

ExploraApp

El proyecto plantea a los estudiantes el desafío de programar una aplicación en App Inventor para jugar en una tablet o celular aprovechando el uso de sensores. El contenido del juego, definido por el docente de aula, requerirá a los estudiantes identificar elementos dispuestos sobre una imagen de fondo que cumplan con una serie de consignas aleatorias.

Las posibilidades de articulación con contenidos curriculares, que la estructura de la app permite, son muy variadas. Por ejemplo, sobre un mapa de Uruguay, se puede orientar a una exploradora para que llegue a un departamento, ciudad, río o cualquier sitio determinado como objetivo; una línea de tiempo de fondo puede ser el escenario para asociar sucesos históricos; la imagen de una célula el escenario para repasar la identificación de sus componentes; el esquema de un proceso productivo, de una ruta comercial el fondo para reconocer diferentes hitos representados. La dinámica del juego contempla la incorporación de un distractor que persigue al explorador para evitar que llegue a su objetivo.

Además de conocer un nuevo entorno de programación, durante todo el proceso de construcción de la app, los estudiantes participan de forma activa para establecer el plan de desarrollo, la interfaz, los procedimientos para ganar o perder. Se encontrarán con la necesidad de identificar subproblemas, encontrar y reutilizar estrategias de solución y experimentar un método de desarrollo incremental. En el proyecto se incorporan, además, las herramientas de programación Procedimientos y Listas.

ExploraApp se presenta como un proyecto:

- Desafiante en relación a la programación involucrada en el desarrollo de una aplicación.
- Flexible en cuanto a los conceptos que se pueden relacionar.
- Creativo en la medida en que sitúa a los alumnos como diseñadores y creadores de su propia app.

Duración: 9 a 10 semanas.

Materiales: App Inventor, computadoras y tablet o celular.

Ficha Curricular ↓	2
Objetivos de aprendizaje de 3° año de Pensamiento Computacional	2
Referencias al Marco Curricular Nacional (CREA Arte interactivo. Nivel 3)	2
Síntesis de la propuesta	4
Acuerdos iniciales de coordinación	4
Bitácora y trabajo en App Inventor	7
ETAPA 1 ↓	8
ETAPA 2 ↓	11
ETAPA 3 ↓	15
ETAPA 4 ↓	19
ETAPA 5 ↓	24
ETAPA 6 ↓	28
ETAPA 7 ↓	32
ETAPA 8 - Opcional ↓	38
ETAPA 9 ↓	44
 ANEXO 1	48
 ANEXO 2	48
 ANEXO 3	48
 ANEXO 4	49
 ANEXO 5	52
 ANEXO 6	53

Objetivos de aprendizaje de 3° año de Pensamiento Computacional

Comunicación y Colaboración

- Colaborar y construir conjuntamente un proyecto grupal.

Resolución de problemas computacionales

- Incorporar la estrategia de división de un problema en subproblemas.
- Seleccionar las herramientas de programación adecuadas para el subproblema a resolver.
- Recuperar estrategias de solución y herramientas de programación construidas en experiencias anteriores para adaptarlas a nuevos problemas.

Análisis de datos, desarrollo y uso de abstracciones

- Analizar los datos importantes y la información relevante de un problema dado.
- Reconocer las generalidades de los comportamientos de sus programas, en términos abstractos, para que puedan ser adaptados, combinados y/o reutilizados.
- Explorar distintas formas de representación y sistematización de los datos para modelar un problema.

Creación de algoritmos, programas y dispositivos

- Utilizar las herramientas de programación adecuadas para implementar sus programas.
- Utilizar el método de desarrollo incremental.
- Comprender la posibilidad de reutilizar partes de programas ya creados en la construcción de programas nuevos.

Evaluación

- Identificar y corregir, con ayuda del docente, errores mediante un proceso sistemático.
- Comparar las diferencias entre versiones y resultados de dispositivos y/o programas.

Contenidos PC:

- Listas ● Procedimientos ● Método de desarrollo incremental ● Reutilización de código

Referencias al Marco Curricular Nacional (CREA Arte interactivo. Nivel 3)

Espacio Técnico - Tecnológico. Unidad curricular Ciencias de la Computación y Tecnología Educativa. Tramo 4

Posibles vinculaciones a otros espacios y unidades curriculares.
A definir por maestro/a de aula

Competencias generales

Comunicación, Pensamiento Creativo, Pensamiento Computacional, Metacognitiva, Relación con los otros, Iniciativa y orientación a la acción.

Competencias específicas

CE1 Selecciona y utiliza medios y formatos digitales, generando producciones, adecuándose a diferentes contextos e interlocutores, para presentar información y comunicarse.
CE4 Indaga e identifica técnicas, herramientas y aplicaciones que la tecnología proporciona para la resolución de problemas así como situaciones que pueden abordarse como problemas computacionales.
CE5 Recupera soluciones propias o ajenas y construye modelos, para resolver problemas simples, en grupo y de forma mediada, enriqueciendo sus construcciones y las de otros.
CE6 Utiliza la programación y dispositivos tecnológicos en la implementación colectiva de soluciones para la resolución de problemas. Contribuye al desarrollo de las competencias generales del MCN: Pensamiento creativo, Pensamiento computacional, Metacognitiva, Iniciativa y orientación a la acción.

Contenidos específicos

Pensamiento computacional - Estrategias para la resolución de problemas o creación de juegos y otros recursos: patrones, reutilización, descomposición, iteración, ensayo y error, método incremental, entre otros.
Programación en lenguajes de bloques: aspectos gráficos, bloques de control, variables, sensores, eventos, operadores.

Criterios de logro

El proceso de construcción de la App presenta una oportunidad para fortalecer el aprendizaje de diversos contenidos del 2do ciclo. Se parte por considerar el contenido a fortalecer y se planifica una propuesta de enseñanza en diálogo con el proceso de aprendizaje de los estudiantes. La temática elegida debe ser posible de ser representada con una imagen y con elementos sobre la misma que se deban reconocer o ubicar.

Se identifican algunos contenidos del 2do ciclo, que podrían articularse:

- **Biología**
Estructura de la célula animal: Cómo imagen de base se utiliza una célula animal y el juego consiste en llegar a el nombre o función de los orgánulos que la aplicación indica.
- **Historia**
Línea de tiempo: Cómo imagen de base se utiliza una línea de tiempo armada con los años de la época o hecho en estudio y el juego consiste en llegar a la fecha o hecho que la aplicación indica.
- **Geografía**
La ubicación de lugares: Cómo base se utiliza un mapa de Uruguay o del tema en estudio y el juego consiste en llegar al departamento/parque nacional/monumento histórico que la aplicación indica.

Materiales complementarios sugeridos

- [Uruguay Educa - Recursos educativos](#)

Referencias al Marco Curricular Nacional (CREA Arte interactivo. Nivel 3)

Reconoce avances tecnológicos en su entorno.
Identifica algunos impactos de la tecnología en su vida cotidiana. Reconoce patrones o características comunes entre elementos y situaciones en la resolución de problemas.
Divide un problema en subproblemas, analizando datos importantes e información relevante.
Plantea y representa gráficamente diferentes soluciones, que pueden resolver un problema dado.
Recupera soluciones construidas en experiencias anteriores y las adapta a nuevos problemas.
Aplica soluciones conocidas en nuevos contextos para elaborar procedimientos más complejos.
Resuelve problemas computacionales utilizando algunas herramientas básicas de programación (condicionales, iteraciones, variables, etc.). Reconoce que un dispositivo automatizado puede interactuar con el entorno.
Planifica, crea y modifica, con ayuda del docente, un programa o solución tecnológica.

Síntesis de la propuesta





 Las etapas organizan unidades conceptuales. Las actividades previstas pueden requerir más de una VC de 45 minutos.

Acuerdos iniciales de coordinación

El diálogo permanente de **docentes remotos (DR)** y **docentes de aula (DA)** es fundamental para llevar adelante esta propuesta.

Decisiones del DA → comunicar a DR :

- Definir el proyecto de aula alrededor de un contenido curricular.
- La dinámica de trabajo para identificar junto a los estudiantes el enunciado que guiará el proyecto y los elementos a identificar.

Decisiones DR → comunicar a DA:

- Explicitar al DA semanalmente los objetivos de cada VC y establecer acuerdos en torno a la dinámica de las clases remotas, la organización espacial necesaria y la participación del DA.

Información que necesita tener el DR:

- Si los estudiantes abordaron contenidos similares en otras instancias o años anteriores.

Rol del DA durante las VC

- En las actividades de **inicio** organiza el intercambio para que los estudiantes relaten al DR lo realizado en el aula.
- En las actividades de **desarrollo**, será importante intervenir para vincular el trabajo a lo realizado en el aula y al proyecto global en el que se inscribe esta propuesta.
- En las actividades de **cierre y reflexión**, su participación es fundamental para recuperar momentos que haya observado durante el desarrollo de las actividades y apelar a experiencias previas de los estudiantes que aporten a las reflexiones propuestas por el DR.
- Durante todo el proyecto serán valiosas las acciones del DA que favorezcan el **vínculo** de los estudiantes con el proyecto y el DR.
- Durante los **intercambios**, facilitar la circulación de la palabra, permitirá que todos los estudiantes tengan oportunidad para expresarse.

Rol del DR durante el proyecto

- Anticipar al DA el modo y el contenido planificado para cada VC.
- Indagar los contenidos programáticos que el DA elige para acompañar la propuesta pedagógica y resignificarlos durante la VC.
- Llevar adelante las clases por VC en conjunto con el DA.
- Gestionar el curso en Crea de la propuesta, realizar los ajustes necesarios y las devoluciones a los estudiantes que correspondan.

Aplicación de ejemplo

Se dispone de una aplicación realizada exclusivamente para consulta de docentes: "Explora Uruguay". Se brindan dos proyectos, en cada uno se utiliza un sensor distintos para desplazarse.

- [Proyecto completo con sensor de orientación](#) y [Proyecto completo con sensor acelerómetro](#)

En este ejemplo, se decidió trabajar con la imagen de base de un mapa político de Uruguay, y el objetivo del juego es ir encontrando los departamentos que indica el enunciado. Para facilitar la prueba de la aplicación se deja en el mapa los nombres de los departamentos. El proyecto cuenta con siete departamentos en total. [Acceso a la carpeta con los archivos de avance de cada etapa.](#)

Bitácora y trabajo en App Inventor

El trabajo de los estudiantes se organizará en una **bitácora por cada grupo**, donde se registra la evolución del Proyecto. Es importante que la dupla pedagógica defina la forma de trabajo en la bitácora:

- Se sugiere compartir una **carpeta en Google Drive**, con la misma cuenta que utilizan para App Inventor, donde se ubique la presentación. Los enlaces a esas carpetas se comparten y enlazan en el aula de Crea.
- Se deja a disposición una  Bitácora Modelo - ExploraApp . Los docentes pueden acordar **otros formatos** para la bitácora respetando la estructura del modelo.

App Inventor acceso e idioma

Se sugiere que los estudiantes tengan una cuenta de acceso a Gmail para simplificar el registro y acceso. Es importante que la configuración se encuentre en Español para facilitar la incorporación del nuevo entorno.



Curso en plataforma Crea ↓

Se destinará una carpeta en Crea para este proyecto dentro del Curso de PC, que contiene una estructura similar a la de esta guía. Este espacio virtual ofrece herramientas de trabajo que servirán al DR a llevar adelante distintos momentos en la VC.



Actividades Interactivas

Las actividades interactivas están pensadas para ser realizadas en distintos momentos en cada etapa. En alguna oportunidad pueden ser una instancia de aprendizaje de los contenidos, en otras pueden formar parte del cierre del desarrollo de la clase. Lo importante es recuperar la resolución de las mismas para realizar una puesta en común.

Foro de evidencias

Los avances del proyecto se comparten en los foros de evidencias. Se puede elegir uno o varios aportes, así como proyectos de programación para analizarlos entre todos durante la VC. Lo importante es socializar la programación y enriquecer el intercambio de ideas.

Reflexión y registro de cierre

A lo largo de toda esta propuesta se propone plasmar los intercambios del cierre en un **registro común** para toda la clase que se va enriqueciendo en cada etapa. Cada pareja de docentes considerará la herramienta más adecuada que permita compartir un enlace con los estudiantes en la plataforma. Puede utilizarse un documento compartido para tomar el registro, una página creada en Crea o incluso mapas conceptuales realizados a partir de los intercambios grupales.

Las dinámicas para la escritura en este archivo podrán ir variando entre una etapa y otra. Algunas veces se puede recurrir a la **escritura por parte de los docentes**, otras veces se puede **recopilar respuestas de un foro**, compilar imágenes de **capturas de pantalla** o solicitar **escrituras parciales** a subgrupos.

ETAPA 1 ↓**El nuevo entorno de programación**

En la VC, se da comienzo a la propuesta de trabajo. Los estudiantes exploran el nuevo entorno de programación, ejecutan en el emulador una aplicación y reconocen en la programación bloques utilizados, entradas y salidas.

