



Trabajo Domiciliario Obligatorio

Programación 3 Colonia Noct. 2025

Asignatura: Programación 3

Profesor responsable: Matías Verges

Fecha de asignación: 28 / 04 / 2025

Fecha límite de entrega: 02 / 07 / 2025 (23:59 h)

1. Objetivo

El propósito de este trabajo domiciliario es **incorporar código funcional y de calidad** al repositorio público **Naitv1**, proyecto central de la asignatura. Cada grupo deberá planificar, implementar, documentar y proponer *pull requests* (PR) para las funcionalidades que se describen a través de *user stories*.

2. Alcance funcional

- Existen **5 Epics** disponibles en el repositorio.
- Cada grupo elegirá exactamente **un Epic** y trabajará **exclusivamente** sobre él durante todo el período del trabajo.
- Estructura de cada Epic:
 - 1 *user story* **A** · 3 pts (obligatoriamente la primera en abordarse).
 - 2 *user stories* **B** · 2 pts c/u (4 pts en total).
 - 3 *user stories* **C** · 1 pt c/u (3 pts en total).
- Completar las seis *user stories* otorga **10 story points = 100 %** de la calificación del trabajo.

3. Cronograma de expectativas

Para ayudarles a distribuir el esfuerzo, se proponen **tres hitos intermedios** con el número de puntos que se espera hayan sido *mergeados* en el repositorio principal. Los hitos son orientativos; podrán ajustarse con la debida justificación y previa comunicación.

Hito	Fecha objetivo	Story points esperados	Historias sugeridas
1	16 / 05 / 2025	3 pts (30 %)	Historia A (3 pts) totalmente integrada
2	08 / 06 / 2025	7 pts (70 %)	Historias B1 y B2 (2 pts + 2 pts)
3 (final)	02 / 07 / 2025	10 pts (100 %)	Historias C1 – C3 (1 pt c/u)

Importante: Los puntos **cuentan** únicamente cuando el PR ha sido revisado, aprobado y *mergeado* en la rama principal por el profesor.

4. Flujo de trabajo y entrega

1. Fork del repositorio Naitv1.

Trabajar siempre sobre su fork y mantenerlo sincronizado con el *upstream* haciendo PULL tan seguido como sea posible. Especialmente después de cada clase.

2. Desarrollo centrado en *user stories*.

Implementar, commitear y documentar cada user story en una rama temática.

3. Pull Request.

- Usen la plantilla de PR prevista.
- Referenciar la user story correspondiente en el título y en la descripción.

4. Revisión & feedback.

- El profesor revisará el PR.
- Deberán responder a los comentarios y subir correcciones.
- Una vez aprobado, se hará *merge* → puntos obtenidos.

5. Evaluación

Aspecto	Peso
Cobertura de requisitos de la <i>user story</i>	40 %
Calidad de código (legibilidad, patrones, SOLID)	40 %
Documentación (código y PR)	10 %
Gestión del flujo de trabajo (commits, ramas, respuestas)	10 %

Se requiere **un mínimo de 5 puntos mergeados** para aprobar el trabajo.

6. Buenas prácticas y recomendaciones

- Iterar pronto y con frecuencia; pequeños PR facilitan la revisión.
- Revisen el *checklist* de la plantilla antes de marcar PR como «Listo para revisar».
- Cumplan con las convenciones de estilo acordadas en clase.

7. Preguntas y soporte

Documentos de referencia en el siguiente repositorio:

<https://github.com/matismasters/prog3/tree/master/obligatorio>

Repositorio principal de la aplicación:

<https://github.com/matismasters/Naitv1>