

Pautas para la Interfaz de Usuario

IDE Android para la Programación de Comportamientos Robóticos

Andrés Nebel - Renzo Rozza

Tutor

Andres Aguirre, Gonzalo Tejera

Cotutores

Rafael Sisto

23 de Agosto del 2013

Instituto de Computación
Facultad de Ingeniería - Universidad de la República
Montevideo - Uruguay

1. Objetivo.....	3
2. Interfaz de Usuario: requerimientos.....	4
2.1 Visibilidad	4
2.2 Velocidad de renderizado.....	4
2.3 Eliminación de mensajes de error	4
2.4 Paradigma reactivo: sintaxis	5
2.5 Ejecución Visible	5
3. Interfaz de Usuario: aspecto visual	6
4. Bocetos de pantalla	7
4.1 Bocetos de Tablets y Ordenadores	8
4.2 Bocetos de dispositivos de menor pantalla	12

1. Objetivo

Este documento pretende detallar las características que la interfaz de usuario debe cumplir para lograr una experiencia satisfactoria por parte de los usuarios al hacer uso del sistema. Para esto se considerará la investigación plasmada en el documento Estado del Arte y los requerimientos relevados hasta el momento, detallados en el documento Especificación de Requerimientos.

El objetivo de la interfaz gráfica será otorgar un sistema lo más intuitivo y amigable posible para el usuario, de modo que el mismo pueda familiarizarse fácilmente con las funciones que posee el entorno de desarrollo (IDE), y también desde el punto de vista gráfico atraiga y divierta. Definiremos una visión general de la interfaz de usuario, la cual modelará un lenguaje de programación visual (LPV) basado en bloques, como lo son TortuBots y Scratch, que permita a personas sin conocimiento informáticos lograr códigos de programación simples para controlar la plataforma robótica Butiá.

Se intenta seguir en todo momento determinadas pautas a nivel gráfico: primero, dar el control al usuario, es decir, llevar a cabo un sistema que reaccione a las necesidades del usuario y que brinde facilidades para lograr los cometidos; segundo reducir la carga en la memoria del usuario, siempre que sea posible el sistema debe almacenar la información pertinente; y por último, lograr que la interfaz sea consistente, toda la información visual está organizada de acuerdo con la capacidad de los dispositivos móviles para su diseño haga uso de todas la presentaciones de los mismos.

2. Interfaz de Usuario: requerimientos

El sistema desde el punto de vista gráfico está definido por las tecnologías utilizadas en su desarrollo, siendo en este caso las principales herramientas HTML5, JavaScript, CSS3 y además otras librerías como complemento, mencionadas en el documento Estado del Arte, ya sea Modernizr, YepNope y JQuery Mobile. Estas son de suma importancia para dar soporte a dispositivos móviles, los cuales cuentan con características muy distintas a los ordenadores de escritorio y por lo tanto representa un gran desafío adaptar un LPV basado en bloques a los displays relativamente pequeños y la manipulación por controladores directos (tecnología táctil). Para satisfacer estas necesidades encontramos necesario definir los requerimientos para la interfaz de usuario:

2.1 Visibilidad

La interfaz gráfica del sistema debe contemplar la mayor visibilidad sobre los elementos desplegados en todo momento, dado fluidez para mantener al alcance todas las funcionalidades, haciendo uso de una única ventana.

2.2 Velocidad de renderizado

Siendo niños de corta edad los usuarios de este sistema, es importante tener presente que pueden distraerse si el sistema demora en desplegar los elementos, por lo que el renderizado de la interfaz gráfica debe ser ágil e intentar aprovechar al máximo las limitadas capacidades de procesamiento de los dispositivos móviles.

2.3 Eliminación de mensajes de error

Desde la interfaz gráfica no desplegará mensajes de error. Esto ha sido utilizado muchos LPVs bajo la teoría de que la experimentación y experiencia enseñará a los usuarios que funciona. Además, al ser un LPV basado en bloques los errores de sintaxis no existen, dado que la única forma para que un comportamiento tenga sentido es cuando los bloques encastran.