Objetivos

Se espera que los estudiantes sean capaces de:

- Familiarizarse con el entorno App Inventor para elaborar programas y cargarlos en el emulador del dispositivo.
- Identificar componentes utilizados en una aplicación y propiedades asociadas.

Coordinación dupla pedagógica**Decisiones del DA**

- Antes de la VC, el DA, debe verificar que los estudiantes cuenten con acceso a una cuenta de Gmail para poder acceder al entorno de programación.
- Conformar grupos de 2 a 4 integrantes para trabajar a lo largo de toda la propuesta.

Decisiones conjuntas entre DA y DR:

- Repasar el Rol del DA durante la VC a partir de los acuerdos iniciales.
- Acordar el modo en el cual se plasmará la bitácora del proyecto por grupo.
- La participación del DA para gestionar las primeras experiencias de los estudiantes con el nuevo entorno será muy valiosa (ingresar al entorno, subir un archivo, ejecutar en el emulador). En comunicación y con la guía del DR, el DA puede acompañar a los estudiantes que necesitan más ayuda.

Información para el DR

Enlaces a la aplicación a utilizar para conocer el entorno de programación.

- App de prueba Etapa 1 (un sensor): [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)
- App de prueba Etapa 1 (dos sensores): [Enlace a la galería](#) o [Enlace a la descarga desde drive](#).

VC 1↓

Inicio con App Inventor

 Desafío

Crear una cuenta en App inventor y probar una aplicación en el emulador del dispositivo.

1. Inicio. Presentación de la propuesta (5 min)

*¿Utilizan aplicaciones en el celular/tablet? ¿Cuáles? ¿Cómo funcionan?
¿Para qué sirven?*

En la puesta en común se pueden distinguir aplicaciones con diferentes propósitos: para jugar, creativas, para comunicarse, entre otros.

Se presenta el proyecto, explicitando que se diseñará y programará una aplicación para jugar y aprender.

2. Desarrollo. Exploración de App inventor (30 min)

¿Qué entornos de programación han utilizado hasta el momento? ¿Se puede construir con ellos una aplicación para usar en distintos dispositivos?

Paso 1: Crear una cuenta en App inventor e instalar el Emulador externo

Cada estudiante debe contar con una **cuenta de gmail** para poder acceder a [App Inventor](#) y crear la cuenta que utilizará durante todo el proyecto. Luego se procede a instalar el emulador que se ejecuta en un dispositivo externo. Deben contar con un dispositivo con sistema operativo Android en el cual se instala la aplicación MIT APP Inventor 2 Companion, esta app es el emulador externo que se usará para probar la aplicación.

 **Importante**

La página de App Inventor puede encontrarse en inglés, pero el mismo puede ser cambiado. Indicar a los estudiantes que cambien el idioma a español para que les resulte más sencillo interactuar con el nuevo entorno de programación.

Paso 2: Explorar el nuevo entorno de programación

¿Qué observan al ingresar a App Inventor? ¿En qué partes se divide la interfaz? ¿Qué componentes encuentran? ¿Para qué creen que sirven estos componentes? ¿Cuántos editores tiene? ¿Cuál es la función de cada uno?

Para facilitar el recorrido del entorno, el DR propone a los estudiantes que incorporen componentes básicos como por ejemplo: botones con diferentes distribuciones y propiedades. Se identifican los dos editores principales: el **editor de diseño** y el **editor de bloques**. El primero permite incorporar los componentes que formarán parte de la aplicación (botones, imágenes, campos de textos, etc.), además permite cambiar **propiedades** que modifican color, posición, tamaño, entre otros. El segundo permite programar el comportamiento de la aplicación, es decir es donde se define como va a funcionar la aplicación. Es importante que se identifique la función que cumple cada editor, para que los estudiantes puedan utilizar a posteriori adecuadamente cada uno.

Paso 3: Probar una aplicación

Se le brinda al estudiante una aplicación sencilla, para que puedan practicar subir un archivo a su cuenta y probarla en el emulador. Además, se aprovecha para que los estudiantes reconozcan los componentes y los bloques que presenta la programación.

Cargar y probar una aplicación

¿Cómo se puede probar la aplicación? ¿Qué dispositivo externo se necesita?

El DR guía el uso del emulador. Al utilizar un dispositivo externo permite probar distintos sensores (componentes no visibles). También posibilita apreciar el diseño de la interfaz adecuada para el dispositivo en particular y realizar ajustes necesarios.

Componentes visibles y no visibles

¿Qué componentes de esta app son visibles? ¿Para qué sirven cada uno? ¿Qué propiedades tienen y cuáles se pueden modificar?. Hay componentes que no se visualizan cuando prueban la aplicación, ¿por qué ocurre esto? ¿Para qué se utilizan?

Los componentes visibles son aquellos que se pueden ver cuando se ejecuta una aplicación, se denominan comúnmente la interfaz del usuario. Pero también hay componentes no visibles que permiten acceso a distintas funcionalidades del dispositivo, por ejemplo los sensores que permiten tomar información externa. Es importante que los estudiantes identifiquen el ingreso de información en componentes visibles y no visibles. Pues esto facilitará reconocer la función de distintos tipos de componentes para incorporar en su propia aplicación.

★ Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de realizar las **Actividades interactivas**.

3. Cierre (10 min)

¿Qué bloques de programación reconocen que hayan usado en otros entornos? ¿Qué bloques identificaron? ¿Qué función realizan? ¿Qué componentes tiene la APP? ¿Para qué se utilizan? ¿Qué otra similitud

encuentran? ¿Qué similitudes tiene App Inventor con los otros entornos utilizados? ¿Y qué diferencias?

Los estudiantes reflexionan y el DR registra las similitudes y diferencias que existen entre distintos entornos de programación. Se alienta la comparación con otros entornos donde se asocian la interfaz y distintos bloques. A través de una puesta en común, el DR, anota en el registro las similitudes o diferencias que vayan descubriendo o relacionado los estudiantes. A modo de ejemplo se presenta la imagen del [Anexo I](#) para que sirva de guía en la identificación.

Registro en Crea

El DR publica en el **Registro Común**. las notas y reflexiones de los intercambios. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Se los invita a resolver las actividades interactivas en las que se repasan los pasos para usar el emulador y los componentes principales.



La Yapa: Propuestas para seguir en casa

Ahora ya conoces App Inventor. **Puedes seguir probando la app**. Realiza distintas modificaciones y vuelve a probarla en el emulador. Por ejemplo: se pueden modificar propiedades (textos, colores, tamaño) o incorporar un nuevo botón y programarlo para que muestre un nuevo mensaje.



ETAPA 2 ↓ El tema de la aplicación	Coordinación dupla pedagógica
<p>En el aula se identifica el tema a abordar que será el insumo fundamental para el desarrollo de la aplicación.</p> <p>Durante la VC, se prueba una aplicación para que los estudiantes identifiquen errores y completen un programa para que continúen familiarizándose con el entorno.</p> <p>Objetivos</p> <hr/> <p>Se espera que los estudiantes:</p> <ul style="list-style-type: none"> • Reconozcan las ventajas de utilizar nombres descriptivos en los componentes de un programa. • Identifiquen, completen y modifiquen errores en un programa asociando entrada y salida esperada. 	<p>Decisiones del DA</p> <ul style="list-style-type: none"> • El recorrido didáctico que se llevará a cabo para alcanzar los propósitos de la instancia de aula. • El nivel de profundidad o alcance en el abordaje del tema elegido. ¿Qué información necesita el DR para alinear su discurso al modo de abordaje? <p>Decisiones conjuntas entre DA y DR:</p> <ul style="list-style-type: none"> • La participación del DA para gestionar las primeras experiencias de los estudiantes con el nuevo entorno será muy valiosa (ingresar al entorno, subir un archivo, ejecutar en el emulador). En comunicación y con la guía del DR, el DA puede acompañar a los estudiantes que necesitan más ayuda. <p>Información para el DR</p> <p>Enlace a la aplicación a utilizar en la Etapa 2.</p> <ul style="list-style-type: none"> • Cuento con Parlanchín: Enlace a la galería o Enlace a la descarga desde drive

AULA ↓

El tema a explorar

Notas para el DA ↓**Propósitos mínimos**

- Acompañar a los estudiantes en la identificación de un tema propicio para desarrollar la aplicación.
- Proponer la elaboración de una pregunta o enunciado que se responda con la identificación de elementos en una imagen. Registrarla en la bitácora.

Propósitos óptimos

- Proponer formas de modelizar el tema elegido que deriven en insumo para la aplicación.

¿Qué tema exploramos?

Se sugiere seleccionar un tema que pueda representarse con una imagen y que los estudiantes deban reconocer partes o lugares de la misma.

Por ejemplo:

- Mapas para reconocer países, departamentos, calles, regiones o lugares destacados.
- La imagen de un organismo, célula o sistema para reconocer sus distintas partes o componentes
- La imagen de una línea de tiempo para ubicar hechos históricos.

En [la bitácora](#) se dispone de un espacio para completar:

- La descripción de la imagen de base del proyecto que permite reconocer o ubicar distintos elementos y del Explorador que recorre la imagen.
- El enunciado que permite reconocer o ubicar elementos en la imagen elegida como base del proyecto.

VC 2 ↓ Explora fallas

Desafío

Esta aplicación no funciona, vamos a corregir errores y completar su programación.

1. Inicio (5 min)

Se retoma lo trabajado en el aula para que los estudiantes cuenten sobre el tema con el cual desarrollarán la aplicación.

2. Desarrollo (30 min)

El DR facilita a los estudiantes el proyecto incompleto “*Aprendo los números con Parlanchín*”. Este proyecto cuenta con los siguientes componentes: un botón Saludar, botones del 1 al 10, un personaje robot y un sensor (acelerómetro). Se invita a los estudiantes que carguen el proyecto en sus cuentas y lo ejecuten en el emulador del dispositivo externo.

Paso 1: Corrige los errores para que Parlanchín cuente correctamente del 1 al 10

¿Qué ocurre al presionar los botones de los números? ¿Por qué ocurre ese error? ¿Es posible arreglar la programación de los botones? ¿Cómo identificar qué bloque deben cambiar?

Al explorar el funcionamiento de la aplicación los estudiantes deben reconocer que al presionar cada botón de los números, no cuenta correctamente del 1 al 10. Se brinda el tiempo a los estudiantes para que encuentren los bloques de programación que deben modificar para corregir los errores y que la aplicación funcione correctamente.

En esta actividad los estudiantes deben asociar el componente del Editor de diseño con el bloque de programación que corresponde en el Editor de bloques y de esta manera poder corregir los errores.

¿Fue fácil corregir el contador de números? ¿Por qué? ¿Qué cambios haría para que resulte más sencillo encontrar un error?

Con guía del DR, identifican la importancia de la elección de los nombres adecuados de los componentes para que un programa sea entendible por otros programadores. De esta forma, se facilita la tarea de localizar errores de programación. También, la importancia de elegir nombres descriptivos para los componentes ya que simplifica identificar los bloques de programación asociados a cada uno. Por ejemplo, si un botón con el nombre `Boton11` al presionarlo debe decir el número 5, en la vista de bloques se dificulta reconocer cual es el componente a programar. Ahora, sí el nombre es `BotonCinco`, se simplifica identificar cuál es el componente que necesito programar para que realice esa acción. Este análisis es importante para que los estudiantes incorporen el hábito de cambiar el nombre a sus componentes y realicen una elección adecuada del mismo, ya que les facilitará la programación de su propia aplicación.

Paso 2: Hay partes de la aplicación que no están programadas, completa la programación que falta.

Se propone a los estudiantes completar la programación de la aplicación. Para esto, deben programar dos componentes: uno visible y uno no visible.

El componente visible es el botón *Saludar*, para que cuando se presione, salude diciendo alguna frase. Se espera que los estudiantes asocien con el contador de números y reconozcan la necesidad de

incorporar el bloque de hacer clic sobre el botón saludar.

Para programar el componente no visible, se pide a los estudiantes que exploren nuevamente la aplicación que probaron en la Etapa 1 y reconozcan la programación de ese componente. De esta forma, deben identificar la programación del sensor acelerómetro cuando se mueve rápidamente el dispositivo donde Parlanchín avisa que no lo sacudan. Se le brinda a los estudiantes el tiempo para que exploren los bloques de programación del componente y, de ser necesario, el DR guía la programación.

Es importante que los estudiantes utilicen en su programa componentes visibles y no visibles porque servirá como aprendizaje en el momento que programen su propia aplicación.



Variantes de complejidad:

Incorpora otra función a la aplicación de Parlanchín. Por ejemplo: que te pregunte el nombre y te salude diciendo el mismo.



Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (10 min)

Entrada, transformación y salida de información

¿Cómo se entera App Inventor de que presionamos un botón? ¿Qué bloque utilizamos para esto? ¿Cómo se entera la aplicación de que se está sacudiendo el dispositivo? ¿Qué bloques utilizamos para esto? ¿Qué hace la aplicación con esa información? Entonces, ¿de qué manera se ingresa información en la aplicación? ¿Los componentes visibles solamente permiten ingresar información?

