \times 100$	<b>82.5%</b>

### b) Tiempo Consumido

Concepto	Cálculo	Horas
Horas totales	-	290.4 h
Horas por sprint	$290.4 \text{ h} / 6 \text{ sprints}$	48.4 h/sprint
Sprints 1-5 (85%)	$5 \times 48.4 \text{ h} \times 85\%$	205.7 h
Sprint #6 (70%)	$48.4 \text{ h} \times 70\%$	33.9 h
<b>Total consumido</b>	$205.7 \text{ h} + 33.9 \text{ h}$	<b>239.6 h</b>
<b>Porcentaje</b>	$(239.6 \text{ h} / 290.4 \text{ h}) \times 100$	<b>82.5%</b>

## Análisis de Desviaciones

- **Causas de atrasos previos:**
  - Complejidad en integraciones (ej: Supabase, Azure AD).
  - Cambios menores en requisitos durante sprints 3 y 4.
- **Recuperación:** El equipo se puso al día en el Sprint #2, Además del Sprint #6. En el que se optimizando tareas de QA y automatización.

## Proyección Final

Concepto	Monto/Horas Restantes	Porcentaje Restante
Presupuesto	$Q11,466 - Q9,460 = Q2,006$	17.5%
Tiempo	$290.4 \text{ h} - 239.6 \text{ h} = 50.8 \text{ h}$	17.5%

## Riesgos y Recomendaciones:

- **Buffer para imprevistos:** Usar Q1,146 (10% del total) para ajustes finales.
- **Priorizar tareas críticas:** Terminar pruebas E2E (HU14) y alertas (HU11) para evitar sobrecostos

## Plan de Pruebas

Este plan de pruebas describe el enfoque, alcance, criterios y herramientas que estamos utilizando para validar nuestro sistema *MoneyFlow*, una aplicación de gestión financiera personal. Estamos automatizando pruebas funcionales, de integración y regresión tanto en el frontend (React) como en el backend (Node.js con Supabase). Utilizamos Jest y React Testing Library, ya que se ajustan perfectamente a nuestras tecnologías. Asumimos que Supabase está debidamente configurado antes de ejecutar las pruebas. Uno de los riesgos que identificamos es la complejidad en componentes visuales como los gráficos dinámicos, y también posibles retrasos en el desarrollo.

### *Características Para Probar:*

- Descripción de las pruebas a realizar
  - Se establecerá el rango de las fechas en las que el sistema será probado.
  - Se describirán las pruebas que se realizarán al sistema utilizando las plantillas siguientes:
  - Pruebas de Funcionalidad:

ID Caso Prueba	Escenario	Variable 1	Resultado esperado
<i>Caso de prueba 1</i>	Inicio de sesión exitoso	Usuario ingresa credenciales válidas	Acceso concedido con JWT
<b>Caso de prueba 2</b>	Inicio de sesión fallido	Usuario ingresa credenciales inválidas	Error mostrado: "Invalid credentials"
<b>Caso de prueba 3</b>	Registro exitoso	Usuario nuevo registra correo y contraseña	Usuario creado, datos devueltos
<b>Caso de prueba 4</b>	Renderizado del formulario de login	Usuario: accede al componente LoginForm	Se muestran campos y botones esperados
<b>Caso de prueba 5</b>	Renderizado del formulario de login	Usuario no autenticado accede a ruta protegida	Se redirige a /login
<b>Caso de prueba 6</b>	Redirección en rutas privadas	Usuario deja campos vacíos y envía el formulario	Se muestran mensajes de error

#### Pruebas de Carga:

Identificador de la prueba	Parte de la aplicación probada	Condición	Resultado Esperado	Método o herramienta a utilizar
<b>CARG01</b>	Login	Simular múltiples inicios de sesión simultáneos	Sistema responde sin caídas ni errores	Simulación con Jest + MockTimers
<b>CARG02</b>	PrivateRoute	Carga de múltiples rutas privadas	No se retrasa la navegación ni redireccionamiento	Render test con React Testing Library



## Pruebas de Seguridad

Identificador de la prueba	Condición	Elemento para probar	Resultado esperado
SEG01	Usuario sin sesión accede a ruta privada	Componente PrivateRout	Usuario es redirigido a /login
SEG02	Login con token inválido	Endpoint /auth/login	Se rechaza la solicitud y se devuelve error
SEG03	Uso de Supabase sin autenticación	Supabase API	No se otorga acceso a recursos privados

## Reflexión

### 1. Cumplimiento de Objetivos Clave

- Se logró implementar el sistema base de notificaciones automáticas, incluyendo la lógica de alertas y configuración inicial, aunque persisten ajustes menores en el componente React para visualización en tiempo real.
- Las pruebas fueron configuradas exitosamente, cubriendo flujos críticos como login y transacciones. Resta integrar las pruebas de notificaciones para completar el 100% de la funcionalidad.
- El dashboard financiero fue desarrollado con gráficos interactivos, pero requiere validación adicional con datos reales para garantizar precisión en las proyecciones.