2.4 Paradigma reactivo: sintaxis

A partir de las experiencias obtenidas con TortuBots, se debe tomar consideración sobre el manejo de variables numéricas para la representación de medidas. Siendo un claro ejemplo la distancia; resulta confuso manejar bloques que expresan la cantidad en metros que el robot Butiá avanzará ya que por lo general no se cumple con exactitud en práctica. Por lo tanto, se debe respetar una sintaxis adecuada para la implementación gráfica de los bloques de acuerdo al paradigma reactivo, manejando bloques que estén mapeados directamente con comportamientos robóticos.

2.5 Ejecución Visible

El sistema contará con la funcionalidad de debugging, y por lo tanto desde la interfaz gráfica se debe dar soporte para lograr que los usuarios entiendan que comportamiento se está ejecutando en determinado momento.

3. Interfaz de Usuario: aspecto visual

Según los requerimientos relevados hasta el momento, el aspecto visual del LPV basado en bloques tendrá a grandes rasgos tres grandes componentes que se desplegarán en pantalla: primero una paleta que contendrá los distintos bloques, los cuales representan tanto comportamientos del robot Butiá como instrucciones lógicas; segundo un workspace donde se podrán encastrar bloques para conformar distintos comportamiento y a su vez un espacio que contenga los comportamientos listos; y tercero, un menú que permita ejecutar, detener y debuggear los comportamientos programados y también opciones para abrir, cerrar y guardar los proyectos.

El sistema a extender para modelar la interfaz de usuario es Waterbear, el cual brinda las condiciones detalladas en el documento Estado del Arte. Este será usado para dar una aproximación al aspecto visual final del sistema.

4. Bocetos de pantalla

Se incluyen en esta sección, bocetos de pantalla que representan los lineamientos de interfaz gráfica a seguir para desarrollar el sistema. Básicamente se pueden separar en 2 grupos: los bocetos para ordenadores de escritorio y tablets que tendrán una disposición de los elementos igual, y otros bocetos para dispositivos móviles de resolución menor como ser celulares. Es de importancia recordar que la importancia de este sistema se centra en la compatibilidad con Tablets, siendo los dispositivos móviles de menor resolución un objetivo opcional de este proyecto.