Se incentiva a los estudiantes a que reconozcan componentes para ingresar información en el dispositivo: los botones y el sensor

acelerómetro. Realizando una puesta en común, el DR registra la función de algunos componentes, propiedades y bloques de programación que se utilizaron. Es importante que se vuelquen componentes que se utilizarán luego en la construcción de la propia aplicación. Además, para generalizar, los estudiantes identifican situaciones cotidianas de ingreso de información a la computadora, los dispositivos involucrados y cómo reacciona la computadora. Por ejemplo, hacer clic en un video que quieren visualizar, hacer doble clic en el acceso directo de un programa, etc.

Registro en Crea

Se publica en el **Registro Común** las notas y reflexiones de los intercambios. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Se los invita a resolver las actividades interactivas en las que se repasan componentes y bloques de programación.

Cada proyecto debe contar con una carpeta y documento colaborativo donde vuelquen el material que se utilizará en el diseño de la aplicación (imágenes y sonidos)



La Yapa: Propuestas para seguir en casa

¿Te gustó completar la programación de una aplicación? Te proponemos otra app: [Suma números](#), observa y completa su programación.



ETAPA 3 ↓ Diseño de la interfaz

En el aula se invita a realizar el bosquejo de la interfaz de la aplicación. Definir colores, tipografía, distribución de los componentes, esta es una actividad que pueden realizar en conjunto con Artes Visuales.

Durante la VC, comienzan un proyecto nuevo y construyen la interfaz y el plan de desarrollo para que sirva de guía de la programación de la aplicación.

Objetivos

Se espera que los estudiantes:

- Identifiquen los componentes visibles y no visibles necesarios para el diseño de su interfaz.
- Diseñen la interfaz de usuario de su aplicación.
- Escriban un plan de desarrollo que contemple la programación de los distintos componentes incorporados en la interfaz.

Coordinación dupla pedagógica

Decisiones del DA

- El recorrido didáctico que se llevará a cabo para alcanzar los propósitos de la instancia de aula.
- El proyecto se verá potenciado en la medida que los estudiantes puedan aportar sus propias creaciones de:
 - El explorador (personaje con el que se juega en la App)
 - La imagen de base (Escenario de juego)

Esta tarea puede articularse con un proyecto de Artes Visuales. Las ilustraciones pueden realizarse con la técnica que se considere oportuna y luego digitalizarse tomando una foto, o bien trabajarse directamente en forma digital.

- Dudas o aportes de los niños que hayan surgido y puedan ser relevantes a la hora de trabajar en la VC el diseño de la interfaz.

Decisiones conjuntas entre DA y DR:

- Acordar el medio que se utilizará para realizar el diseño de la interfaz. Si se decide utilizar la plantilla digital el DR, debe realizar las copias del documento para cada grupo.

Información para el DR

Proyecto de ejemplo de avance para la Etapa 3:

- [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓

Bosquejo de la interfaz de la aplicación

Notas para el DA ↓

**Propósitos mínimos**

- Orientar a los estudiantes en el bosquejo de la interfaz de la aplicación donde representen los elementos principales (imagen a explorar, explorador, botón de inicio de partida).
- Acompañar a los estudiantes en la generación de los archivos de imagen (.png) y guardado en una carpeta compartida en Drive.
- Brindar el espacio para completar la bitácora.

Propósitos óptimos

- Continuar con la modelización del tema elegido.
- Favorecer el uso de porcentajes para reconocer variaciones en una área determinada.

El diseño de la interfaz es un momento de mucha creatividad, es la oportunidad de pensar en botones, gráficos, iconos, fondos, etc, que necesita la aplicación. Pero cuando nos encontramos con la pantalla en blanco pueden surgir algunas dudas: ¿Qué componentes hay que incorporar? ¿Para qué? ¿De qué tamaño? ¿Cuál sería la distribución? ¿Qué colores utilizar?

Recomendaciones sobre tamaño y resolución de las imágenes: La imagen de base elegida debe ser más larga que ancha para lograr un mejor ajuste en la pantalla de los dispositivos. También es importante que el fondo sea transparente o blanco, para que sólo se destaque la imagen de base. En el [Anexo 2](#) se encuentran ejemplos de interfaz completa de aplicaciones para que el DA tenga de referencia.

Es importante que el bosquejo de interfaz de la aplicación cuente como mínimo con:

- un explorador, (nuestro personaje en la APP)
- una imagen de base para identificar elementos de la misma, (nuestro escenario de juego)

El bosquejo se puede hacer utilizando: lápiz y papel, un papelógrafo, alguna aplicación de dibujo o utilizando la plantilla [digital](#) (en el [Anexo 3](#) se proporciona una plantilla para imprimir). Es una oportunidad importante para trabajar con porcentajes para determinar qué espacio ocupará cada elemento.

Los estudiantes dibujan o descargan las imágenes necesarias para usar en la aplicación y las guardan en una carpeta del proyecto.

En la bitácora se dispone de espacios para completar los elementos principales incluidos en el bosquejo de interfaz y para que describan brevemente la función de cada uno. Este insumo servirá para armar el **Plan de Desarrollo en la VC siguiente**.

VC 3↓

La interfaz en App Inventor y Plan de desarrollo

 Desafío

Armar el diseño de la interfaz y el Plan de Desarrollo de la aplicación.

1. Inicio. Diseñar y probar la interfaz en el móvil (10 minutos)

Se retoma el diseño realizado en el aula y los estudiantes describen el bosquejo de interfaz que hicieron.

Luego, con guía del DR, arman un proyecto nuevo en App Inventor. Cada elemento que tienen en el bosquejo de la interfaz se materializa en el proyecto en un determinado componente. Un primer diseño puede contemplar sólo los componentes visibles *Imagen de base* y *Explorador*. Para esto, se necesita incorporar componentes de la Paleta Dibujo y animación.

 Atención

Puede ocurrir que la interfaz diseñada esté incompleta. Pero, para el armado del proyecto, se debe respetar la propuesta de los estudiantes y dejar que la completen a medida que se avance. De esta forma, pueden identificar componentes que faltan en el momento en que los necesiten para el desarrollo de la aplicación. Cada cambio iterativo que se realice sobre la interfaz se verá también reflejado en el Plan de Desarrollo, esto propicia una forma de pensar la resolución para que puedan realizar el mismo proceso en otros proyectos.

Paleta Dibujo y animación

Aquí se encuentran dos componentes principales para la aplicación:

Lienzo: es el espacio que ocupará la *Imagen de Base*. Este componente debe tener **un tamaño fijo**, pues en la Etapa 5 se incorporan los puntos (*Pelotas*) para ubicar la sección de la imagen base y estos deben estar siempre en la misma posición.

SpriteImagen: es el componente que permitirá incorporar la imagen que va a tener movimiento: el *Explorador*.

Si, además, en el bosquejo de la interfaz tienen otros componentes como botones y texto, se deben tener en cuenta las siguientes paletas:

Paleta Interfaz de usuario: Elementos de uso común por los usuarios

Aquí pueden encontrar componentes propios de una interfaz de aplicación: botones y etiquetas. Se incorporan dentro de un Contenedor de Disposición.

Paleta Disposición: Segmentación de la interfaz

Posiblemente en los bosquejos diseñados por los estudiantes no se encuentre incorporado ningún componente de disposición. El DR, teniendo en cuenta la propuesta de los estudiantes, puede sugerir incorporar componentes de la Paleta Disposición para dividir la pantalla en diferentes áreas. El trabajo de diseño consiste en crear contenedores horizontales, verticales o matriciales y ajustar las propiedades para distribuir los componentes. En estos contenedores se incorporan principalmente botones y etiquetas de textos. La participación de el DA en este punto es importante para trabajar con porcentajes, por ejemplo: si se considera el ancho de la interfaz el 100% y se incorporan dos componentes horizontales, determinar qué porcentaje se le asigna a cada uno, para luego ajustar las propiedades asociadas.

Sugerencia

AppInventor es una herramienta muy poderosa. Al principio puede ser abrumadora, pero debemos enfocarnos en el desafío y concentrarnos en los componentes sugeridos que deben utilizarse. ¡Tendremos tiempo para explorar y conocer más bloques cuando comprendamos mejor cómo funciona el entorno y qué podemos hacer con ellos!

A medida que incorporen los componentes, el DR, incentiva a los estudiantes a que nombren cada componente de forma que sea fácil identificarlos cuando se programe y a que ajusten las propiedades asociadas. Luego, los estudiantes deben ejecutar su proyecto en el emulador de sus dispositivos y corroborar que el diseño cumple con la propuesta diseñada.

2. Desarrollo. Plan de desarrollo (25-30 min)

¿Cómo funciona la aplicación? ¿Para qué se utilizará la aplicación? ¿Cuándo se gana? ¿Cuándo se pierde? ¿Qué elementos se deben programar? ¿Cuál se debe programar primero y cuál después? ¿Por qué?
En propuestas anteriores, se les proporcionaba a los estudiantes el plan de desarrollo, por lo cual ahora se propone que sean ellos quienes piensen qué deben programar y el orden en que lo harán.

Teniendo en cuenta la interfaz diseñada, en grupo, debaten y escriben los pasos que deben programar para que su aplicación funcione correctamente.

Atención

Puede ocurrir que el plan esté incompleto, o que no siga la secuencia adecuada, pero se debe respetar la propuesta de los estudiantes y dejar que aborden los problemas “del modo equivocado”. Para que puedan identificar tareas que faltan o que deben desarrollar antes en su plan, y de este modo se apropien de la planificación. Este plan de desarrollo está pensado para que sea dinámico, se puede ir modificando acorde las dificultades que encuentren los estudiantes para llevar a cabo lo planificado. Cada cambio iterativo que se realice sobre el plan para luego continuar con el desarrollo favorece la independencia en la forma de pensar la resolución y permite que apliquen esta forma de trabajo en otros proyectos.

Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (10 minutos)

Plan de desarrollo común

Posiblemente, los estudiantes hayan escrito distintos caminos para llegar a completar la programación de la aplicación, por lo cual, se hace una puesta en común para obtener un plan de desarrollo unificado para el curso.

Una vez escrito se repasa el plan de desarrollo para analizar si falta alguna tarea o si están de acuerdo en el orden.

De esta manera, los estudiantes pueden identificar qué agrega cada paso y qué queda por resolver. El DR completa el Plan de desarrollo común teniendo en cuenta los aportes de cada uno de los grupos y los estudiantes completan en su bitácora el Plan de desarrollo consensuado. En el [Anexo 4](#) se encuentra un ejemplo de Plan de desarrollo.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase los componentes visibles que incorporaron en la interfaz y propiedades asociadas.



La Yapa: Propuestas para seguir en casa

Ahora ya sabes armar una interfaz de una aplicación. Te invito a armar otra interfaz: ¿Cómo te gustaría que fuese? ¿Qué elementos contendría? ¿Sobre qué tema sería? ¿Quiénes la van a utilizar?



ETAPA 4 ↓ El Explorador

En esta etapa se propone programar el movimiento del personaje principal. En el aula se proporciona un espacio de juego para reconocer la mecánica que utilizará la aplicación.

Durante la VC, los estudiantes programan el comportamiento del Explorador utilizando los datos de entrada que brinda un sensor para que se desplace por la pantalla.

En la reflexión final, se explicitan los pasos para la reutilización de programas como una manera de organizar la elaboración de soluciones computacionales.

Objetivos

Se espera que los estudiantes:

- Identifiquen la necesidad de incorporar un sensor como un componente para capturar datos de entrada de un programa.
- Determinen y programen expresiones aritméticas que representen las relaciones entre los valores del sensor y el movimiento en pantalla esperado de un componente.
- Reconocer una secuencia organizada de pasos para la reutilización de programas al momento de resolver problemas computacionales.

Coordinación dupla pedagógica

Decisiones del DA

- El recorrido didáctico que se llevará a cabo para alcanzar los propósitos de la instancia de aula.
- La forma de modelizar y llevar a cabo una instancia del juego en el aula.
- Las actividades de aula para fortalecer el trabajo con coordenadas cartesianas.
- Dudas o aportes de los niños que hayan surgido y puedan ser relevantes a la hora de trabajar en la VC.

Información para el DR

- La forma en que se aborda el contenido del proyecto en el Aula, para relacionar la mecánica utilizada con la programación en la VC.

Proyecto de ejemplo de avance para la Etapa 4:

- Con sensor acelerómetro: [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)
- Con sensor de orientación: [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓

Funcionamiento del juego

Propósitos mínimos

- Favorecer, desde la experimentación, la comprensión del contenido curricular elegido y el funcionamiento del juego a programar.
- Establecer una referencia del sistema de coordenadas cartesiano y la pantalla de una aplicación y asociar suma y resta de coordenadas con desplazamientos en el plano.

Propósitos óptimos

- Articular la producción de los elementos visuales del juego con el Área de Conocimiento Artístico.

