

Especificación de Requerimientos

IDE Android para la Programación de Comportamientos Robóticos

Andrés Nebel - Renzo Rozza

Tutor

Andrés Aguirre, Gonzalo Tejera

Cotutores

Rafael Sisto

30 de Julio del 2013

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

1. Introducción.....	3
1.1 Resumen	3
1.2 Propósito.....	3
1.3 Alcance	3
1.4 Visión general.....	4
2. Descripción general	5
2.1 Perspectiva del producto.....	5
2.1.1. Interfaces de usuario.....	5
2.1.2. Interfaces con hardware.....	5
2.1.3. Interfaces con software.....	5
2.1.4. Restricciones de memoria.....	5
2.2 Usuarios.....	6
2.3 Características de los usuarios.....	6
3. Requerimientos funcionales	7
3.1 Entorno y Ejecución	7
3.3 Proyectos	8
3.4 Comportamientos	8
4. Requerimientos no funcionales	9
5. Requerimientos de documentación.....	11

1. Introducción

1.1 Resumen

En este documento se pretende dar una visión global sobre los requerimientos del proyecto propuesto, brindando un acercamiento a la descripción del sistema y detallando el alcance del mismo según lo relevado en el documento de presentación de proyecto y en las reuniones de relevamiento de requerimientos con los tutores del proyecto, además de lo definido en el análisis del estado del arte.

1.2 Propósito

Establecer un acercamiento a los requerimientos del sistema, los que serán confirmados y validados en futuras reuniones a realizar. De esta manera, se logra obtener una primera guía para el diseño de la solución del sistema a construir. También será utilizado como herramienta de apoyo a la validación de la solución.

1.3 Alcance

El proyecto consiste en la realización de un entorno de desarrollo (IDE) para la implementación de comportamientos robóticos que modele un lenguaje de programación visual (LPV). El mismo, debe ser soportado por el sistema operativo Android y permitir a personas de corta edad poder programar tareas que controlen a la plataforma robótica Butiá utilizando una arquitectura perteneciente al paradigma reactivo.

La metodología de desarrollo de los comportamientos debe ser mediante la construcción de figuras (bloques) que representen con simplicidad código de programación robótica. Dicha funcionalidad, deberá estar inspirada en sistemas como TortuBots y Scratch. Dentro de los dispositivos con sistema operativo Android estará principalmente orientado a las Tabletas.

El sistema debe permitir ejecutar los comportamientos implementados y poseer la funcionalidad de debugging, en el sentido de que debe ilustrar al usuario en tiempo real que parte del código está ejecutando el robot Butiá.

1.4 Visión general

A continuación en este documento puede encontrarse una descripción general del sistema, los requerimientos funcionales y no funcionales, así como también los requerimientos de documentación

2. Descripción general

El objetivo general de este sistema radica en la construcción de una nueva herramienta de software para uso escolar, la cual permita a los alumnos complementar su aprendizaje por medio de la programación de comportamientos robóticos. El mismo se modelará en base a las experiencias obtenidas con las actividades integradoras del proyecto Butiá y con el uso de TortuBots.

2.1. Perspectiva del producto

2.1.1. Interfaces de usuario

Ver Documento "Pautas para la interfaz de Usuario".

2.1.2. Interfaces con hardware

Se debe contar con un dispositivo con sistema operativo Android, que posea las características de tamaño y resolución de pantalla que posee una Tablet estándar que asumiremos como 800 x 480 píxeles y entre 7" y 10" de pantalla.

Se dispondrá de una SBC con características similares a la de una Raspberry PI y una controladora USB4Butiá que disponibiliza sus funcionalidades a través de la aplicación Bobot.

2.1.3. Interfaces con software

Se destacan las interfaces de Toribio y Lumen para la administración de la ejecución de los comportamientos además del servidor web, y la utilización de la aplicación Bobot.

2.1.4. Restricciones de memoria

Las Tablets y los dispositivos que corren Android en general suelen ser artefactos de poca memoria RAM así como poca capacidad de procesamiento. Es importante que el sistema a desarrollar contemple dichas restricciones en su implementación.