4.1 Bocetos de Tablets y Ordenadores

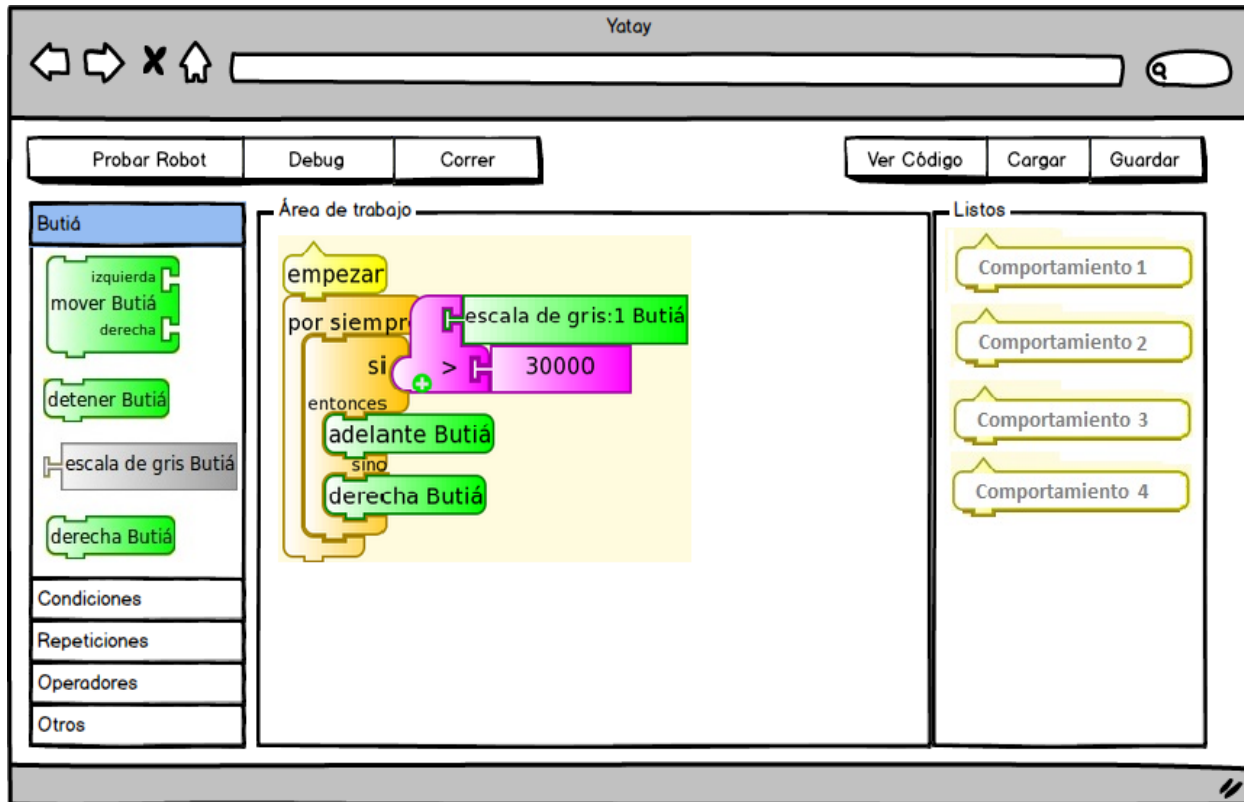


Imagen 1: Pantalla principal, entorno de trabajo. A la izquierda se muestran los bloques que existen en el sistema. Si están en verde están disponibles para agregar, mientras que si están en gris no. En el centro se encuentra el comportamiento que se encuentra actualmente en edición. A la derecha se muestra los comportamientos que han sido creados.