Notas para el DA ↓**Fortalecer el contenido curricular.**

Para modelizar la aplicación y poner en juego los conocimientos necesarios, se propone un juego. Dibujar con tiza en el piso la imagen de fondo del tema elegido (el mapa, organismo, esquema). Cada estudiante toma el rol de Explorador alternadamente para llegar al destino que debe reconocer en la imagen.

Por ejemplo: la imagen dibujada es el mapa de Uruguay, el docente de aula nombra un departamento y el estudiante debe llegar a ese sitio antes de que lo toquen con una pelota de papel o antes de un determinado tiempo.

Coordenadas cartesianas

Para trabajar con pares de coordenadas, se puede recurrir a un buen conjunto de recursos: coordenadas geográficas (Geografía), tabla pitagórica (Matemática), interpretación de gráficas (Física), el juego de la Batalla Naval, o este proyecto en App inventor.

[Coordenadas Cartesianas en App Inventor](#)

VC 4 ↓ Movimiento del Explorador

1. Inicio (5 min)

¿Por qué es importante jugar al juego que van a programar?
¿Reconocieron los elementos de la interfaz que diseñaron desde el principio o fueron descubriendo algunos a medida que jugaban?

El objetivo del DR es establecer una relación entre la interfaz construida en la etapa anterior y la dinámica de juego que llevaron a cabo en el aula. Los estudiantes recuperan el *Plan de Desarrollo* e identifican qué componente van a programar primero.

2. Desarrollo(30 min)

Paso 1: Posición del Explorador en la pantalla

¿Cómo podemos conocer la posición de un componente en App Inventor? ¿Qué propiedades cambian cuando movemos un componente? ¿Qué pasa si las modificamos manualmente? ¿Cómo se mueve un componente cuando aumentamos el valor de X o de Y? ¿Dónde se ubica el Explorador si cambiamos las propiedades de X e Y por 0?

Al examinar las propiedades del componente *Explorador* se observa que dos de sus propiedades son X e Y. Se permite que los estudiante experimenten realizando modificaciones a esas propiedades, pues ese reconocimiento servirá de insumo para cuando se programe el sensor de inclinación. Se concluye que estas propiedades permiten el desplazamiento del objeto hacia la derecha a medida que se incrementa el valor de X y hacia abajo a medida que se incrementa el valor de Y. Es oportuno comparar con otros entornos de programación utilizados en años anteriores: ¿Dónde han visto estas coordenadas? ¿Cuáles son las similitudes y diferencias?

💡 Desafío

El Explorador debe desplazarse por la pantalla cuando se inclina el dispositivo.

Paso 2: Identificar el uso de un sensor

¿Cuándo debe moverse el Explorador? ¿Qué componente permite en App Inventor detectar la inclinación del dispositivo? En las aplicaciones que probaron en las etapas anteriores, ¿había alguna que detectaba la inclinación del dispositivo? ¿Qué componente utilizaba para ello? ¿Qué bloque se utiliza para programar ese componente?

El objetivo es que los estudiantes revisen las aplicaciones utilizadas; para identificar de qué manera se ingresa la inclinación del dispositivo, reconocer qué componente no visible deben incorporar en su proyecto y que bloque se utiliza para programarlo.

🔔 Atención

En este momento se motiva a los estudiantes a que revisen las aplicaciones usadas en las etapas 1 y 2, para que analicen el código e identifiquen qué parte del programa es la que detecta la inclinación del dispositivo. El DR, presenta a **la mochila** como la herramienta que permite la reutilización de código entre proyectos.



¿En qué otros programas han trabajado con eventos? ¿Funcionan esos bloques igual que en este programa?

Con la guía del DR, reconocen el uso de un sensor y del **evento** asociado. En App inventor los componentes tienen bloques de

programación que son **Eventos**. Estos son partes de un programa que pueden suceder, sin que sepamos a priori cuando van a ocurrir. Al programar utilizamos los Eventos para determinar cómo queremos que reaccione nuestro programa cuando ocurra.

Por ejemplo, al presionar el botón “saludar” de la aplicación que probaron en la Etapa 1, ¿Qué ocurría? ¿Y si sacudían el celular?. En esa aplicación hay distintos eventos, uno que espera a que se presione el botón, cuando se presiona, ejecuta las instrucciones para saludar. También hay otro evento que espera la modificación en los datos que recibe de entrada el sensor acelerómetro, cuando se modifica ese dato se ejecuta un evento y decía “no me muevas tan rápido”. El DR, puede a la vez ir mostrando la programación de la aplicación para que los estudiantes identifiquen los bloques que son eventos e incorporen en la mochila esas porciones del programa para reutilizarlos en el sus proyectos..

Sugerencia

Exploración guiada: Las intervenciones docentes, deberán hacerse desde una perspectiva orientadora que permita acompañar a los estudiantes. Ellos deben ser capaces de deducir, razonar y detectar formas de ingreso de información en la aplicación. Al alcanzar esta etapa han usado y explorado otras aplicaciones que le servirán como base para programar su propio sensor.

Importante

Para programar el movimiento del Explorador puede utilizarse **uno** de los sensores que se encuentran en los dispositivos: Orientación o Acelerómetro. Se recomienda usar el Acelerómetro en el caso en que los estudiantes no cuenten con información sobre los sensores en sus dispositivos. Además, simplifica la asociación en el desplazamiento en los ejes X e Y.

Paso 3: Programar el movimiento del Explorador

El DR brinda el espacio para que los estudiantes incorporen el **Sensor**. Los estudiantes exploran las propiedades y bloques asociados e incorporan el **Evento** que detecta la inclinación del dispositivo para que

el *Explorador* se mueva por la pantalla a medida que se lo inclina. Para lograr esto deben armar una expresión numérica que involucre las coordenadas cartesianas del *Explorador* y los valores que detecta el sensor. Se alienta a los estudiantes a experimentar con distintas fórmulas para conseguir el efecto deseado con la guía por parte del DA que contemple sus conocimientos previos en el área de matemática. También pueden reconocer en las aplicaciones usadas anteriormente la parte del programa que necesitan replicar en su aplicación.

Variantes de complejidad

Ahora que ya se desplaza tu Explorador, puedes modificar la velocidad del desplazamiento y así ajustar la misma a tu agrado.

Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (10 minutos)

Componentes y bloques de programación: *¿Qué bloque de programación debo utilizar? ¿En que debo pensar antes de decidir? ¿En qué situaciones de su vida cotidiana se reacciona ante un evento?*

En App Inventor los componentes tienen distintos bloques eventos que pueden programarse. Para decidir qué bloque debe utilizarse al programar, primero se debe realizar una exploración sobre los bloques que ofrece la herramienta. Luego pensar qué evento debería utilizar para lograr la acción que busco. Se guía a los estudiantes a identificar eventos en nuestro entorno, por ejemplo, hacer clic en tiktok sobre un video o al presionar un icono de una aplicación en el celular ¿Qué ocurre?

Reutilización de código de un programa: *¿A qué programa de los que probaron se parece el movimiento del Explorador? ¿Por qué? ¿Nos ayudó al programar haber identificado estos parecidos? ¿De qué manera? Los docentes acompañan a los grupos para que identifiquen*

en su propia experiencia de programación cómo llevaron a cabo la combinación y adaptación de programas existentes.

Programas aparentemente distintos, pueden tener un comportamiento similar y ser capaces de identificar estas similitudes sirve de punto de partida para el diseño de su propio programa. El adaptar y reutilizar programas es sumamente frecuente en programación, ya que es una manera de evitar resolver de nuevo problemas que ya están resueltos y utilizar estas soluciones para construir programas que resuelvan problemas nuevos.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase el componente no visible incorporado en su proyecto, asociando funciones, eventos y coordenadas cartesianas.



La Yapa: Propuestas para seguir en casa

¡Felicidades! Lograste que el *Explorador* se desplace. Ahora, puedes incorporar efectos de sonido. Puedes grabar tus propios sonidos o descargarlos de internet. Luego revisa qué componente necesitas incorporar y en qué partes del programa habría que iniciar la reproducción de los mismos.



ETAPA 5 ↓ Jugar una partida

En esta etapa van a jugar la primera partida. En el Aula se aborda el contenido definido para el proyecto para escribir reglas que determinen aciertos del juego.

Durante la VC, se programa una de las características principales de la aplicación: que el jugador juegue una partida y gane.

Al terminar esta etapa se tiene una primera versión incompleta pero funcional del juego.

Objetivos

Se espera que los estudiantes:

- Establezcan generalizaciones e identifiquen condiciones sobre el contenido definido para el proyecto.
- Formulen reglas de juego con estructura de alternativa condicional.
- Identifiquen la necesidad de incorporar nuevos componentes para programar una regla de acierto del juego.
- Reconozcan a la computadora como una máquina que procesa información de entrada y produce un resultado en consecuencia.

Coordinación dupla pedagógica

Decisiones del DA

- La actividad propuesta para el aula contempla escribir el enunciado y reglas del juego, el DA podrá ampliar esta idea sumando los aspectos que considere oportunos para el tema a utilizar en la aplicación.

Información para el DR

- La forma en que se aborda el contenido del proyecto en el Aula, para relacionar la mecánica utilizada con la programación en la VC.

Proyecto de ejemplo de avance para la Etapa 5:

- [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓ El enunciado

Notas para el DA ↓



Propósitos mínimos

- Proponer la elaboración de una pregunta o enunciado que se responde con la identificación de elementos en una imagen y registrarla en la bitácora.
- Establecer las pautas necesarias para que los estudiantes definan el enunciado a cumplir en el juego, el conjunto de respuestas posibles y la condición que determina si una respuesta es correcta.

Propósitos óptimos

- Diseñar actividades que permitan a los estudiantes en su rol de jugadores, construir las estrategias necesarias para resolver la consigna que planteará la aplicación.

Pensar en el enunciado

Para esta instancia de aula y las siguientes, la excusa de prepararse para ser un excelente jugador de su propio juego, constituye una oportunidad para continuar trabajando con el contenido seleccionado para el desarrollo de la aplicación.

Los estudiantes deberán decidir:

- ¿Qué enunciado funcionará como “consigna”?

Para acercarse a cómo decidir cuándo el *Explorador* llegó a un objetivo correcto (es decir, que cumple la consigna planteada), deberán explorar y establecer una regla o generalización. Se sugiere diseñar actividades que favorezcan un trabajo exploratorio que habilite a los estudiantes a elaborar estrategias que valgan “siempre” e identificar un enunciado o pregunta para reconocer objetivos de la imagen de base.

Escribir en la [bitácora](#) la descripción del funcionamiento del juego y el enunciado que permite identificar un elemento en la imagen de base del proyecto.

Ejemplo:

La aplicación consiste en un juego donde un enunciado se cumple a partir del posicionamiento de un *Explorador*:

Enunciado

Lleva a la exploradora al departamento *Artigas*

Regla

Si la exploradora llega a Artigas entonces gana.

VC 5↓ Ganar el juego

1. Repaso del Plan de Desarrollo y de la Interfaz (10 min)

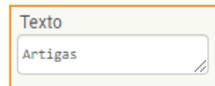
Se retoma lo trabajado en el aula para completar la interfaz de la aplicación en esta VC.

¿Dónde muestra la aplicación el enunciado escrito? ¿En qué parte se muestra la variación del enunciado? ¿Qué componentes necesitamos agregar a la interfaz para poder mostrar ese contenido?

Si todavía no lo hicieron, los estudiantes deben agregar a sus proyectos dos componentes de la **Paleta Interfaz** del tipo *Etiqueta*. Recordar que estos elementos se incorporan dentro de un componente **Disposición** para realizar una distribución porcentual adecuada. Los estudiantes deben modificar las propiedades para que muestren en la interfaz de la aplicación la parte del enunciado que se mantiene fija en una *Etiqueta* y la parte variable que representa los puntos a reconocer en la Imagen de Base en otra *Etiqueta*.

Atención

En esta etapa la parte variable del enunciado se asigna desde la propiedad *Texto*. En una etapa posterior se programa para que esta asignación se realice de forma aleatoria.



2. Desarrollo (30 min)

Se recupera el **Plan de Desarrollo** para identificar cuál es la próxima tarea. Se espera que en el desarrollo se cuente con un punto que contemple la programación necesaria para resolver el desafío de esta etapa. En caso contrario, el DR, orienta a los estudiantes para que se incorporen a la acción requerida.

Parte 1: Marcar el lugar a donde se debe llegar

Desafío

Mostrar un mensaje de acierto cuando el Explorador llegue a un objetivo.

¿Cuándo se gana el juego? ¿Cómo se detecta que el *Explorador* llegó al lugar que indica el enunciado? ¿Qué componente debemos incorporar? Con la guía del DR, se incorpora de la **Paleta Dibujo y animación** el componente *Pelota* para marcar uno de los lugares donde debe llegar el Explorador. Se revisan sus propiedades y se realizan los ajustes necesarios.

Atención

Para la propiedad *ColorDePintura* del componente *Pelota* se debe elegir *Ninguno*, para que no se vea el componente cuando se juegue en la aplicación. Pero, mientras se realizan pruebas, se recomienda que tenga un color visible para que se pueda probar el funcionamiento correcto del proyecto.



