

### Deuda Técnica

### Objetivo de Aprendizaje

El objetivo de esta actividad de aprendizaje es presentar a los estudiantes las prácticas LEAN y Agile tomando el concepto muy simple de deuda técnica en el contexto del desarrollo de software, hacer que experimenten su impacto y tratar de tomar medidas para contrarrestarlo.

### Objetivos de Aprendizaje y Resultado

Después de jugar este escenario, los estudiantes podrán:


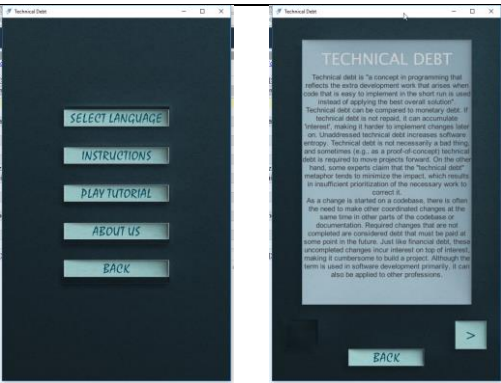

- Comprender el concepto de "deuda técnica".
- Aplicar los mejores procedimientos y técnicas para equilibrar la "deuda técnica" y las versiones de software.

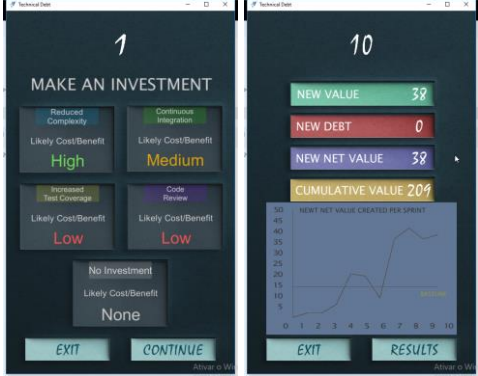
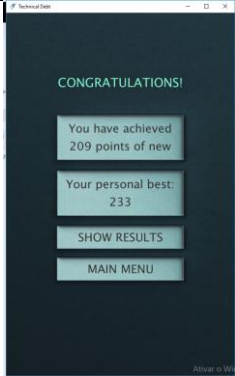
### Cómo Usar LEAP

En este juego, el jugador asume el rol de un gerente de equipo de desarrollo de software trabajando en un proyecto que durará 10 sprints. Su trabajo es crear la mayor cantidad de nuevo valor de software para el final del proyecto. Para ello, el jugador tiene que equilibrar el proceso de desarrollo de software que genera el nuevo valor (NV), pero aumenta la deuda técnica (TD) y la reducción real de TD a través de varias medidas de inversión disponibles.

Durante cada turno, el equipo de desarrollo de software tiene una capacidad finita para crear un nuevo valor de software y tratar con deb técnico. Al comienzo del juego, el jugador tiene un cierto número de puntos disponibles para crear un nuevo valor (NV), y un cierto número de puntos para la deuda técnica (TD).

Cada turno (que representa un sprint), el jugador tira un dado para crear un nuevo valor y toma el total de todos los puntos acumulados. Luego tira los dados técnicos de la deuda y suma ese número. El nuevo valor neto (NNV) creado en cada turno es el total de NV menos el total de TD.

<p>La pantalla inicial permite iniciar un nuevo juego o ver las opciones de configuración.</p>	
<p>El menú de opciones permite configurar el idioma del juego, ver algunas instrucciones del juego y jugar un juego tutorial. La primera vez que juegues debes consultar las instrucciones y el tutorial. Mostrar a los alumnos cómo cambiar el idioma del juego.</p>	
<p>Jugar el tutorial es importante porque permite ver cuál es el efecto de aplicar una inversión.</p> <p><b>Complejidad reducida</b>  Efecto: eliminar 7 puntos en promedio de TD y agregarlos en NV.  Costo: 7 puntos en promedio de NV a TF para 3 sprints.</p> <p><b>Integración continua</b>  Efecto: eliminar 3.5 puntos en promedio de TD y agregarlos en NV.  Costo: 10.5 puntos en promedio de NV a TD para 2 sprints.</p> <p><b>Mayor cobertura de prueba</b></p>	