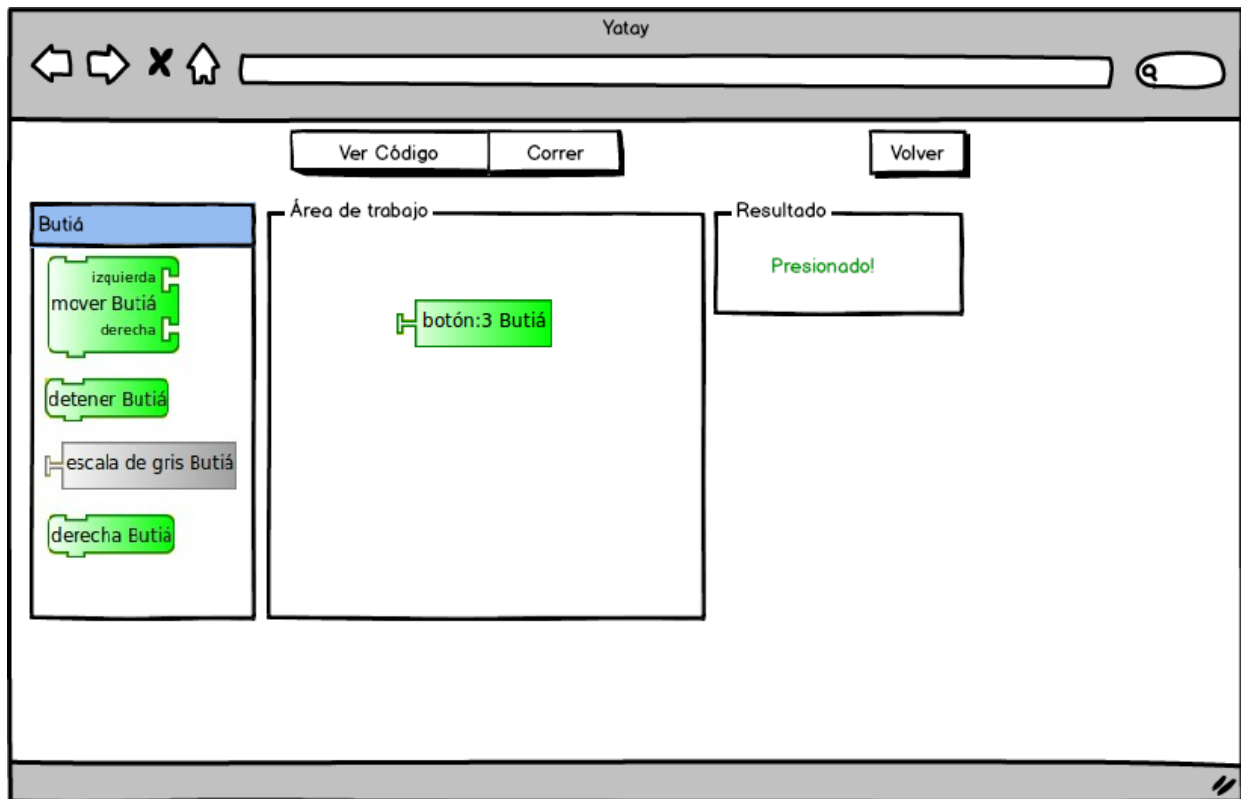


Imagen 2: Pantalla de ejecución rápida. Permite probar los sensores y actuadores del robot sin la necesidad de crear un comportamiento que lo contenga. Es útil a los efectos de la calibración del robot. A la izquierda muestra los sensores y actuadores disponibles (si están en verde) a agregar. En el centro se muestra el sensor/actuador siendo probado. A la derecha se muestra el resultado de la ejecución.

