

高等院校信息技术规划教材

可视化开发Android应用程序 ——拼图开发模式App Inventor

王向辉 张国印 谢晓芹 编著

1



清华大学出版社

014004543

TN929.53-43
26

高等院校信息技术规划教材

可视化开发Android应用程序 ——拼图开发模式App Inventor

王向辉 张国印 谢晓芹 编著



清华大学出版社



北航

C1691668

TN 929.53-43
26

内 容 简 介

本书介绍了一种崭新的 Android 开发模式,将繁琐的代码开发变为轻松的拼图游戏,不仅可以简化开发过程、降低开发难度,还可以提高开发效率,让开发者在 Android 应用程序开发过程中充满乐趣。

本书分为 8 章,涉及的内容包括 App Inventor 的开发环境搭建、用户界面开发、数据存储与访问、动画与游戏、地图应用开发等方面,较全面地覆盖了 Android 程序开发所涉及的内容。

本书内容丰富,实用性强,既可作为高等院校信息技术的教材,也可供相关专业人士参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

可视化开发 Android 应用程序:拼图开发模式 App Inventor /王向辉等编著. --北京:清华大学出版社, 2013

高等院校信息技术规划教材

ISBN 978-7-302-32888-9

I. ①可… II. ①王… III. ①移动终端—应用程序—程序设计—高等学校—教材 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2013)第 136396 号

责任编辑:袁勤勇 薛 阳

封面设计:傅瑞学

责任校对:时翠兰

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:15.25

字 数:351 千字

版 次:2013 年 9 月第 1 版

印 次:2013 年 9 月第 1 次印刷

印 数:1~2000

定 价:29.00 元

产品编号:052618-01

前言

foreword

Android 是当今应用最为广泛的智能手机平台,具有丰富的软件资源。Android 软件开发具有一定的难度,一般需要开发者具备一定的软件开发知识和经验,App Inventor 的出现,将非程序人员编写 Android 应用程序的愿望变成了现实。App Inventor 创造的拼图开发方式,简化了复杂的程序编码过程,极大地提升了学习者对软件编程的兴趣,并为初学者创造了一个轻松的开始。

书中所涉及的内容包括 App Inventor 的开发环境搭建、用户界面开发、数据存储与访问、动画与游戏、地图应用开发等方面,较全面地覆盖了 Android 程序开发所涉及的内容。

全书共分 8 章:

第 1 章介绍 App Inventor 的起源、优势与不足,展示出 App Inventor 开发的一些作品,并对互联网上的 App Inventor 学习资源进行简单的介绍。

第 2 章详细说明 App Inventor 开发环境的安装和账号注册方法,并简单地介绍如何使用模拟器和手机进行调试。

第 3 章介绍开发 App Inventor 应用程序的基础知识和基本方法,说明 App Inventor 的界面编辑器和模块编辑器的作用和使用方法。

第 4 章详细介绍 App Inventor 程序开发的基础内容,包括条件判断、循环、列表和函数,并简单说明如何区分模块的事件、属性和方法。

第 5 章介绍 App Inventor 界面设计和开发的方法,重点介绍常见的基础控件、媒体控件和社交控件,并对屏幕的布局方式进行讲解。

第 6 章介绍使用 App Inventor 开发游戏与动画的方法,详细讲解画布、精灵和球体的控件使用,并介绍碰撞检测的原理。

第 7 章介绍 App Inventor 数据存储机制,主要讲解 TinyDB 控件的存储、读取、更新和删除数据的方法。

第 8 章介绍 App Inventor 进行地图应用开发的方法,讲解如何

使用位置传感器和谷歌地图的方法,并简单介绍通知控件。

本书由哈尔滨工程大学王向辉、张国印、谢晓芹负责主要编写工作,参与本书编写和校核工作的还有孙宇彤、杨月、宁凡强、张鑫或、何志昌和李晓光,这里对他们的辛苦工作表示衷心的感谢。

同时感谢谷歌(中国)的朱爱民先生、东北大学的李丹程和刘莹老师,感谢他们对 Android 教学和科研工作的帮助,以及对哈尔滨工程大学 Android 人才培养基地的支持。

App Inventor 是一个新兴的开发模式,很多方面还在不断地完善和变化。由于能力和水平所限,虽然竭尽全力,但书中仍然难免存在不足和疏漏的地方,希望各位专家、教师和同学能毫不保留地提出所发现的问题,与编者共同讨论,编者的邮箱为 wangxianghui@live.cn。