Parte 2: Programar que se muestre un mensaje de éxito cuando el jugador llegue al objetivo correcto.

En esta parte se busca motivar para que los estudiantes reconozcan la necesidad de usar la alternativa condicional, a partir del problema: El *Explorador* debe llegar a X lugar para ganar.

Se puede escribir usando el enunciado: **Si** el Explorador llega a la meta **entonces** gana, para luego intentar armar la estructura en App Inventor.

El DR brinda el espacio para que los estudiantes exploren los bloques asociados al componente *Pelota* e identifiquen el Evento *EnColisiónCon* y programen ese evento.

El DR, invita a los estudiantes a que revisen los proyectos que probaron en la Etapa 1 y Etapa 2, para mirar sus programas e identificar si incluyen alternativas condicionales. De esta forma,

reconocer bloques que se requieren utilizar en App Inventor en el armado de esa estructura. Como esta es la primera vez que los utilizan en este entorno, puede ser necesaria una guía más intensa para combinar adecuadamente los bloques.

Una vez armada la condición de la alternativa condicional, cuando ocurre que es verdadera debe mostrar un mensaje de acierto. Para ello se incorpora un nuevo componente.

Mostrar mensaje de éxito

¿Cómo se mostraban mensajes en los otros entornos de programación? ¿Qué bloques se utilizaban? ¿Hay esos bloques en App Inventor?

El DR, puede abrir algún entorno conocido por los estudiantes, y recordar que bloques utilizaban para mostrar un mensaje en la ejecución. En App Inventor, para realizar esa acción, se debe incorporar un nuevo componente no visible de la **Paleta Interfaz de usuario**. El componente *Notificador* permite emitir avisos durante la ejecución de una aplicación. Se brinda el espacio para que los estudiantes incorporen el nuevo componente, adapten las propiedades al proyecto y completen la programación de la alternativa condicional.

Atención

En esta etapa solo se programa llegar a un solo objetivo. En la etapa 7 se programan los demás.

Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3.Cierre (10 min)

La alternativa condicional como regla. ¿Qué tipos de reglas nos permite expresar la alternativa condicional? ¿Qué situaciones de nuestra vida cotidiana podríamos expresar con esta estructura?

Se retoma la experiencia de programación para explicitar que construyeron un programa en App Inventor que expresa reglas para aciertos y fallos. Los estudiantes identifican que fue posible incorporar reglas del tipo “Si se cumple una condición entonces realizar una acción”. Se propone buscar ejemplos de estas situaciones en nuestra vida cotidiana y escribirlos bajo esta estructura para reforzar su generalización.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase alternativas condicionales.



La Yapa: Propuestas para seguir en casa

¡Felicidades! Lograste tener la primera versión de tu juego. Ahora, puedes incorporar *una imagen y sonido para cuando el juego se gane, debe estar relacionada con la temática del juego o puede ser el mismo Explorador en distintas posiciones*. ¿Qué componente debes agregar? ¿En qué parte incorporas la programación de este nuevo componente?



ETAPA 6 ↓ Una lista de objetivos

En el aula los estudiantes continúan abordando el contenido definido para el proyecto y se familiarizan con la regla que caracteriza los aciertos en el juego.

Durante la VC, se realiza un juego desenchufado para identificar la utilidad de tener una lista. Se incorpora una lista en la programación de la aplicación para identificar todos los objetivos del juego.

Al terminar esta etapa se tiene una segunda versión incompleta pero funcional del juego.

Objetivos

Se espera que los estudiantes sean capaces de:

- Utilizar una lista para almacenar y representar un conjunto de valores.
- Aproximarse a la noción de listas como una herramienta para representar un conjunto de valores en un programa.

Coordinación dupla pedagógica

Decisiones del DA

- La actividad propuesta para el aula contempla escribir el enunciado, identificando la parte que se repite y la parte variable que permite reconocer distintos elementos en una imagen. El DA podrá ampliar esta idea sumando los aspectos que considere oportunos para el tema a utilizar en la aplicación.

Información para el DR

- La forma en que se aborda el contenido del proyecto en el Aula, para relacionar la variación de los enunciados escritos con la creación de la Lista en la VC.

Proyecto de ejemplo de avance para la Etapa 6:

- [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓**Escribir una lista de objetivos****Notas para el DA ↓****Propósitos mínimos**

- Favorecer, desde la experimentación, la comprensión del contenido curricular elegido y el funcionamiento del juego a programar contemplando todas las variaciones de los enunciados.

Propósitos óptimos

- Diseñar actividades que permitan a los estudiantes en su rol de jugadores, construir las estrategias necesarias para resolver la consigna que planteará la aplicación.
- Seleccionar con los estudiantes y trabajar una canción con estribillo sobre algún tema vinculado a cualquier área del conocimiento (Lengua, Ciencias Sociales, Arte, etc.)

Escribir los elementos a los que llega el Explorador

Los estudiantes retoman el enunciado escrito en la etapa anterior para completar ahora con **todos** los objetivos a reconocer de la imagen:

- ¿Cuáles son los posibles “objetivos”?
- ¿Sirve el enunciado escrito como “consigna” para reconocer distintos objetivos?

Escribir en la [bitácora](#) los enunciados en el cual se identifique la parte que se mantiene sin cambios y la parte variable, que permite identificar diferentes elementos en la imagen de base del proyecto.

Ejemplo:

La aplicación consiste en un juego donde un enunciado se cumple a partir del posicionamiento de un *Explorador*:

Enunciado

Lleva a la exploradora al departamento (regularidad) *Artigas*(variación)
Lleva a la exploradora al departamento (regularidad) *Soriano*(variación)

La canción será utilizada en la Etapa 7, pueden elegir alguna que hayan trabajado durante el año, siempre y cuando tenga un estribillo que se repita.

VC 6 ↓ Crear una lista

Desafío

Mostrar de forma aleatoria el nombre de los distintos objetivos

1. Inicio (15 min)

Un breve intercambio sobre la necesidad de incorporar más objetivos a la aplicación, habilita la presentación del juego **Recuerda++**.

En esta actividad se busca que los estudiantes comprendan la utilidad de contar con una **Lista como herramienta para recordar una colección de elementos homogéneos**.

Se divide en equipos y se definen los roles: un estudiante *Memoria*, quien tendrá la tarea de recordar elementos y estudiantes *Buscadores* quienes deben conseguir los elementos que indique *Memoria*.

La dinámica cuenta con tres momentos. Brindar poco tiempo a cada uno, contribuirá a instalar un ambiente lúdico:

Momento 1: Un estudiante Memoria

1. El DR muestra una lista de 15 elementos (útiles escolares) solo a los estudiantes *Memoria* de cada equipo.
2. *Memoria* debe ir a su equipo y nombrar de a uno a los *Buscadores* los elementos que deben conseguir.
3. *Memoria* muestra al DR los elementos que consiguió. Gana el juego el equipo que recuerda y consigue la mayor cantidad de elementos.

Momento 2: Dispositivo para recordar

Se realizan los mismos pasos que en el nivel 1, pero ahora, además, pueden tener un papel para anotar y recordar elementos. Se les pide a los estudiantes *Memoria* que pongan un nombre a su papel para identificar lo que va a anotar.

Momento 3: Se agregan y eliminan elementos

Se realizan los mismos pasos que en el nivel 1, pero ahora, el DR debe eliminar uno de los elementos y agregar más a la lista.

¿En qué nivel tuvieron mayor facilidad al recordar los elementos? ¿Qué rol juega el papel que usaron? ¿Se pudieron incorporar nuevos elementos? ¿Se pudo eliminar un elemento?

Los estudiantes observan que el papel sirvió como dispositivo para recordar los elementos. Esto fue necesario para no olvidar y recordar todos los elementos que solicita el DR.

Los docentes guiarán al grupo a reconocer que:

1. Al **comienzo** del momento 2, solo tenían un papel. Después, durante el desarrollo del juego anotan todos los elementos que solicitan siguiendo un orden. De esta forma escribieron una lista de elementos.
2. **Recuperan** los elementos anotados, cuando leen el papel a sus compañeros.
3. **Modifican** los elementos anotados, cuando eliminan un elemento que no se pedía e incorporan nuevos.

Este desagregado permitirá asociar el papel a una herramienta de programación para recordar varios datos homogéneos: **las listas**.

¿Qué nombre puede tener esta lista? Es necesario resaltar la importancia del nombre que se le da a la lista ya que si bien sus elementos fueron cambiando sus elementos pertenecen a un mismo conjunto.

2. Desarrollo (15 min)

Parte 1: una lista en nuestro proyecto

¿Qué datos necesitan que se recuerden en su aplicación? ¿Cómo podemos hacer para que nuestro programa recuerde muchos datos? (asociar esta pregunta a la temática de la aplicación que desarrollan, ej: recuerde muchos departamentos)

El DR muestra en su pantalla la categoría Bloques integrados: *Listas* y señala que allí se encuentran los bloques que permiten crear, modificar

y conocer un valor de una lista, fomentando que los estudiantes establezcan relaciones entre estos bloques y el juego **Recuerda++**.

El DR brinda el espacio para que los estudiantes agreguen una lista a su proyecto y carguen la misma con los elementos trabajados en el aula.

Atención

En esta etapa la lista se completa con tres elementos. Posteriormente se pueden incorporar los restantes.

Parte 2: Mostrar los enunciados cambiando la parte variable en la aplicación

¿Dónde muestra la aplicación el enunciado escrito? ¿En qué parte se muestra la variación del enunciado? ¿Qué componentes necesitamos modificar de la interfaz para poder mostrar en la parte variable los elementos que tenemos en la lista?

Los estudiantes recuerdan en qué lugar de la aplicación se muestra el enunciado y la parte variable del mismo. En la etapa 5 se incorporó en el proyecto el enunciado y la parte variable modificando las propiedades de los componentes Etiquetas. Esta vez, los estudiantes deben modificar la parte variable con los elementos de la lista y mostrar de a un elemento en *Etiqueta* correspondiente.

En este momento se debe programar un componente *Botón*, para que cada vez que se lo presione cambie la parte variable del enunciado.

Se **revisa la interfaz y el plan de desarrollo** para completar ambos incorporando este nuevo componente.

Atención

Recordar la importancia de la elección de los nombres adecuados de los componentes para facilitar la programación.

Antes de implementar la programación del botón, es importante que el DR muestre y practique con los estudiantes extraer un elemento de una lista. Ejemplo mostrar la siguiente lista y preguntar: ¿Qué día se

encuentra en la posición 2? ¿Y en la posición 6? ¿Cómo puedo hacer para que muestre un día en cualquier posición?

lunes	martes	miércoles	jueves	viernes
-------	--------	-----------	--------	---------

Los estudiantes exploran los eventos del botón y de la lista para que, con la guía del DR, programen que al hacer clic sobre el *Botón* en la *Etiqueta*, en pantalla se muestre de forma aleatoria un elemento de la lista.

Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (10 min)

*¿En qué situaciones de nuestra vida cotidiana podríamos necesitar una **lista**? ¿Qué programas o plataformas conocen donde se almacena una lista de elementos?*

Repasamos con los estudiantes las posibilidades que dan las **listas**. Se hace hincapié en que las listas sirven para recordar un conjunto homogéneo de datos y que, una vez definida, se puede ir leyendo los valores en distinto orden.

Si es necesario el DR ofrece algunos ejemplos:

- Compras: cuando se va a realizar una compra de muchos elementos escribimos una lista para recordar y no olvidar nada.
- Música: las plataformas de música nos permiten armar una lista con nuestras canciones favoritas.
- Juegos: muestra una lista con el ranking de los mejores jugadores.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes. Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase listas.



La Yapa: Propuestas para seguir en casa

¡Ya tienes tus primeros elementos en la lista!. Ahora puedes agregar todos los elementos que necesites reconocer en tu imagen.



ETAPA 7 ↓

Llegar a todos los objetivos

En esta etapa van a jugar varias partidas. En el Aula se aborda la escritura de canciones para identificar estrofas que se repiten y la forma de nombrarlas.

Durante la VC, se programa el jugador para que juegue varias partidas donde puede ganar o perder. Se introduce la noción de procedimiento como una forma de encapsular tareas y reutilizar código.

Al terminar esta etapa tienen una versión completa del juego.

Objetivos

Se espera que los estudiantes sean capaces de:

- Reconocer los procedimientos como una forma para dividir un problema en subproblemas y como estrategia para reutilizar código.
- Utilizar procedimientos para mejorar la legibilidad de sus programas.
- Utilizar distintas herramientas de programación trabajadas en etapas anteriores para completar el proyecto.

Coordinación dupla pedagógica**Decisiones conjuntas entre DA:**

- El recorrido didáctico que se llevará a cabo para alcanzar los propósitos de la instancia de aula.
- Las actividades de aula para fortalecer el trabajo que se realizará en la VC.
- Dudas o aportes de los niños que hayan surgido y puedan ser relevantes a la hora de trabajar en la VC.