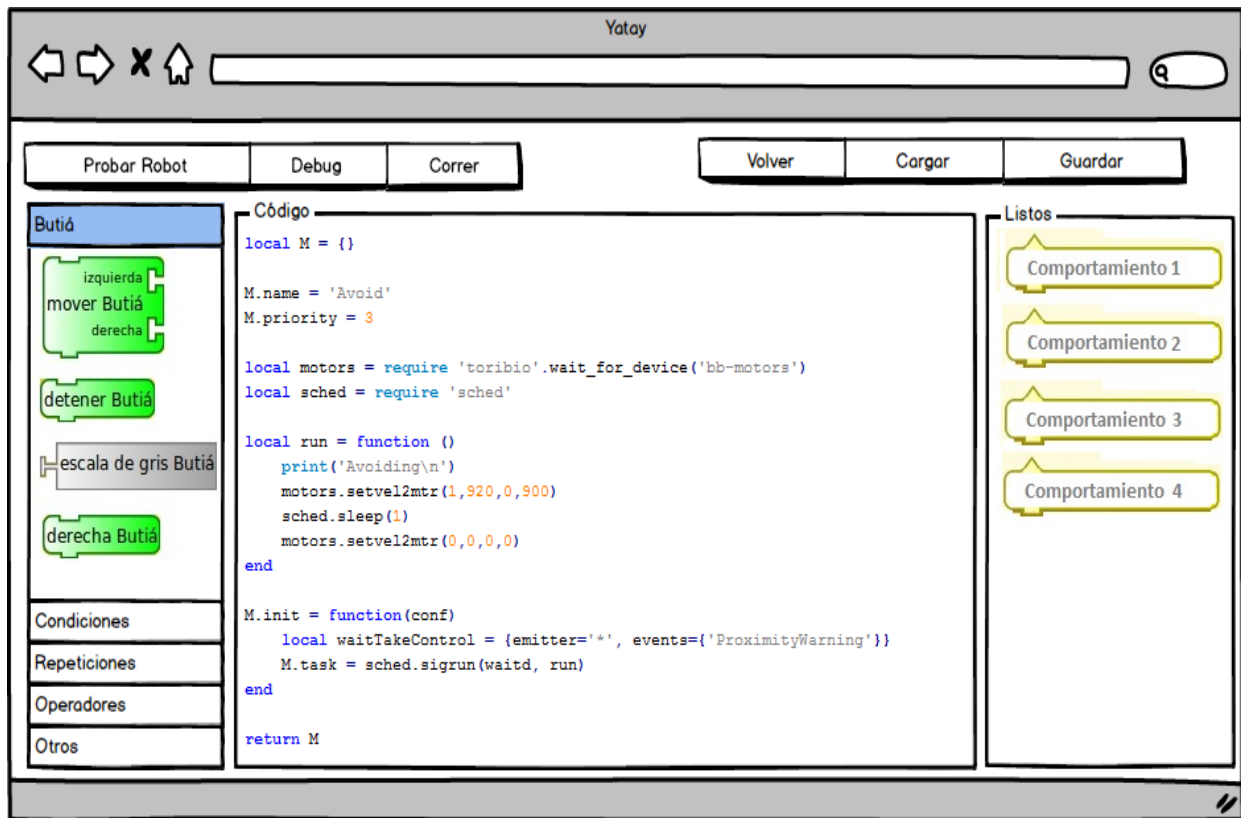


Imagen 3: Vista del código (en este sistema será código lua) generado por el comportamiento en edición. Si se presiona volver se regresa a la imagen 1.

