

Proyecto.

Curso: Introducción a la Programación.

Docente: Alejandro Alfaro Quesada. (aalfaroq@utn.ac.cr)

Fecha de entrega: miércoles 30 de julio del 2025.

Hora de entrega: 11:59 pm.

Indicaciones Generales.

- La prueba se desarrolla bajo la modalidad de proyecto, con tiempo de inicio y fin predeterminado.
- Toda prueba evaluada es considerada como un único proyecto, es decir, aunque la evaluación sea dividida por secciones, la completitud de uno o más rubros puede afectar la funcionalidad del sistema, y esto, podría influir en la nota final.
- La prueba puede ser desarrollada en un máximo de dos personas. En cuanto a los equipos de trabajo hay dos reglas que se deben contemplar, las cuales son:
 - Para los casos que son parejas, si por alguna razón no puedan continuar trabajando juntos, ambos deben continuar con el código fuente que tienen hasta ese momento; es decir, no se permite que se unan a otro compañero y/o utilizar el código fuente de otro compañero.
 - Aunque el trabajo sea desarrollado entre dos personas, en términos de la evaluación y asignación de la nota, ambos estudiantes serán evaluados de manera individual; lo anterior, para el caso en que el docente compruebe que uno de los estudiantes tuvo poca o nula participación en el desarrollo del proyecto, de acuerdo con las preguntas que se realizarán en el momento de la defensa del proyecto.
- Si algún estudiante es sorprendido en actos fraudulentos, ya sea que los provoque o los consienta, su prueba y la de los demás implicados, será automáticamente anulada.
- El docente ejecuta el código fuente que está en el proyecto enviado para comprobar que realiza lo solicitado. Por lo que, el docente NO modificará ninguna línea de código, por lo tanto, debe asegurarse que el código fuente funciona correctamente.
- El proyecto con el código solo se recibe y se valida desde el Campus Virtual de la UTN. Por cualquier otro medio que se envíe al docente no será revisado.
- Para la explicación del trabajo el estudiante debe estar de manera presencial en la clase, de lo contrario, se rebajan 15 puntos de la nota que obtenga en el momento de la evaluación. Para el caso del grupo 2, deben estar presencialmente en el Congreso Tecnológico de ISW.

Propuesta Libre

Nota: Para resolver este ejercicio se prohíbe el uso de librerías como numpy, pandas o similares de alto nivel; también el uso de motores de simulación avanzada como SimPy, Arcade, Panda3D o similares (si considera alguna y tiene dudas, consulte al docente), debe ser utilizando únicamente las estructuras de datos vistas en clase; además, se prohíbe el uso de archivos de cualquier formato, el uso de las librerías o archivos mencionados invalida por completo la prueba.

A su vez, cada estudiante/grupo debe trabajar sobre el caso asignado. Cualquier solución que no esté basada en el contexto entregado se considerará inválida con nota de cero.

También, cada estudiante/grupo debe asistir de manera presencial cuando el docente se lo indique para explicar cómo abordará la solución del proyecto previo a su desarrollo.

Objetivo

Desarrollar un sistema interactivo utilizando Pygame que simule la resolución de un problema real, transformado en una experiencia de tipo juego o simulador. El sistema debe ser visual, tener reglas propias claramente definidas, incluir eventos dinámicos y simular comportamientos autónomos de usuarios con roles definidos.

Secciones del sistema.

Cada estudiante o grupo debe construir su sistema desde cero, incorporando lo siguiente:

1. **Problema:** Dada la situación que se le asignó, defina de manera clara y breve el problema que se desea resolver.
2. **Meta:** Explique qué debe lograr el usuario en el sistema para considerar que ha tenido éxito. Como guía, responda la siguiente pregunta ¿Qué debe alcanzar el sistema para considerarse que la simulación fue exitosa?
3. **Usuarios y roles:** Cree al menos tres tipos de usuarios con comportamientos distintos (ej. clientes, trabajadores, vigilantes). Cada tipo de usuario debe:
 - a. Tener una o más metas personales.
 - b. Tener un comportamiento definido (¿cómo decide moverse? ¿qué evita? ¿cómo reacciona ante eventos?).
4. **Reglas del juego:** Describa las reglas principales que determinan el funcionamiento del sistema, por ejemplo: ¿Cómo se mueven los usuarios?,

¿Qué acciones están permitidas o restringidas?, ¿Qué condiciones cambian su comportamiento?, ¿Cómo interactúan los usuarios entre sí?

5. **Eventos dinámicos:** Incluya al menos dos eventos que pueda modificar el entorno durante la simulación, los cuales deben tener un impacto real en la lógica y decisiones del sistema. Un evento puede ser:
 - a. **Temporal:** ocurre por tiempo (ej. cada 20 segundos).
 - b. **Condicional:** ocurre si pasa algo (ej. mucha gente en una zona).
 - c. **Aleatorio:** se activa al azar durante la simulación.
 - d. **Ejemplos:**
 - i. **Cafetería:** Se descompone una caja registradora entonces aumenta el tiempo de espera.
 - ii. **Parqueo:** Hay un choque o accidente entonces se bloquean ciertos espacios en el parqueo.
 - iii. **Seguridad:** Si se detecta una persona no autorizada en una zona restringida entonces se llama a seguridad.
 - iv. **Gimnasio:** Si se daña una máquina entonces se redistribuyen los horarios para los usuarios.
6. **Condiciones de victoria o puntuación:** Definir claramente cómo se mide el éxito o fallo en el juego; por ejemplo, por puntos ganados, por cantidad de personas atendidas, entre otros; además desarrollar qué sucede si gana o pierde. Debe definirse claramente qué se considera “éxito”, “fracaso” o “resultado medio”.
7. **Requisitos técnicos:** Debe haber un mínimo de tres ventanas del juego propiamente, las cuales son menú principal, simulación del juego, resultados o estadísticas; pantallas informativas se pueden agregar, pero no se consideran dentro de esas tres.

Documentación.

Debe crear y subir junto con la solución un documento (Word) en donde explique lo siguiente:

- **Lógica del algoritmo:** debe explicar claramente cómo funciona el algoritmo que desarrolló para resolver los problemas de las secciones que se listan en el cuadro siguiente; en cada explicación coloque una imagen del código donde realiza el proceso, así como datos que ud considere relevantes para explicar lo que desarrolló.
- Para el momento de la defensa del proyecto debe presentar ejemplos para probar la funcionalidad del sistema de acuerdo con lo solicitado.



- Agregue la siguiente tabla completa:

Sección	¿La sección está completamente terminada y funcional? Indique sí o no.	Si colocó que no, indique una breve descripción del por qué no está completa la sección.
Problema.		
Meta.		
Usuarios y roles.		
Reglas del juego.		
Eventos dinámicos.		
Condiciones de victoria o puntuación.		
Indicaciones generales.		

Exposición.

Para la exposición de este proyecto debe asistir el día y la hora que el docente le indique.

Este porcentaje se obtiene de acuerdo con la participación y explicación de aspectos técnicos de cada estudiante durante la defensa de su proyecto.

Observaciones.

Días previos a la entrega y exposición, el docente indicará los detalles de la exposición y el formato de la presentación.