

Taller: Submódulos, Librerías y APIs

Fecha: 4 de agosto de 2025
Estudiante: Maximiliano Madrid Benavides
Curso: [2B]

Taller 1: "Mi ecosistema personal como repositorio Git"

Propósito

Analizar mi rutina diaria aplicando conceptos de control de versiones, identificando módulos independientes, actualizaciones constantes y áreas que requieren refactorización para optimizar mi gestión personal.

Arquitectura de mi sistema personal

Mi día a día se estructura como un repositorio con múltiples módulos interconectados:

Módulo	Descripción	Frecuencia
academics/	Actividades universitarias: clases, proyectos, estudios	Diario
fitness/	Rutina de ejercicio y actividad física	4-5 veces/semana
social/network/	Interacciones con amigos y familia	Variable
hobbies/reading/	Lectura de libros técnicos y ficción	Nocturno
work/freelance/	Proyectos independientes de programación	Fines de semana
maintenance/	Tareas domésticas y autocuidado	Según necesidad

Mi framework personal: task-manager-system

He desarrollado un sistema híbrido entre Trello y Google Calendar que funciona como mi librería principal:

- **Gestión de deadlines** con alertas automáticas
- **Categorización por prioridad** usando etiquetas de color
- **Sincronización multiplataforma** para acceso desde cualquier dispositivo
- **Templates reutilizables** para tipos de tareas recurrentes

Commits recientes

Mejoras implementadas esta semana:

- feat(sleep-optimization): Implementé rutina de desconexión digital 1 hora antes de dormir

- `fix(procrastination)`: Técnica de bloques de 45 minutos con descansos de 15
- `update(workspace)`: Reorganización del espacio de estudio para mayor concentración

Bugs detectados:

- `issue(time-management)`: Sobreestimación constante del tiempo disponible
- `bug(notification-overload)`: Exceso de notificaciones causando dispersión

Retrospectiva semanal

Aspecto	Estado	Observaciones
Logros	✅	Mayor consistencia en horarios de estudio
Desafíos	⚠️	Dificultad para mantener equilibrio social-académico
Próximas mejoras	🎯	Implementar sistema de recompensas por objetivos cumplidos

Taller 2: "Mi stack tecnológico personal"

Propósito

Documentar y evaluar las herramientas y metodologías que he desarrollado como "dependencias" reutilizables en diferentes contextos de mi vida académica y personal.

Librería principal: `madrid-productivity-toolkit`

Funciones exportadas:

```
javascript
```

```
// Módulo de planificación
planificarSemana(objetivos[], recursos[])
// Distribuye tareas según disponibilidad y prioridades

evaluarProgreso(metricasActuales, metricasObjetivo)
// Calcula porcentaje de avance y sugiere ajustes

// Módulo de aprendizaje
crearMapaConceptual(tema, subtemas[])
// Estructura visual para materias complejas

aplicarFeynmanTechnique(concepto)
// Simplifica explicación hasta dominio completo

// Módulo de bienestar
gestionarEnergia(actividades[], intensidad[])
// Balancea actividades demandantes con recuperación
```

Casos de uso exitosos:

- **Matemáticas Discretas:** crearMapaConceptual() me ayudó a conectar grafos, lógica y conjuntos
- **Programación Web:** planificarSemana() dividió el proyecto final en sprints manejables
- **Preparación de exámenes:** aplicarFeynmanTechnique() mejoró mi comprensión profunda

Evaluación del toolkit

Criterio	Puntuación	Justificación
Modularidad	9/10	Cada función es independiente y combinable
Documentación	7/10	Clara para uso personal, necesita mejoras para otros
Mantenibilidad	8/10	Fácil de actualizar según nuevas necesidades
Escalabilidad	6/10	Funciona bien para carga actual, incierto para más complejidad

Taller 3: "API de interacción social: maximiliano.dev/api"

Propósito

Diseñar una interfaz conceptual que permita a otros entender mis patrones de disponibilidad, comunicación y colaboración, optimizando las interacciones interpersonales.

Documentación de endpoints

GET /disponibilidad

```
json

{
  "lunes_viernes": {
    "mañana": "ocupado (clases 08:00-12:00)",
    "tarde": "disponible (14:00-18:00)",
    "noche": "estudio personal (19:00-22:00)"
  },
  "fines_semana": "flexible, confirmar con anticipación",
  "modo_exámenes": "disponibilidad_limitada: true"
}
```

GET /preferencias_comunicacion

```
json

{
  "urgente": "llamada o WhatsApp",
  "academico": "email o Discord",
  "social": "redes sociales",
  "tiempo_respuesta": {
    "urgente": "< 30 minutos",
    "normal": "< 4 horas",
    "no_urgente": "< 24 horas"
  }
}
```

POST /colaboracion

```
json

{
  "tipo_proyecto": "required",
  "deadline": "required",
  "complejidad": "opcional",
  "mis_fortalezas": ["backend", "documentación", "testing"],
  "areas_aprendizaje": ["frontend avanzado", "diseño UX"]
}
```

GET /estado_actual

```
json
{
  "energia_nivel": 75,
  "carga_academica": "media",
  "disponibilidad_mental": "alta",
  "proyectos_activos": 3,
  "modo_concentracion": false
}
```

Métricas de la API

Métrica	Estado actual	Meta
Tiempo de respuesta	Variable	Consistente
Disponibilidad	85%	90%
Claridad en comunicación	Buena	Excelente
Gestión de expectativas	Mejorable	Óptima

Taller 4: "Evolución de mi sistema de estudio v1.0 → v3.0"

Propósito

Analizar la evolución de mi metodología de estudio como si fuera software en desarrollo, identificando mejoras incrementales y cambios de arquitectura importantes.