2.2 Usuarios

Estudiantes:

Los usuarios a los que apunta esta aplicación son estudiantes que en general ya tuvieron sus primeras experiencias de desarrollo con TortuBots, o que han utilizado Scratch, por lo que se puede asumir que conocen al robot Butiá y a la modalidad de construcción basada en bloques. Aunque esto no fuera así, la participación del docente capacitado permitirá al alumno resolver los distintos cuestionamientos que surjan. Los estudiantes suelen desarrollar sus aplicaciones en grupos, pensando en conjunto, desde el armado del robot hasta la programación del comportamiento que resuelve la problemática.

Docentes:

Los estudiantes tendrán como ayuda una guía docente, pero en general la experiencia de los mismos no es mayor a la de los estudiantes. Esto es debido a que también dieron sus primeros pasos tanto en programación, como en robótica, con TortuBots. Los docentes sin embargo corren con la ventaja de haber tenido una capacitación básica por parte de docentes de Robótica que les permite resolver las problemáticas que escapan a la orientación trivial de la aplicación. Ambos tipos de usuarios utilizarán el sistema de la misma forma sin existir funcionalidades, vistas o privilegios distintos.

2.3 Características de los usuarios

Las características expuestas a continuación describen a los usuarios en su mayoría, pero no en su totalidad, ya que el conjunto de los mismos es muy amplio y es probable que existan usuarios finales cuyos conocimientos sean mayores a los descriptos a continuación.

Poseen conocimientos básicos de la programación y de la robótica adquiridos a través de la experiencia con la herramienta TortuBots. Conocen el robot Butiá, así como también, la forma de trabajar con los módulos conectados al mismo; averiguar su propósito, funcionalidades brindadas y el tipo de valores que reciben u ofrecen. No poseen experiencia ni conocimiento en la estructuración de aplicaciones a través de una cierta arquitectura. No son usuarios informáticos avanzados, por lo tanto no tienen conocimientos básicos de comunicación entre aplicaciones ni conocimientos de programación medios o avanzados.

3. Requerimientos funcionales

3.1 Entorno y Ejecución

Entorno de ejecución: Debe ejecutar de forma aceptable en Tablet con sistema operativo Android, siendo opcional el funcionamiento aceptable en dispositivos celulares con sistema operativo Android así como otros dispositivos y/o ordenadores con sistemas operativos distintos. El servidor Web deberá correr en Linux y funcionar de manera aceptable con los recursos de hardware de una SBC Raspberry Pi.

Debugging: El sistema deberá tener una funcionalidad que permita al usuario ver qué bloques implementados están siendo ejecutados en tiempo real, enlenteciendo la ejecución, si fuese necesario, de forma que el usuario pueda apreciar gráficamente la ejecución de un determinado comportamiento implementado.

Múltiples usuarios: El sistema deberá soportar múltiples usuarios. Permitiendo desarrollar a múltiples usuarios distintos comportamientos. Si bien no será un requisito trabajar de manera cooperativa sobre el mismo comportamiento o la misma paleta, deberá permitir que cada usuario desarrolle un comportamiento y que lo envíe para su ejecución al servidor donde se ejecutarán en conjunto según la arquitectura basada en prioridades definida en el documento de Estado del Arte.

Orientación pedagógica: El sistema tendrá una orientación hacia el aprendizaje constructivista y estará basado en las ideas de TortuBots, Scratch y eButia.

3.2 Robot

Paradigma robótico: La aplicación debe permitir la construcción de comportamientos robóticos que estén basados en el paradigma reactivo, siguiendo en dicha construcción y ejecución la arquitectura basada en prioridades definida en el documento de Estado del Arte.

Kit Robótico: El robot al que estará orientado este sistema, es principalmente el robot Butiá definido en el documento de Estado del Arte. Sin embargo, se debe mantener una cierta modularización que permita de manera simple extender el sistema a otros kit robóticos.

Soporte al robot Butiá: El sistema ofrecerá como se menciona en el requerimiento “Kit Robótico” soporte al robot Butiá. Dentro del enfoque de la aplicación a la programación basada en bloques, el sistema debe tener bloques que permitan utilizar las funciones de los sensores y actuadores del robot Butiá que sean ofrecidos por el programa Bobot.

Sensores Genéricos: La aplicación tendrá alguna funcionalidad que permita la utilización de sensores nuevos que sean incluidos en el kit Robótico.

Calibración del Robot: El sistema debe proveer una funcionalidad que permita la calibración de los sensores y actuadores del robot Butiá.