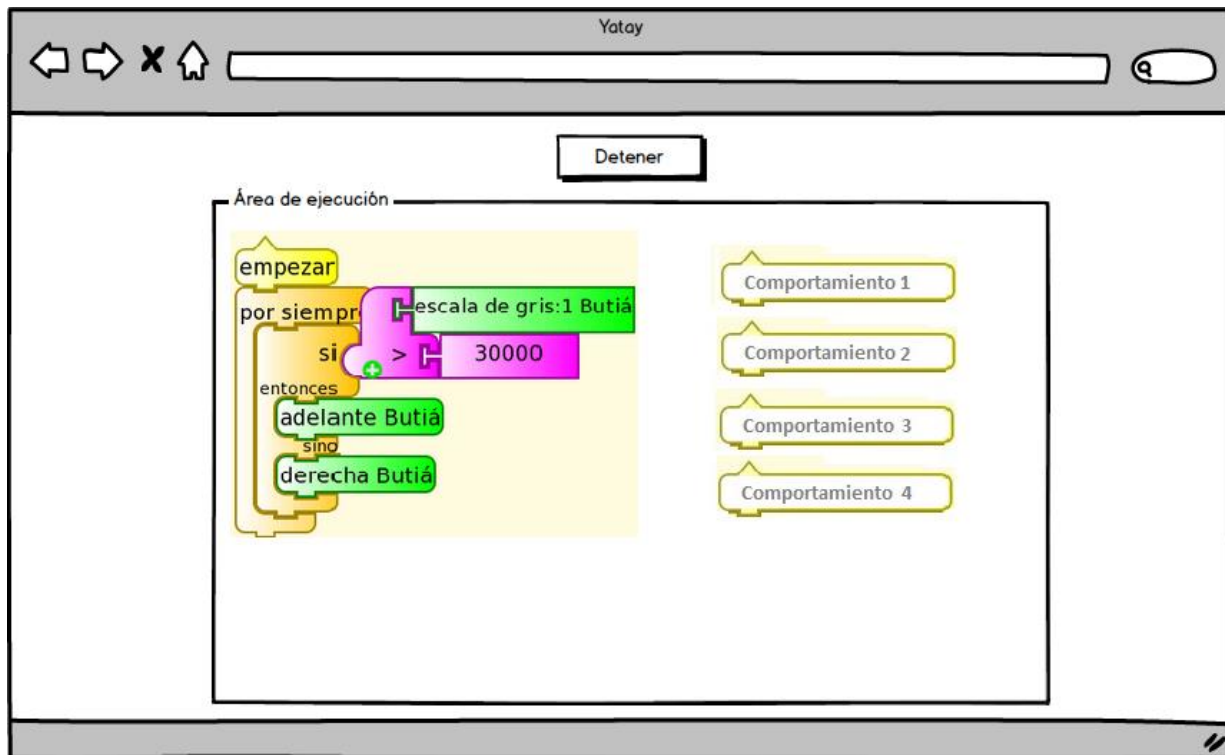


Imagen 4: Vista de ejecución de los comportamientos. Si se encuentra en modo debug se iluminarían los comportamientos que se están ejecutando así como sus bloques.

4.2 Bocetos de dispositivos de menor pantalla

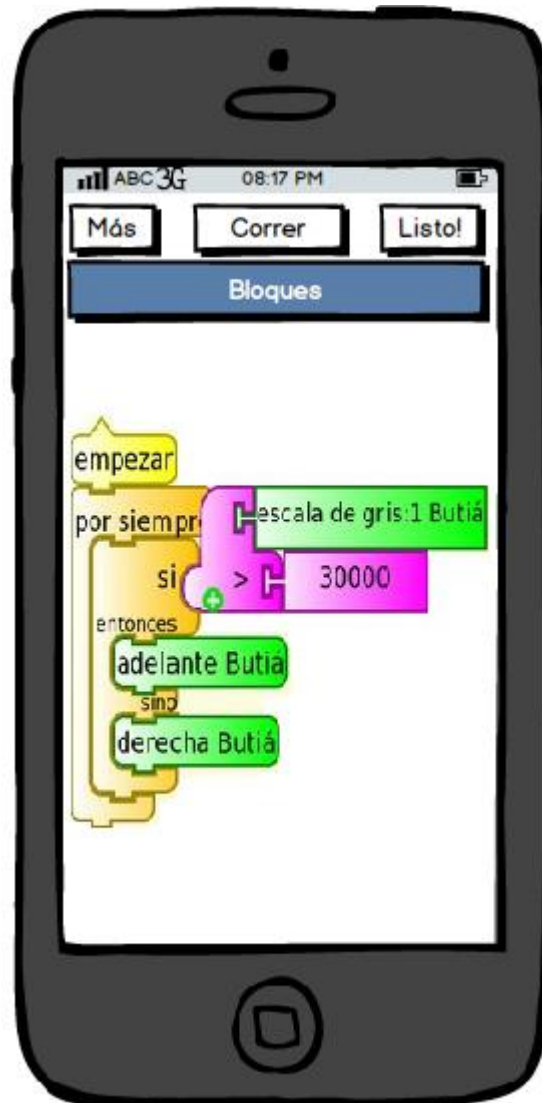


Imagen 1: Pantalla principal, entorno de trabajo. En el centro se encuentra el comportamiento que se encuentra actualmente en edición. Si se presiona bloques se despliega la imagen 2. Si se presiona Listo! se mueve el comportamiento al conjunto de comportamientos listos.



Imagen 2: Pantalla de bloques. Se disponibilizan todos los bloques que pueden ser agregados al comportamiento en edición.

