



ESCUELA POLITÉCNICA NACIONAL  
CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE

**Integrantes:** Fernando Nagua, Ariel Mora

**Fecha:** 03/07/2025

**Curso:** GR1SW

## 1. Taller #1

**Dentro de este taller se desarrollarán las siguientes actividades:**

- Divide actividades diarias (estudiar, cocinar, descansar, entrenar) como \*submódulos\* de tu rutina.
- Evalúa cuáles se pueden mejorar o reutilizar en distintos contextos (por ejemplo, el mismo enfoque de estudio para distintas materias).
- Documenta mejoras aplicadas (como "commits") y haz una retrospectiva semanal para identificar avances o retrocesos

De esta manera, tomando en cuenta la rutina desempeñada durante el cronograma de clases, se obtuvo lo siguiente:

### 1. Repositorio: rutina-semanal

#### a. Submódulos (actividades diarias)

- i. *Estudio/*
  1. Contiene scripts y workflows de estudio para cada materia
- ii. *Cocina/*
  1. Contiene recetas, tiempos y preparación de las 5 comidas diarias dentro del cronograma de actividades.
- iii. *Descanso/*
  1. Contiene un plan de pausas, ocio (música, películas) y sueño.
- iv. *Entrenamiento/*
  1. Contiene rutinas de gimnasio divididas por grupo muscular.

### 2. Mejora y reutilización de los submódulos

#### a. Submódulos

- i. *Estudio/*
  1. **Reutilizable:** Técnica Pomodoro y checklist de tareas aplicable a cualquier materia.
  2. **Refactor:** Extraer plantilla de sesión de estudio (pomodoro-config.json) y todo.md común.
- ii. *Cocina/*
  1. **Reutilizable:** Plan de meal-prep semanal.
  2. **Refactor:** Crear un archivo menú-semanal.yml con menús parametrizables por día y hora.
- iii. *Descanso/*
  1. **Reutilizable:** Lista de reproducción y tiempos de desconexión.
  2. **Refactor:** Script break-reminder.sh para recordatorios cada 60 min de tarea.
- iv. *Entrenamiento/*
  1. **Reutilizable:** Plantilla de progreso con series, repeticiones y pesos.



## ESCUELA POLITÉCNICA NACIONAL CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE

2. **Refactor:** Macro generate-plan.py que crea la rutina diaria según el día de la semana.

### 3. Historial de “Commits” (Mejoras aplicadas)

```
commit alb2c3d # Fecha: lunes 30-06-2025
Author: Ariel David Mora Jarrin <arielabc389@gmail.com>
Message: feat(estudio): extraer plantilla Pomodoro y to-do
genéricos

commit d4e5f6g # Fecha: martes 01-07-2025
Author: Ariel David Mora Jarrin <arielabc389@gmail.com>
Message: chore(cocina): parametrizar menú en menu-semanal.yaml

commit h7i8j9k # Fecha: miércoles 02-07-2025
Author: Ariel David Mora Jarrin <arielabc389@gmail.com>
Message: feat(descanso): añadir script break-reminder.sh para
pausas cada hora

commit l0m1n2o # Fecha: jueves 03-07-2025
Author: Ariel David Mora Jarrin <arielabc389@gmail.com>
Message: refactor(entrenamiento): implementar generate-plan.py
para rutina semanal

commit p3q4r5s # Fecha: viernes 04-07-2025
Author: Ariel David Mora Jarrin <arielabc389@gmail.com>
Message: docs(README): documentar estructura de submódulos y
flujo de trabajo
```

### 4. Retrospectiva Semanal

#### a. Qué Funcionó

- Consistencia en estudio gracias al template Pomodoro: mejor enfoque en calidad de software y aplicaciones web.
- Meal-prep automatizado, redujo el tiempo de cocina diario
- Recordatorios de descanso, permitieron mantener la productividad y evitaron la fatiga.
- Plan de entrenamiento dinámico, permitió cumplir todas las sesiones de gimnasio.

#### b. Oportunidades de Mejora

- Automatizar más: Integrar alertas en el calendario o app de productividad para los bloques de estudio/, descanso/ y entrenamiento/.

