

CHECKLIST

para corregir el proyecto Home Banking

Para que un proyecto apruebe u obtenga categoría hacker, tiene que cumplir con todas las condiciones del checklist correspondiente.

Condiciones para entregar _

- **Archivos mínimos entregados:** En la carpeta entregada se incluyen los archivos index.html, style.css y homebanking.js, disponibles en los recursos descargables. Puede incluir archivos con funcionalidades extra.
- **Enlace de JS a HTML:** Se utiliza la etiqueta HTML correspondiente para conectar el archivo HTML con el JS.
- **Funciones desarrolladas:** Revisar que el archivo homebanking.js tenga las funcionalidades principales desarrolladas (que no estén vacías)

Condiciones para aprobar _

- **Declaración de variables:** Se declaran correctamente todas las variables que indican las guías del proyecto
- **Funcionalidad de cambio de límite de extracción:** Permite que el usuario cambie su límite de extracción.
- **Funcionalidad de depósito de dinero:** La funcionalidad de depósito de dinero suma correctamente dinero en la cuenta. Como se indica en la guía 2, es necesario que la responsabilidad de sumar dinero se delegue en otra función.
- **Funcionalidad de extracción de dinero:** Se resta correctamente dinero de la cuenta aplicando todas las validaciones indicadas en la consigna

(montos múltiplos de 100, que no superen el límite de extracción, si hay dinero suficiente en la cuenta).

- **Funcionalidad de pago de servicios:** Realiza la funcionalidad de pago de servicios correctamente utilizando la sentencia SWITCH y aplicando la validación que indica la guía 3 (si hay dinero suficiente en la cuenta).
- **Funcionalidad de transferencia de dinero:** La funcionalidad de transferencia de dinero resta correctamente dinero de la cuenta aplicando las validaciones que indica la guía 3 (permitir transferencias solo a cuentas amigas, no permitir transferencia de un monto de dinero mayor al de la cuenta).
- **Funcionalidad de inicio de sesión:** La funcionalidad de inicio de sesión valida si el usuario ingresó el código de seguridad correcto. Si lo ingresa mal, retiene el dinero de la cuenta, dejando el saldo de la cuenta en \$0.
- **Actualizar variables en el HTML:** Cada vez que el saldo de la cuenta o el límite de extracción sufren modificaciones, las mismas se ven reflejadas en el HTML (DOM).
- **Alertas:** Se muestran las alertas en las funciones correspondiente según lo indican las guías.

Condiciones para Categoría Hacker _

- **Código modularizado y reutilizable:** Se delegan responsabilidades en otras funciones para lograr un código limpio y ordenado. Se evita la repetición de código creando funciones que realicen las acciones que se repiten constantemente.
- **Nueva funcionalidad o validación extra:** Además de las funcionalidades o validaciones indicadas en las guías, se agregan una o varias extra.

- **Prolijidad en el código:** La indentación y tabulación del código facilita la lectura

Link al checklist en Markdown

https://github.com/acamica/evaluacion-tecnica/blob/master/checklist_home_banking.md

Podés copiar el contenido que se encuentra en este link y agregarlo a tu devolución.