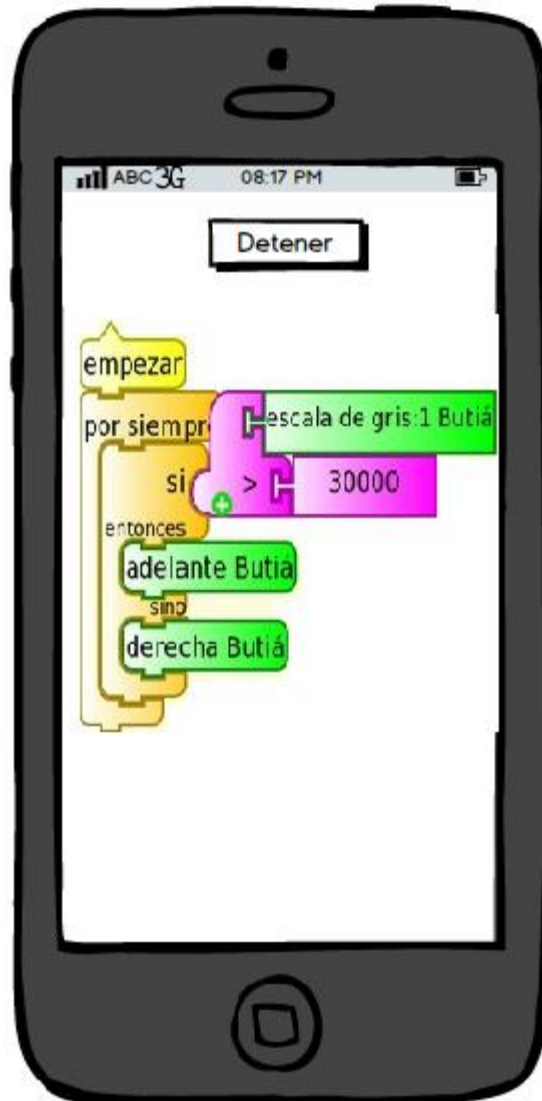


Imagen 3: Pantalla de ejecución, si se encuentra en debug se iluminaran los bloques que están siendo ejecutados.

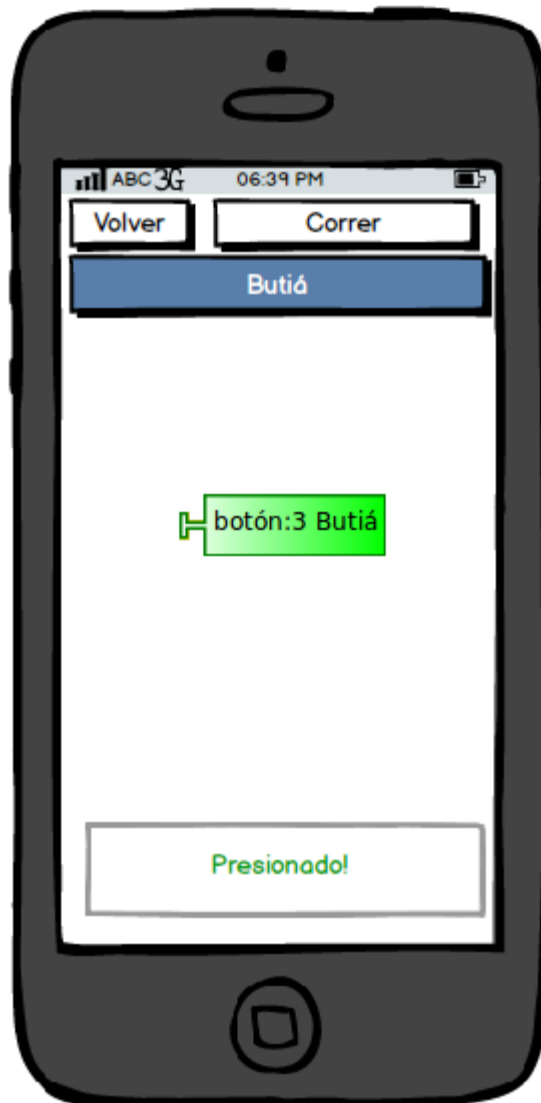


Imagen 4: Pantalla de ejecución rápida. Permite probar los sensores y actuadores del robot sin la necesidad de crear un comportamiento que lo contenga. Es útil a los efectos de la calibración del robot. En el centro muestra el bloque que se ejecutó. Arriba, el menú Butiá permite acceder a los bloques de los sensores y actuadores disponibles para ejecutar. Abajo muestra el resultado de la ejecución del bloque.