### 2. Taller #2

Dentro de este taller se desarrollarán las siguientes actividades:

- Cada estudiante identifica una habilidad que domina (hacer resúmenes efectivos, cocinar rápido, resolver problemas de mates, etc.)
- Construye una “librería” con funciones reutilizables, documentadas en Notion o Google Docs, que otras personas podrían usar.



## ESCUELA POLITÉCNICA NACIONAL CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE

- Evalúa la claridad, reusabilidad, y facilidad de integración con otras “rutinas”.

En el siguiente link se encuentra el desarrollo del taller #2: [https://www.notion.so/Taller-2-22733ab0eeb38046be84e3f0814058ea?source=copy\\_link](https://www.notion.so/Taller-2-22733ab0eeb38046be84e3f0814058ea?source=copy_link)

### 3. Taller #5

**Dentro de este taller se desarrollarán las siguientes actividades:**

- Reflexiona sobre intentos de “copiar-pegar” soluciones de otros en tu vida (rutinas de estudio de YouTubers, apps de productividad, etc.).
- Evalúa cuándo la personalización fue necesaria y cuándo la reutilización fue efectiva.
- Cierra el taller con una discusión sobre \*cuándo y cómo adaptar vs. reaprovechar\* componentes.

#### 1. Reflexión sobre “copiar-pegar” soluciones ajenas

En mi caso he probado múltiples rutinas de estudio de YouTubers populares. De esta manera, adoptaba sus horarios de Pomodoro de 50/10, sus listas diarias en Trello y sus “técnicas de revisión rápida” antes de los exámenes. Sin embargo, al principio noté que esas rutinas, aunque bien diseñadas, no encajaban con mi realidad. Esto se debe a que, mis clases empiezan por la tarde, mis descansos necesitaban más flexibilidad y me distraía demasiado con notificaciones. También intenté apps de productividad “prefabricadas” (como Notion templates de trackers de hábitos) sin personalizar las columnas: acababa acumulando datos inútiles y desmotivándome.

#### 2. Evaluación de personalización vs reutilización

##### I. Cuando la personalización fue necesaria

- a. Mis horarios de clases (14–20 h) y mis sesiones de gimnasio (pecho/bíceps, espalda/tríceps, hombros) exigían bloques distintos a los genéricos de 25 min/5 min. Tuve que ajustar la duración de mis Pomodoro a 45/15 min para adaptarlos a el desgaste de mis materias y mi nivel de concentración.
- b. Mi menú de cinco comidas diarias necesitaba un plan de meal-prep que considerara refrigeración y variedad, no el meal-prep de 3 comidas estándar que se suele promocionar.

##### II. Cuando la reutilización fue efectiva

- a. Las plantillas de checklist en Notion (tareas, asignaturas, seguimiento de progreso) me sirvieron casi sin cambios.
- b. El script simple de “break-reminder” cada 60 min funcionó perfectamente para mis sesiones largas de programación.

#### 3. Discusión: ¿Cuándo y cómo adaptar vs reaprovechar componentes?

##### a. Reaprovechar

- a. Esto se debe utilizar cuando la solución ajena cubra una funcionalidad genérica, la cual no tenga ataduras al contexto: por ejemplo, un checklist de tareas, un cronómetro de Pomodoro base o un tracker de hábitos.

##### b. Adaptar



## ESCUELA POLITÉCNICA NACIONAL CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE

- a. Cuando el entorno aporte restricciones distintas (horarios atípicos, requerimientos dietéticos, fases de proyecto concretas) se debe tomar el componente ajeno como punto de partida y modificar parámetros clave: duración de bloques, categorías de tu tracker, esquema de datos, fórmulas de cálculo. Ahora bien, el proceso de adaptación suele implicar:
  - i. **Identificar** qué parte del componente estándar no encaja.
  - ii. **Extraer** la lógica reutilizable
  - iii. **Refactorizar** para inyectar otros valores
  - iv. **Probar** en la rutina real y ajustar según retroalimentación.