Información para el DR

- La forma en que se aborda el contenido del proyecto en el Aula, para relacionar la actividad realizada con la procedimientos en la VC.

Proyecto de ejemplo de avance para la Etapa 7:

- [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓

Una canción sobre el tema

Notas para el DA ↓

**Propósitos mínimos**

- Brindar el espacio para la escritura de la letra de una canción.
- Favorecer la identificación de la parte que se repite en la letra y formas de nombrar en la escritura la reutilización de una estrofa.

Propósitos óptimos

- Articular con música la canción que van a utilizar.

Una canción con estribillo

En esta actividad el DA puede seleccionar una canción relacionada con el tema elegido para el proyecto u otra que considere pertinente. Esta es una actividad que pueden realizar en conjunto con Música.

En el aula se reproduce la canción para que los estudiantes escuchen la letra. En un segundo momento se les pide que transcriban la letra de la canción. El DA hace pausas en la reproducción para brindarle a los estudiantes el tiempo prudente para que escriban la letra.

Finalizada la escritura se incursiona para indagar si algún estudiante identificó la parte que se repetía y si la nombró de alguna manera en especial (Estribillo, estrofa X).

¿Hay alguna parte de la canción que se repite? ¿Cuántas veces? ¿Escribieron todas las veces esa parte? ¿Cómo la identificaron? ¿Cómo se llaman en música las estrofas que se repiten?

Pueden reescribir la letra de la canción pero ahora el estribillo se escribe una sola vez y luego en vez de repetirlo completamente, sólo se escribe la palabra *estribillo*.

Variamos la canción

Con las mismas estrofas se les pide a los estudiantes que armen una nueva canción. El DA puede orientar a identificar todas las estrofas con un nombre y luego armar la secuencia. Ejemplo

Estrofa 2
Estribillo
Estrofa 1
Estrofa 3
Estribillo

VC ↓ Procedimientos

⌚ Actividades previstas para requerir más de una VC de 45 minutos

💡 Desafío

Mostrar un mensaje de acierto cuando el Explorador llegue a un objetivo que coincida con el enunciado.

1. Inicio (5 min)

¿Qué canción escribieron? ¿Tiene una estrofa que se repita? ¿Con qué nombre la identificaron? ¿Inventaron una nueva canción? ¿Tuvieron que escribir la letra de nuevo?

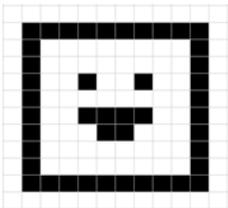
Se retoma de la actividad realizada en aula la solución propuesta por los estudiantes para no escribir más de una vez la parte que se repite de una canción.

El DR cuenta que en programación hay una forma de identificar un conjunto de instrucciones que realizan una acción en particular para utilizarla en distintos lugares del programa: los **procedimientos**. Para apropiarse de la nueva herramienta de programación se proponen dos opciones a continuación. Cada DR, decide cual implementará.

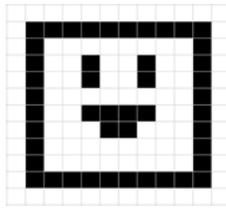
2A. Desarrollo - Procedimiento(35 min)

Emojis en hojas cuadriculadas(opción 1)

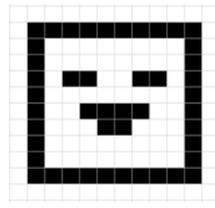
En esta actividad te planteamos el desafío de escribir un programa para que alguien que nunca vio como son Pi, Xe, y Les pueda dibujarlos siguiendo las instrucciones.



PI



XE



LES

La dinámica de esta actividad desenchufada se encuentra detallada en el [Anexo 5](#).

Una vez finalizado, se analizan los grupos de instrucciones que escribieron: ¿Qué grupo de instrucciones pueden aprovechar para dibujar los otros dos emojis? ¿Cómo podríamos expresar cada grupo de instrucciones para no tener que escribirlas nuevamente? ¿Nos sirve la estrategia de solución pensada para representar esos grupos?

Se relaciona la solución con la forma que utilizaron en el aula para identificar el estribillo de la canción. El DR, indica a los estudiantes que la herramienta del lenguaje básica para expresar subtareas y estrategias en programación es el **procedimiento**. En esta actividad se utilizan los **procedimientos de dos formas**: primero para dividir un problema en varios más sencillos; donde cada uno se resuelve con un procedimiento específico y segundo para reutilizar un grupo de instrucciones que se repiten.

EL DR guía a los estudiantes para que identifiquen la importancia de pensar primero la estrategia de solución, dividir el problema en subtareas y de esta forma escribir **procedimientos** que representan una acción, por ejemplo: *Pintar los cuadraditos de la cara*.

Los **procedimientos** se definen con un nombre y una secuencia de instrucciones. Es importante que el nombre de un procedimiento describa apropiadamente la tarea que cumple. **Para el ejemplo, un nombre descriptivo del procedimiento sería *Pintar los cuadraditos de la cara*, caso contrario sería el nombre *Pintar un cuadrado* pues no representa la acción que realiza el procedimiento en nuestro problema**. Una vez definidos, se puede recurrir a los procedimientos desde cualquier parte del programa usando su nombre. Cada vez que se llama a un procedimiento se ejecutan las instrucciones que tiene asignadas, de esta forma nos permite reutilizar un conjunto de instrucciones que se repiten.

Procedimientos en Pilas bloques(opción 2)

Se propone realizar tres desafíos en Pilas bloques ([Anexo 6](#))

En estos desafíos, los estudiantes, con guía del DR, primero tienen que explicitar la estrategia de solución y se valora como una parte fundamental de la solución. Luego representar esa división mediante tareas, que en la herramienta se representan al crear procedimientos, por último incorporar las primitivas a los procedimientos definidos.

★ Importante

Permitir que se expliciten distintas propuestas y entre todos analizar la conveniencia o no de utilizar una u otra estrategia de solución.

En el [Anexo 6](#) se propone una dinámica de resolución para el primer desafío, de la misma forma se deben trabajar los dos restantes.

2B. Desarrollo - Implementación (35 min)

Se recupera el **Plan de Desarrollo** para identificar cuál es la próxima tarea. Se espera que en el desarrollo se cuente con un punto que contemple la programación necesaria para resolver el desafío de esta etapa. En caso contrario, el DR, orienta a los estudiantes para que incorporen la acción requerida.

Parte 1: Marcar los lugares a donde se debe llegar

¿Cuándo se gana el juego? ¿Cómo se detecta que el Explorador llegó al lugar que indica el enunciado? ¿Qué componente debemos incorporar?

Los estudiantes recuerdan la forma de identificar en el juego un objetivo en la Etapa 5, se incorpora de la **Paleta Dibujo y animación** los componentes *Pelota* necesarios para marcar cada uno de los lugares donde debe llegar el Explorador que contiene la **Lista**. Se revisan sus propiedades y se realizan los ajustes necesarios.

Parte 2: Programar para que se muestre un mensaje de éxito cuando el jugador llegue a un objetivo correcto.

En esta parte se busca motivar que los estudiantes reconozcan la necesidad de modificar la alternativa condicional que ya hicieron en la Etapa 5, a partir de resolver el problema: El *Explorador* debe llegar a X lugar que indica el enunciado para ganar.

Se puede escribir de la forma:

Si el Explorador llega a un objetivo Y coincide con el objetivo del enunciado entonces gana

Se destaca que es posible construir una condición más compleja a partir de otras más simples y da la pista de revisar la categoría de bloques Integrados ⇒ Lógica. Se espera que los estudiantes recuerden los bloques lógicos $< > y <$ $> y < > o < >$. Se puede reforzar su significado con ejemplos de la vida cotidiana. Como es la primera vez que arman una estructura compleja en este entorno, puede ser necesaria una guía más intensa para combinar adecuadamente los bloques.

Al completar la programación de todos los objetivos se espera que los estudiantes reconozcan la necesidad de utilizar **procedimientos** para agrupar las instrucciones cuando se *gana*. Se define el procedimiento *Ganar partida* y se lo llama cada cada vez que el componente explorador colisiona con un objetivo que coincide con el enunciado

★ Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (15 min)

¿En qué situaciones de su vida cotidiana piensan estrategias de solución?

En muchas tareas que realizamos a diario pensamos en estrategias de solución. Por ejemplo, para una evaluación seguramente organicen las forma que van a estudiar, ahí establecen una estrategia de solución para estudiar y aprobar la evaluación. La estrategia pensada por un estudiante puede coincidir o no con la establecida por otros compañeros. Una posibles estrategia de solución puede ser:

- 1- Realizar un resumen
- 2- Estudiar el resumen
- 3- Responder algunas preguntas/realizar actividades

Cada uno de esos puntos representa una tarea que se realiza para cumplir con el objetivo. Ante un problema(matemático, de la vida, u otros) se piensa de qué manera se puede resolver. En ese momento, están pensando en una posible estrategia de solución, pero en general, no se explicita. Las estrategias de solución se pueden expresar a través de considerar pequeños sub-problemas o tareas cuyas soluciones combinadas provean la solución al problema general. En programación para explicitar la estrategia de solución usamos los procedimientos.

¿Se imaginan cómo sería el programa para dibujar a los emojis si no tuvieran definidos procedimientos? ¿Cómo harían para modificar una palabra del estribillo de una canción si la estrofa está escrita 5 veces en lugares diferentes? ¿Se simplificará esta tarea si la estrofa estaría identificada con la palabra Estribillo y estuviera escrita una sola vez?

(El DR para representar la situación puede tener escrito el programa de los emojis sin procedimientos y lo mismo para una canción con la estrofa del estribillo escrita varias veces)

Definir procedimientos nos permite agrupar instrucciones que realizan una tarea específica y el nombre que se le asigne debe comunicar de manera legible su propósito a quién lo lea. Además, son útiles para no repetir una secuencia de instrucciones idénticas en los programas. Bien utilizados mejoran notablemente la legibilidad de los programas. Por otro lado, si se debe modificar una parte de un programa sólo se realizan cambios en un solo lugar.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase alternativas condicionales



La Yapa: Propuestas para seguir en casa

¡Y conseguiste que te juego llegue a distintos objetivos!. Puedes hacer que inicie una nueva partida sacudiendo el dispositivo. Puedes mirar el proyecto de la Etapa 2 para ver que sensor necesitas. ¿A qué otro objeto va a ser igual la programación? ¿Puedes definir un procedimiento?



ETAPA 8 - Opcional↓ El Enemigo

En esta etapa se propone programar el movimiento del personaje Enemigo. En el aula se proporciona un espacio para reconocer el cambio en la mecánica del juego al incorporar un enemigo.

Durante la VC, los estudiantes programan el comportamiento del Enemigo para que se desplace por la pantalla siguiendo al Explorador. También se puede incorporar vidas y/o puntaje utilizando variables.

Por ejemplo, cada vez que el Enemigo alcanza al Explorador, pierde una vida.

Al terminar esta etapa se tiene una versión completa del juego donde se puede ganar, perder vidas y sumar puntaje.

Objetivos

Se espera que los estudiantes:

- Armen expresiones aritméticas que relacionen el movimiento del Explorador y el desplazamiento esperado del Enemigo.
- Formulen reglas de juego con estructura de alternativa condicional para perder el juego.
- Reconozcan a la computadora como una máquina que procesa información de entrada y produce un resultado en consecuencia.

Coordinación dupla pedagógica

Decisiones del DA

- El recorrido didáctico que se llevará a cabo para alcanzar los propósitos de la instancia de aula.
- La forma de modelizar y llevar a cabo una instancia del juego en el aula.

Decisiones conjuntas entre DA y DR:

- Acordar si se va a realizar la VC opcional de diversificación de proyectiles y cómo esto impacta en las actividades de aula.
- Evaluar las consignas planteadas por los estudiantes y proponer una que funcione para incorporar el enemigo en el proyecto.

Información que necesita tener el DR:

- Las actividades que se desarrollaron en el aula para retomarlas y establecer asociaciones con el proyecto.

Proyecto de ejemplo de avance para la Etapa 8:

- [Enlace a la galería](#) o [Enlace a la descarga desde drive](#)

AULA ↓ Contenido del proyecto

Notas para el DA ↓



Propósitos mínimos

- Favorecer, desde la experimentación, la comprensión del contenido curricular elegido y el funcionamiento del juego a programar incorporando un enemigo.

Propósitos óptimos

- Articular la producción de los elementos visuales del juego con el Área de Conocimiento Artístico.

Repasar el contenido curricular.

Se sugiere trabajar con el contenido propuesto para el proyecto analizando quién puede ser un enemigo en el juego. Por ejemplo: si se trabaja con la célula, el enemigo puede ser un virus.

Para modelizar esta nueva situación que ocurrirá en la aplicación y poner en juego los conocimientos necesarios, se puede retomar la dinámica implementada en las etapas anteriores, pero ahora incorporando a uno de los estudiantes como enemigo.

Por ejemplo: la imagen dibujada puede ser el mapa de Uruguay, la maestra nombre un departamento y el estudiante explorador debe llegar a ese sitio antes que el enemigo lo atrape.