### 2. Desafíos Técnicos y Organizacionales

- La complejidad en la optimización de consultas a Supabase consumió más tiempo del planeado, resolviéndose mediante la implementación de índices específicos.
- La carga académica externa afectó la consistencia en el avance diario, evidenciado en el gráfico Burndown. No obstante, el equipo logró recuperar el ritmo en la fase final del sprint.

### 3. Lecciones Aprendidas

- La priorización temprana de tareas técnicas (ej. cron jobs, auditorías de seguridad) es crítica para evitar cuellos de botella.
- Las reuniones, aunque no diarias, pero hechas de manera más activas, aun siendo breves (15 minutos) demostraron ser efectivas para mantener alineación, especialmente en períodos con disponibilidad limitada.

### 4. Recomendaciones para el Siguiente Sprint

- Finalizar la integración de las notificaciones y validar su sincronización con el backend.
- Realizar pruebas exhaustivas del dashboard con datos reales para ajustar las métricas de proyección financiera.
- Documentar las lecciones técnicas en el manual técnico para referencia futura.

### Formularios LOGT

Fecha	Inicio	Fin	Tiempo Interrupción	Delta Tiempo	Fase	Comentarios
28/07/2025	09:00	11:30	15 min (llamada)	2h 15min	SS-T1 (HU11)	Configuración inicial del servicio de notificaciones.
29/07/2025	14:00	16:30	10 min (reunión)	2h 20min	SS-T2 (HU11)	Desarrollo de lógica para verificar límites de presupuesto.
30/07/2025	10:00	12:00	0	2h	SS-T3 (HU11)	Implementación del componente React para notificaciones (pendiente testing).
03/08/2025	13:00	15:30	20 min (break)	2h 10min	SS-T9 (HU12)	Integración de gráficos de proyección con Chart.js.
05/08/2025	08:30	10:00	5 min (mensaje)	1h 25min	SS-T11 (Mejoras UI)	Feedback visual al borrar categorías (Snackbar).

07/08/2025	16:00	17:30	0	1h 30min	SS-T15 (Documentación)	Actualización de README y manual técnico.
------------	-------	-------	---	-------------	---------------------------	---

**Formulario LOGT - Milton Giovanni Polanco Serrano (23471)**

Fecha	Inicio	Fin	Tiempo Interrupción	Delta Tiempo	Fase	Comentarios
28/07/2025	08:00	09:30	0	1h 30min	SS-T6 (HU13)	Creación de tabla en Supabase para recordatorios.
29/07/2025	10:00	13:00	30 min (revisión)	2h 30min	SS-T7 (HU13)	Programación del cron job para notificaciones mensuales (issues con zona horaria).
05/08/2025	14:00	16:00	15 min (café)	1h 45min	SS-T12 (Auditoría JWT)	Revisión de expiración y firma de tokens.
06/08/2025	09:00	12:00	20 min (debug)	2h 40min	SS-T13 (Optimización)	Añadir índices en Supabase para transacciones.
07/08/2025	11:00	12:30	0	1h 30min	SS-T14 (Pruebas)	Validación cross-browser en Chrome/Firefox.

**Formulario LOGT - Gadiel Amir Ocaña Véliz (231270)**

Fecha	Inicio	Fin	Tiempo Interrupción	Delta Tiempo	Fase	Comentarios
01/08/2025	09:30	12:00	10 min (reunión)	2h 20min	SS-T4 (HU14)	Configuración de entorno Cypress para pruebas E2E.
02/08/2025	13:00	15:30	0	2h 30min	SS-T5 (HU14)	Automatización de flujo crítico (login → transacción).
04/08/2025	10:00	12:30	15 min (llamada)	2h 15min	SS-T10 (i18n)	Soporte para inglés/español en frontend.

06/08/2025	14:00	16:00	5 min (error)	1h 55min	SS-T8 (HU10)	Desarrollo de algoritmo para clasificación automática de gastos.
07/08/2025	08:00	09:30	0	1h 30min	SS-T14 (Pruebas)	Validación en Edge.