3.3 Proyectos

Gestión de proyectos: En la aplicación se deben presentar herramientas con las funcionalidades de gestión de proyectos: creación, guardado y apertura.

Pantalla de gestión: La ubicación de las funcionalidades de gestión dentro del sistema deberá ser en la pantalla única principal o a lo sumo estar a un acceso de distancia, regla que se mantendrá en general para todas las funcionalidades.

Persistencia de proyectos: Se debe poder almacenar un proyecto con el código generado de cada comportamiento.

3.4 Comportamientos

Nombre y Prioridad: Los comportamientos creados deben estar identificados (no tiene por qué ser internamente en el sistema) por un nombre elegido por el usuario, lo cual permite un entendimiento ágil del comportamiento, además de una prioridad asignada que indicará la importancia de la ejecución del comportamiento frente a otros tal como fue descrito en el documento de Estado del Arte. Tanto el nombre como la prioridad de cualquiera de los comportamientos podrán ser modificados en cualquier momento, a excepción de cuando estos se estén ejecutando.

Pantalla de comportamientos: La metodología de trabajo dentro del sistema consistirá de un espacio para la edición de comportamientos y otro espacio en donde se podrán colocar minimizados los comportamientos ya creados, de forma en que en cualquier momento se puedan seleccionar de manera sencilla para su edición. Esto se definirá gráficamente en el documento de Pautas para la Interfaz de Usuario.

Simplicidad: Las herramientas que ofrece el sistema deben orientar la implementación a comportamientos sencillos que estén basados en el paradigma reactivo.

4. Requerimientos no funcionales

En esta sección se describirán los criterios que permiten juzgar la operabilidad del sistema, los cuales serán de gran importancia a la hora de tomar consideraciones para la arquitectura del sistema.

Tiempo de reacción

Las órdenes ejecutadas en el sistema deben llegar al robot en un tiempo menor a un segundo y medio, de la misma forma el feedback de los comportamientos que llegan al sistema desde la plataforma robótica deben estar por debajo de esa cantidad.

Portabilidad

El sistema debe contemplar portabilidad a la mayor cantidad de dispositivos móviles que cumplan con las restricciones especificadas, haciendo énfasis en los dispositivos Android.

Escalabilidad

El sistema debe permitir agregar nuevas funcionalidades sin encontrar limitaciones desde la codificación, concretamente debe ser posible incorporar nuevos bloques para controlar sensores que se agreguen al robot Butiá, debe ser extensible la lógica que define la arquitectura robótica, entre otras.

Costo

La aplicación debe ser de código abierto, libre, sin costo para su utilización. Todo usuario podrá hacer uso del sistema si cuenta con los componentes físicos requeridos, así como hacer modificaciones al sistema basados en las reglas del software libre.

Mantenibilidad

El sistema debe poder adaptarse a cambios que se generen tanto en la plataforma robótica como en la aplicación extendida para la interfaz gráfica.

Multilinguaje

El sistema debe permitir la configuración del lenguaje desplegado por la interfaz gráfica del mismo, dando la posibilidad en principio de visualizar el texto en español e inglés.

Modularización

El sistema debe mantener un nivel de modularización tal que le permita con facilidad realizar los puntos de Mantenibilidad y Escalabilidad, obteniendo como resultado una aplicación creada en módulos que se adaptan a los cambios de las librerías que usan así como también permiten la re-implementación o extensión de un módulo en particular sin que los demás resulten afectados.

5. Requerimientos de documentación

Manual de usuario: Debe proveerse un manual de usuario orientado a los estudiantes. En el mismo debe explicarse cómo y por qué calibrar un robot. También como crear comportamientos, así como una descripción simple de la arquitectura

Manual de instalación: Se incluirá un manual de instalación que explicará paso a paso cómo realizar la instalación del sistema.

Página web: Se debe implementar una página web donde se irá actualizando toda la información relacionada con la elaboración del sistema. Esta poseerá noticias, mantendrá versiones del código en desarrollo y documentos del proyecto.

Pautas para la interfaz de usuario: Se realizará un documento de pautas de la interfaz de usuario. Este poseerá información de los principales estilos gráficos que se usarán en el sistema así como información del estructuramiento de las pantallas incluyendo si fuese necesario bocetos de estas.