<p>Efecto: elimina 3 puntos del TD en cada sprint.  Costo: 3.5 puntos en promedio de NV a TD para 3 sprints.</p> <p><b>Revisión de código</b></p> <p>Efecto: Reduce el TD en una cantidad aleatoria.  Costo: 3.5 puntos en promedio desde NV a TD para 2 sprints.</p> <p><b>Sin inversión</b></p> <p>Efecto: ninguno  Costo: ninguno</p>	
<p>Comience el juego y explique a los alumnos las cinco opciones, solo para recapitular los conceptos aprendidos en el tutorial. Después de eso, deja que los jugadores elijan las opciones y sigan el ciclo de 10 sprints.</p>	
<p>Al final, verifica los resultados de todos los estudiantes. Haga que los estudiantes con el puntaje más alto expliquen su método a los otros estudiantes.</p>	

## Colaboración de Clase

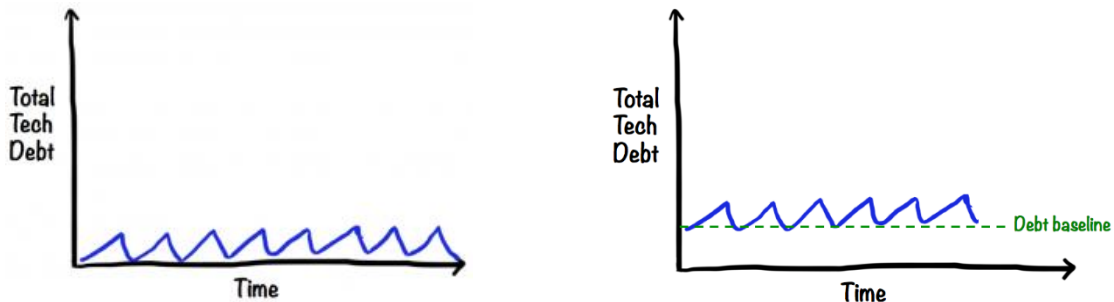
Preguntas para estimular la discusión en el aula:

- ¿Tiene sentido no desarrollar la mejor solución de inmediato?
- ¿La necesidad de refactorización implica que el código existente es malo e ineficiente?

- ¿Tiene sentido el concepto de "deuda técnica" en otras tecnologías de desarrollo de software como en los modelos de cascada?
- ¿Cuál sería la curva ideal para la deuda técnica?

## Evaluación

Haga que los estudiantes jueguen y evalúen la curva de deuda técnica que lograron, en comparación con las curvas "ideal" (imagen izquierda) o "ideal real" (imagen derecha) propuestas por Henrik Kniberg. Haga que los estudiantes expliquen la relación de la curva que lograron con estos dos.



## Materiales auxiliares

- Techopedia, deuda técnica, disponible en: <https://www.techopedia.com/definition/27913/technical-debt>
- Wikipedia, deuda técnica, disponible en: [https://en.wikipedia.org/wiki/Technical\\_debt](https://en.wikipedia.org/wiki/Technical_debt)
- Ward Cunningham, Debt Metaphor, disponible en: <https://www.youtube.com/watch?v=pqeJFYwnkjE>
- Steve McConnell, 10x Mejores prácticas de desarrollo de software: Deuda técnica, disponible en: [http://www.construx.com/10x\\_Software\\_Development/Technical\\_Debt/](http://www.construx.com/10x_Software_Development/Technical_Debt/)
- Henrik Kniberg, Deuda Técnica Buena y Mala, disponible en: <http://blog.crisp.se/2013/10/11/henrikkniberg/good-and-bad-technical-debt>