Versión 1.0 - "Caótico pero funcional"

Período: Inicio de carrera

Características:

- Estudio de última hora
- Apuntes dispersos en papel
- Sin planificación estructurada

Métricas:

- Efectividad: 60%
- Estrés: Alto
- Sostenibilidad: Baja

Versión 2.0 - "Organización básica"

Período: Segundo semestre

Mejoras implementadas:

- `feat(digital-notes)`: Migración a apuntes digitales
- `feat(calendar-integration)`: Uso de Google Calendar
- `fix(last-minute-cramming)`: Planificación con 1 semana de anticipación

Métricas:

- Efectividad: 75%
- Estrés: Medio
- Sostenibilidad: Media

Versión 3.0 - "Sistema maduro" (Actual)

Mejoras actuales:

- `feat(spaced-repetition)`: Implementación de repaso espaciado
- `feat(active-recall)`: Técnicas de recuperación activa
- `feat(pomodoro-adapted)`: Bloques de tiempo personalizados
- `refactor(note-system)`: Estructura Cornell + mapas mentales

Métricas actuales:

- Efectividad: 85%
- Estrés: Bajo-Medio
- Sostenibilidad: Alta

Roadmap v4.0

Próximas características:

- `feat(ai-assistance)`: Integración de herramientas IA para resúmenes
- `feat(peer-learning)`: Sistema de estudio colaborativo
- `optimization(review-cycles)`: Automatización de ciclos de repaso

Taller 5: "¿Cuándo copiar y cuándo crear desde cero?"

Propósito

Reflexionar sobre mis experiencias adoptando sistemas y metodologías externas versus desarrollar soluciones propias, evaluando cuándo cada enfoque es más efectivo.

Experimentos fallidos (Copy-paste sin adaptación)

Morning Routine de "influencers"

- **Qué copié:** Rutina 5:30 AM con meditación, ejercicio y journaling
- **Resultado:** Abandoné después de 3 días
- **Lección:** Mi ritmo circadiano natural es diferente, necesito personalización

Método Cornell para todo

- **Qué copié:** Sistema de notas de Cornell sin modificaciones
- **Resultado:** Ineficiente para materias prácticas como programación
- **Lección:** Una herramienta no sirve para todos los contextos

Pomodoro ortodoxo

- **Qué copié:** 25 min trabajo + 5 min descanso exactos
- **Resultado:** Interrumpía mi flow state
- **Lección:** Los tiempos deben adaptarse al tipo de tarea

Adaptaciones exitosas

Getting Things Done (GTD) modificado

- **Qué adapté:** El sistema de captura y organización
- **Modificación:** Simplifiqué las categorías y usé herramientas digitales
- **Resultado:** 40% menos estrés por tareas olvidadas

Técnica Feynman personalizada

- **Qué adapté:** Explicar conceptos complejos de forma simple
- **Modificación:** Agregué grabaciones de audio para autorrevisar
- **Resultado:** Comprensión más profunda en materias técnicas

Spaced Repetition System

- **Qué adapté:** Intervalos de repaso basados en curva de olvido
- **Modificación:** Ajusté intervalos según dificultad personal de cada tema

- **Resultado:** 60% mejor retención a largo plazo

Framework de decisión: ¿Reutilizar o crear?

Criterio	Reutilizar cuando...	Crear cuando...
Tiempo disponible	Es limitado	Tengo tiempo para experimentar
Complejidad	La solución existe y es estable	Mis necesidades son muy específicas
Experiencia previa	Otros han validado la solución	He fallado con soluciones similares
Contexto	Mi situación es similar a otros	Mi contexto es único

Reflexión final

Aprendizajes clave

1. **Modularidad personal:** Separar aspectos de mi vida en módulos independientes me permite optimizar cada área sin afectar las demás.
2. **Versionado continuo:** Como en software, mi crecimiento personal se beneficia de mejoras iterativas y reflexión sobre versiones anteriores.
3. **APIs sociales:** Ser explícito sobre mis formas de interacción mejora la calidad de mis relaciones y colaboraciones.
4. **Adaptación inteligente:** La clave no está en copiar soluciones completas, sino en entender principios y adaptarlos a mi contexto específico.

Próximos sprints personales

- **Sprint 1:** Implementar sistema de métricas personales para medir progreso objetivo
- **Sprint 2:** Desarrollar "documentación" clara de mis procesos para facilitar colaboración
- **Sprint 3:** Crear sistema de backup y recuperación para cuando mis rutinas fallen

Como desarrollador en formación, aplicar conceptos técnicos a la vida personal no solo mejora mi organización, sino que también refuerza mi comprensión de estos conceptos en el ámbito profesional.

El resultado es un sistema personal más robusto, escalable y mantenible.