Los estudiantes dibujan o descargan la imagen para usar en la aplicación como enemigo y las guardan en una carpeta del proyecto. En la [bitácora](#) se dispone de un espacio para incorporar al enemigo y para que describan brevemente la función. Este insumo servirá para completar el **Plan de Desarrollo y el proyecto en la VC siguiente.**

VC 8 - A ↓ Opcional Movimiento del Enemigo

1. Inicio (15 min)

¿Cuándo se pierde en el juego? ¿En el plan de desarrollo contemplamos esta opción? ¿Qué componente necesitamos agregar a la interfaz para poder incorporar un enemigo?

Se retoma lo trabajado en el aula para completar la **interfaz** de la aplicación y el **Plan de desarrollo** donde se describen las acciones asociadas para programar el enemigo. Luego, deben agregar a sus proyectos un componente de la **Paleta Dibujo y animación** del tipo *SpriteImagen* del mismo modo que hicieron con el Explorador. Recordar que estos elementos se incorporan dentro del *Lienzo* **pues es la zona donde funciona el juego**. Los estudiantes deben modificar las propiedades X e Y para establecer dónde quieren que se muestre inicialmente el enemigo. Otra propiedad importante que deben definir es la *Velocidad* de desplazamiento que tendrá.

2. Desarrollo (20 min)

Parte 1: El enemigo sigue al Explorador

¿En qué dirección debe moverse el enemigo? ¿Qué información necesita para desplazarse?

Los estudiantes revisan sus proyectos e identifican el **evento** asociado al sensor que se programó para mover al Explorador. El DR alienta a los estudiantes a revisar los bloques que contiene el componente Enemigo para que identifiquen el bloque *ApuntarEnLaDirección*, este bloque se asocia a las coordenadas del Explorador, de esta forma el Enemigo se mueve siguiendo la ubicación de este.

Parte 2: Se pierde el juego

Desafío

El Enemigo alcanza al Explorador y el juego muestra un mensaje “perdiste”.

En esta parte se busca motivar para que los estudiantes reconozcan la necesidad de usar la alternativa condicional, a partir del problema: El *Enemigo* persigue al *Explorador* ¿Qué ocurre cuando lo alcanza?.

Brindar a los estudiantes un espacio de reflexión con el objetivo de que identifiquen la necesidad de usar nuevamente alternativa condicional para representar la situación:

Si el Enemigo Explorador alcanza al Explorador **entonces** pierde

Se retoma el uso de alternativa condicional en app Inventor ya que los estudiantes estarán familiarizados con el uso realizado en la Etapa 5. En este caso se utiliza en mismo bloque *EnColisiónCon* pero esta vez del Enemigo . El DR, invita a los estudiantes a que revisen en su proyecto el uso de alternativa condicional y de esta forma, armen nuevamente la estructura pero para que cumpla esta nueva condición. Se brinda el tiempo necesario para que los estudiantes intenten armar solos la alternativa condicional y el DR sólo interviene con preguntas para destrabar situaciones que presenten dificultad. Para mostrar el mensaje “Perdiste”, al igual que cuando se gana, se utiliza el mismo componente no visible *Notificador* ya incorporado en la Etapa 5 . El componente permite emitir avisos durante la ejecución de una aplicación.

Importante

Antes del cierre considerar las recomendaciones respecto a la importancia de compartir los avances del proyecto en el **Foro de evidencias** y realizar las **Actividades interactivas**.

3. Cierre (10 min)

La computadora como máquina que procesa información de entrada y produce información de salida. Se propone el mensaje de acierto y fallo como **salida de información**.

¿Cuándo entra información al dispositivo al jugar en la aplicación? ¿Qué sucede después? ¿Se les ocurren otras situaciones en las que suceda esta secuencia al usar una computadora?

Se recupera la noción de **evento como entrada de información** para establecer analogías e identificar un flujo de información en ambos sentidos entre usuario y máquina. También se reconstruye la secuencia en que el usuario inclina el dispositivo para mover al **Explorador**, la computadora lo desplaza en la pantalla y finalmente muestra el acierto cuando llega al objetivo correcto. A partir de estas interacciones, se construye entre todos una caracterización de **computadora como una máquina capaz de recibir información, procesarla y generar información de salida**. Los estudiantes proponen otras situaciones en las que identifiquen ingreso, procesamiento y salida de información.

El programa como responsable de la relación entre la información de entrada y la de salida.

¿La salida es siempre la misma? ¿De qué depende?

Los estudiantes analizan las situaciones que propusieron e identifican que la salida depende tanto de información de entrada (inclinación del dispositivo) como de las reglas escritas en el programa para mostrar cuando se gana o pierde. A partir de esta observación se completa el esquema agregando el programa en el interior de la computadora como un componente que, si bien no es tangible es fundamental para determinar qué hace y para qué podemos usar una computadora. Los estudiantes agregan situaciones en las que identifiquen ingreso, egreso y procesamiento de información con un programa. Por ejemplo

el ingreso en CREA con una contraseña, en el que, además, se puede identificar una alternativa condicional:

- Información de entrada: usuario y contraseña.
- Procesamiento: decidir si el usuario y contraseña que ingresó son correctos; SI es correcto, entonces ingresar al sitio de CREA; SI es incorrecto, entonces, mostrar un mensaje indicando que el usuario o la contraseña son incorrectos.
- Información de salida: pantalla con las aulas de CREA en el caso de que la contraseña sea correcta, mensaje de error en caso de que sea incorrecto el acceso.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa en las que se repase alternativas condicionales.



La Yapa: Propuestas para seguir en casa

¡Felicidades! Lograste tener una nueva versión de tu juego. Ahora, puedes incorporar *una imagen y sonido para cuando el juego se pierde, debe estar relacionada con la temática del juego o puede ser el mismo Explorador en distintas posiciones*. *¿Qué componente debes agregar? ¿En qué parte incorporas la programación de este nuevo componente?*



VC 8 -B ↓ Opcional Vidas y puntaje

Desafío

Tienen tres intentos para alcanzar el objetivo y cada vez que llegan a un objetivo suman 10 puntos.

1. Inicio (10 min)

¿Qué debemos incorporar para poder tener vidas y puntaje? ¿En el plan de desarrollo contemplamos esta opción? ¿Qué componentes se tendrán que incorporar en la interfaz?.

Se retoma lo trabajado en el aula para completar la **interfaz** de la aplicación y el **Plan de desarrollo** donde se describen las acciones asociadas para programar vidas y puntaje.

Los estudiantes deben agregar a la interfaz de sus proyectos componentes de la **Paleta Interfaz** del tipo *Etiqueta*. Recordar que estos elementos se incorporan dentro de un componente **Disposición** para realizar una distribución porcentual adecuada. Modifican las propiedades para que muestren en la interfaz de la aplicación las vidas al inicio y el puntaje en cero.

Se exploran los proyectos utilizados en la etapa 1 y 2, para que los estudiantes puedan ver qué elementos de esos proyectos pueden reutilizar.

2. Desarrollo (25 min)

Desafío 1: Cada vez que el enemigo colisiona con el elemento explorador se debe restar una vida.

¿Qué herramienta de programación nos permite que un programa recuerde un valor?

El DR muestra en su pantalla la categoría Variables y señala que allí se encuentran los bloques que permiten crear, modificar y conocer el valor de una variable. El estudiante debe incorporar una variable vida y programar el evento colisión del enemigo, para que reste vidas.



Para determinar si perdió se necesita el uso de alternativas condicionales y el operador lógico Y. Es importante que primero se enuncie la condición para luego intentar armar la estructura con bloques. Los estudiantes deben tomar como referencia las alternativas complejas armadas en la Etapa 7 para poder armar la nueva estructura completa.

Ejemplo de enunciado:

Si <El enemigo choca con Explorador **Y** la Variable Vida es menor igual a 0> **entonces** Pierde

Sino Restar una vida

Se espera que los estudiantes identifiquen la acción **perder vidas** y definan un **procedimiento** que agrupe las instrucciones que realizan la resta de la variable vidas. También puede surgir la necesidad de definir un **procedimiento Detener partida** para agrupar acciones que se repiten cuando se gana y pierde.

Desafío 2: Cada vez que el Explorador llega a un objetivo, suma puntaje.

Para programar esta acción los estudiantes deben identificar en qué evento deben incorporar el bloque, y luego armar una estructura similar que para las vidas.



3. Cierre (10 min)

Los programas recuerdan información durante su ejecución: ¿Qué programas o plataformas conocen donde se almacena información y qué se almacena?

Si es necesario el DR ofrece algunos ejemplos:

- Otros juegos: cantidad de vidas disponibles, número de nivel alcanzado, nombre del jugador, equipo del jugador, datos de configuración
- Editores de texto: se recuerda el texto escrito hasta el momento y si estamos usando algún formato particular, como negrita o color.
- Aplicaciones de dibujo: recuerda el tipo de lápiz seleccionado para dibujar.

Registro en Crea

El DR publica el registro con las notas y reflexiones de los intercambios en el **Registro Común**. Pueden incorporarse capturas de pantalla de las actividades de los estudiantes.

Invitar a los estudiantes a resolver las actividades interactivas de la etapa.



La Yapa: Propuestas para seguir en casa

¡Felicidades! tu juego está muy completo. Para continuar mejorándolo puedes agregar que muestre distintas imágenes y reproduzca distintos *sonidos para indicar cuando pierde una vida y cuando suma puntos. También puedes agregar un botón para reiniciar el juego. ¿Qué componentes debes agregar? ¿En qué parte incorporas la programación de estos nuevos componentes?*



<p>ETAPA 9 ↓ Cierre del proyecto</p>	<p>Coordinación dupla pedagógica</p>
<p>En esta última etapa, se planifica la difusión del proyecto en la comunidad.</p> <p>En la VC, se realizan los últimos ajustes y la metacognición del trayecto de PC.</p> <p>Se realiza en Crea, la actividad de evaluación final del proyecto.</p> <p>Objetivos</p> <hr/> <p>Se espera que los estudiantes sean capaces de:</p> <ul style="list-style-type: none"> ● Reconocer y reflexionar sobre su propio proceso de aprendizaje durante todo el proyecto. ● Compartir sus producciones con sus compañeros, familias y la comunidad educativa. 	<p>Decisiones conjuntas entre DA y DR:</p> <ul style="list-style-type: none"> ● El DA fue quien presenció todo el trabajo de los estudiantes, acompañó y recibió sus inquietudes más urgentes y pudo observar directamente el desarrollo de cada grupo. Por lo tanto, cuenta con el insumo fundamental para el ejercicio de metacognición propuesto. Su rol en esta actividad es evocar los aspectos de la experiencia que se vuelven relevantes con las sucesivas preguntas del DR y que tal vez los niños no identifican o valoran a priori. También, puede transmitirle al DR situaciones que haya notado como particularmente significativas para los estudiantes para que el DR las considere al momento de hacer la reflexión final. <p>Decisiones del DA</p> <ul style="list-style-type: none"> ● Forma en que compartirán las aplicaciones con las familias y comunidad educativa.

AULA ↓

Cierre del tema del proyecto

Notas para el DA ↓



Propósitos mínimos

- Propiciar una instancia de cierre del proyecto de aula

Propósitos óptimos

- Planificar junto a los estudiantes la forma en que compartirán las aplicaciones creadas con la comunidad educativa.

Aplicación terminada

Planificar el día o la forma en que se compartirán las aplicaciones creadas, con los compañeros, las familias u otros estudiantes de la escuela. Esta etapa de cierre, es una oportunidad para afianzar el tema que se desee destacar, evaluar y dialogar sobre los aprendizajes curriculares que les dejó el proyecto.

VC 9↓

Ajustes finales y metacognición

1. Ajustes finales y publicación de los proyectos (15 min)

Los grupos resuelven y ajustan sus proyectos. A medida que van finalizando, los suben a CREA.

2. Metacognición del proyecto (20 min)

A partir de la dinámica de cierre que los docentes acuerden, el propósito de la metacognición es que los estudiantes reflexionen sobre los siguientes aspectos:

Herramientas de programación aprendidas.

La herramienta de programación es nueva por lo que todos los bloques son relevantes en este proyecto. Pero particularmente se pueden considerar las dos vistas de App inventor y la función particular de cada una. Componentes de entrada y salida de la aplicación. Listas y Procedimientos y su asociación tanto con su uso en el proyecto como en otras situaciones cotidianas discutidas en las reflexiones de las VCs.

Los programas como producción humana.

¿Son conscientes de que crearon un programa de computadora?
¿Cómo se crearán los otros programas que usamos cotidianamente?
La aplicación como un programa construido por los estudiantes para jugar y fortalecer un determinado tema y generalización de la tarea de programación: todos los programas que usan en sus dispositivos (para comunicarse, entretenimiento, dibujar, navegar por internet, etc) fueron construidos por personas, como un conjunto de instrucciones para la computadora. Esta observación es una oportunidad para que

los estudiantes tomen conciencia de lo poderoso de la tarea de aprender a programar.

Estrategia de solución

¿Recuerdan en qué situaciones fue indispensable detenernos a pensar una forma de resolver el desafío presentado? ¿Por qué creen que lo trabajamos de esa manera? ¿Conviene detenernos a pensar cómo resolver los desafíos o ir avanzando por prueba y error? Se sugiere recuperar el Plan de desarrollo de la aplicación y las estrategias de solución que fueron utilizando etapa a etapa para resolver desarrollar cada punto. Importante destacar la reutilización de código que se hicieron de otros proyectos y el desarrollo incremental de la aplicación.

Proceso de aprendizaje

¿Qué cosas de las que hicimos no conocían? (qué aprendimos)
¿Cómo relatarían a otra persona el camino recorrido durante este proyecto? (cómo lo aprendimos), ¿qué lograron construir con sus nuevos conocimientos? (para qué les sirvió), ¿en qué otras situaciones creen que pueden utilizar lo aprendido? (generalización).

Aspectos emocionales

¿Distinguen alguna actitud propia o de los compañeros que los ayudó a avanzar con el proyecto? ¿Se escucharon todas las voces y se buscó que se reflejan en el producto final? ¿Se tuvieron en cuenta las ideas y fortalezas de cada miembro del equipo? ¿Cómo enfrentaron el error? ¿Lo tomaron como una oportunidad de aprendizaje o los

frustró? ¿Por qué? ¿Recuerdan cómo se sintieron frente al desafío al iniciar este proyecto? ¿Y cómo se sienten ahora con lo logrado?

3. Cierre (10 min)

El DR invita a los estudiantes a resolver la evaluación del proyecto.

Registro en Crea

Los estudiantes comparten sus proyectos finales en CREA.



La Yapa: Propuestas para seguir en casa

¡Excelente! ¡Ya tienes tu primera aplicación completa funcionando! Puedes seguir trabajando para agregarle más cosas. Tiempo para llegar hasta el objetivo. Cambio de velocidad en el Explorador. También pueden comenzar a pensar en tu próxima aplicación, ¿qué te gustaría programar?

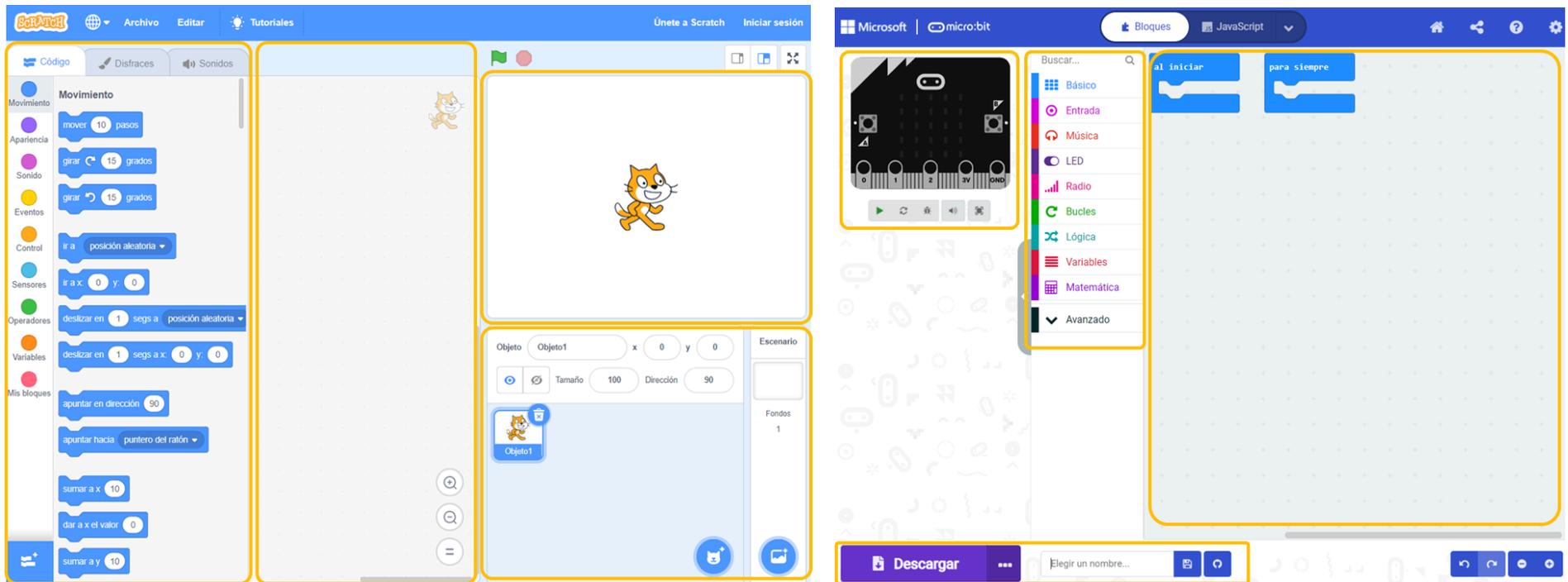


¡Ahora te toca a ti!

ANEXO 1

Imagen guía para reconocer similitudes y diferencia entre entornos de programación

¿Cómo aparecen en App Inventor cada una de estas partes? ¿Aparecen?



[Volver a Etapa 1](#)

ANEXO 2

Ejemplos para la selección de imágenes

Explora la Célula Animal

Vidas: 3

Ubica donde se encuentra: **la Mitocondria**

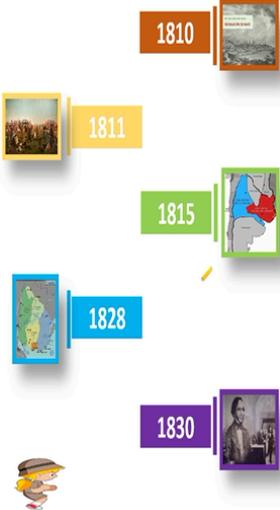


Iniciar

Explora la historia de Uruguay

Vidas: 3

En qué año: Artigas derrota a los realistas en la Batalla de Las Piedras



Iniciar

Explora Uruguay

Vidas: 3

Lleva a la exploradora al departamento: **Soriano**

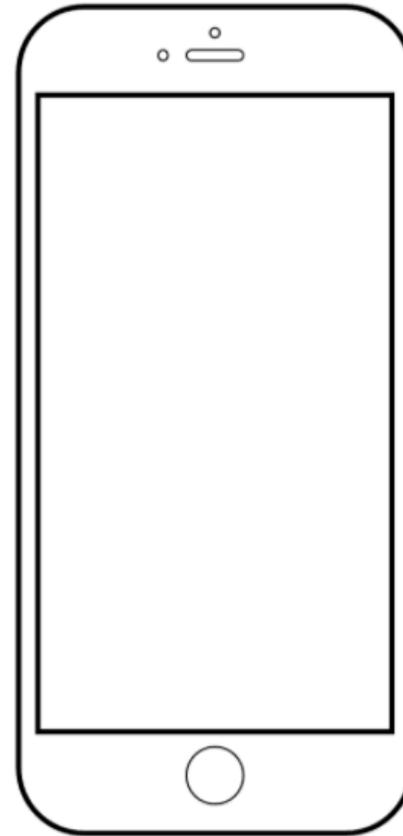


Iniciar

[Volver a Etapa 3](#)

ANEXO 3

Plantilla para el diseño de la interfaz



[Volver a Etapa 3](#)

 **ANEXO 4****Plan de Desarrollo de ejemplo****Plan de desarrollo de ejemplo:**

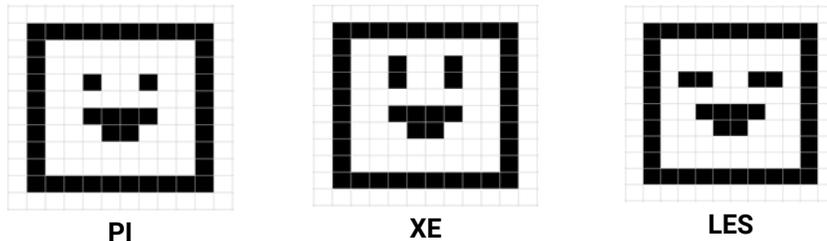
1. Armar la interfaz
2. Mover el Explorador cuando se inclina el dispositivo
3. Hacer que el Explorador llegue a un objetivo y si es el correcto, gane.
4. Incorporar varios objetivos a los que tiene que llegar el Explorador.
5. Programar el botón nueva partida.
6. Programar todos los objetivos a los que el Explorador debe llegar
7. *Lograr que el enemigo persiga al Explorador.(opcional)*
8. *Hacer que el enemigo toque a el Explorador para que pierda(opcional)*
9. *Programar las vidas.(opcional)*
10. *Programar puntaje.(opcional)*

[Volver a Etapa 4](#)

ANEXO 5

PI XE LES

Vamos a escribir un programa para que, alguien que nunca vio como son Pi, Xe, y Les, pueda dibujarlos siguiendo las instrucciones.



Observamos los dibujos: ¿Qué cosas poseen todos los emojis en común? ¿Hay alguna parte de los emoji que se repita? ¿Qué es diferente en los emojis? Los estudiantes observan los emojis e identifican patrones. Luego de unos minutos los estudiantes comparten sus respuestas. El DR escribe una lista con las características similares y diferentes que encuentren.

Pensar cómo se pueden dibujar: ¿Cómo se puede dibujar a PI? ¿Qué pasos hay que seguir? ¿Todos pensaron los mismos pasos?, y para los otros emojis, ¿Sirven los mismos pasos?

El DR anota la propuesta explicitada por los estudiantes. Una posible propuesta puede ser:

- 1- Pintar los cuadraditos de la cara
- 2- Pintar los cuadraditos de la boca
- 3- Pintar los cuadraditos de los ojos

El DR, explica a los estudiantes que cada vez que se quiere solucionar un problema es necesario, primero, contar con una idea de cómo resolverlo. Esto se conoce como **Estrategia de solución**.

En este caso, se está dividiendo el problema en tareas, y por lo tanto explicitando una posible estrategia de solución para dibujar los emojis.

Escribir el programa

Una vez consensuada la estrategia de solución, se presenta a los estudiantes el lenguaje de flechas y se muestran partes de programas para dibujar a PI que deben completar y/o nombrar adecuadamente.



Nombre: _____	Nombre: _____
(↘ →)9	(↑)3
(↘ ↑)9	(→)4
(↘ ←)9	(↘ →)2
(↘ ↓)9	()
	()

Figura a

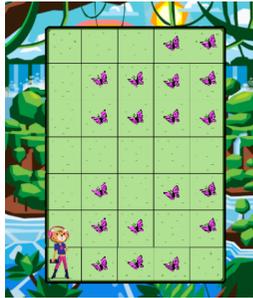
Figura b

Se les brinda el tiempo necesario para que identifiquen que parte de PI dibuja el conjunto de instrucciones de la Figura a. Una vez identificado, se le asigna un nombre representativo a la acción que realiza. Por ejemplo: Pintar los cuadraditos de la cara / Dibujar la cara. Luego se pasa a completar y nombrar el conjunto de instrucciones de la Figura b, que representa la boca. Por último, y teniendo como referencia las figuras anteriores, se les pide que escriban y nombren el conjunto de instrucciones para los ojos.

[Volver a Etapa 7](#)

ANEXO 6

Actividades en Pilas Bloques



[Pilas Bloques - Mariposas](#)



[Pilas Bloques - Barrilete cósmico](#)



[Pilas Bloques - Latas o papeles](#)

Ejemplo resuelto del primer desafío

- Pensar estrategia de solución:**
 - Ir a la parte inferior de la selva misionera
 - Recorrer sacando fotos a las mariposas
 - Ir a la parte superior de la selva misionera
 - Recorrer sacando fotos a las mariposas
- Se representa la estrategia propuesta con procedimientos:**

AL EMPEZAR A EJECUTAR

IR A LA PARTE INFERIOR DE LA SELVA MISIONERA
 RECORRER LA PARTE SUPERIOR SACANDO FOTOS
 IR A LA PARTE SUPERIOR DE LA SELVA MISIONERA
 RECORRER LA PARTE SUPERIOR SACANDO FOTOS

DEFINIR IR A LA PARTE SUPERIOR DE LA SELVA MISIONERA + 📷

DEFINIR RECORRER LA PARTE SUPERIOR SACANDO FOTOS + 📷

DEFINIR IR A LA PARTE INFERIOR DE LA SELVA MISIONERA + 📷

DEFINIR RECORRER LA PARTE INFERIOR SACANDO FOTOS + 📷

Primitivas

Por último se incorporan las primitivas necesarias para cumplir con cada acción que presentan los procedimientos. En este punto los estudiantes pueden identificar que para recorrer cada parte de la selva necesitan las mismas instrucciones, por lo cual se puede dejar un solo procedimiento con un nombre representativo.

Este desafío admite otras estrategias de solución, por lo cual si los estudiantes proponen otra solución es importante mostrar y analizar cada propuesta.

[Volver a Etapa 7](#)