App Inventor 屏蔽了 Android 程序开发中复杂的编程细节,因此可供没有程序基础的低年级学生和非计算机专业学生学习,可以在大学一年级和二年级开设这门课程。

所有示例代码和教学资源(教学大纲、教学 PPT、习题答案等)均可在清华大学出版社(<http://www.tup.tsinghua.edu.cn>)或哈尔滨工程大学的网站(<http://android.hrbeu.ed.cn>)上下载。

编 者

2013 年 4 月于哈尔滨工程大学

目录

Contents

第 1 章 Android 与 App Inventor	1
1.1 Android 简介	1
1.2 App Inventor 起源	3
1.3 App Inventor 的优势与不足	5
1.4 App Inventor 作品展示	7
1.5 App Inventor 学习资源	9
习题	12
第 2 章 App Inventor 开发环境	13
2.1 开发前准备	13
2.2 安装 App Inventor 软件	16
2.3 注册 Gmail 邮箱账号	19
2.4 启动 App Inventor	20
2.4.1 界面设计器	22
2.4.2 模块编辑器	23
2.5 程序调试	24
2.5.1 Android 模拟器	24
2.5.2 USB 连接手机	25
2.5.3 WiFi 连接手机	26
习题	27
第 3 章 第一个 App Inventor 程序	28
3.1 创建新工程	28
3.2 界面设计	29
3.3 逻辑模块开发	33
3.4 程序调试	36
习题	37



第 4 章 程序设计基础	38
4.1 程序设计的基本结构	38
4.2 条件判断	38
4.2.1 if 模块	39
4.2.2 布尔表达式	39
4.2.3 ifelse 模块	40
4.2.4 条件嵌套	41
4.3 列表	42
4.3.1 静态列表	42
4.3.2 获取列表项	44
4.3.3 遍历列表	44
4.3.4 动态列表	47
4.3.5 列表嵌套	48
4.4 循环结构	49
4.4.1 foreach	49
4.4.2 while	49
4.5 函数	51
4.5.1 定义与调用	51
4.5.2 函数参数	53
4.5.3 函数返回值	54
4.6 模块分类	55
4.6.1 事件	56
4.6.2 属性	57
4.6.3 方法	57
习题	58
第 5 章 用户界面设计	59
5.1 控件概述	59
5.2 基础控件	60
5.2.1 按钮、标签和图像	60
5.2.2 文本框、复选框和密码框	63
5.2.3 列表选项和时钟	68
5.3 屏幕布局	76
5.3.1 水平布局	77
5.3.2 垂直布局	78
5.3.3 表格布局	78

5.4 媒体控件	79
5.4.1 录像机	79
5.4.2 视频播放器	80
5.4.3 选图工具	83
5.4.4 音频播放器	84
5.5 社交控件	88
5.5.1 选取联系人	89
5.5.2 选取号码	89
5.5.3 邮件地址工具	89
5.5.4 拨号	90
5.5.5 短信息	91
习题	95
第 6 章 动画与游戏	96
6.1 画布	96
6.1.1 画布介绍	96
6.1.2 画布使用	96
6.1.3 相机与加速度传感器	101
6.1.4 示例——画图板	103
6.2 图像精灵	108
6.2.1 精灵介绍	108
6.2.2 精灵使用	108
6.2.3 示例——打地鼠	110
6.3 高级动画功能	116
6.3.1 碰撞检测	116
6.3.2 球的使用	117
6.3.3 方向传感器	118
6.3.4 示例——乒乓球	120
习题	125
第 7 章 数据存储与访问	126
7.1 基础功能	126
7.2 高级功能	131
7.3 示例——注册登录	132
习题	136



第 8 章 地图应用开发	137
8.1 位置传感器	137
8.2 通知控件	139
8.3 谷歌地图	144
习题	150
附录 A Build-In(内置)模块	151
A.1 Definition 模块集(定义模块集)	151
A.2 Text 模块集(文本模块集)	152
A.3 Lists 模块集(列表模块集)	153
A.4 Math 模块集(数学模块集)	155
A.5 Logic 模块集(逻辑运算模块集)	158
A.6 Control 指令集(控制指令集)	158
A.7 Colors 模块集(色彩模块集)	160
附录 B 控件简介	162
B.1 Basic(基本)控件	162
B.2 Media(媒体)控件	177
B.3 Animation(动画)控件	183
B.4 Social(社交)控件	189
B.5 Sensors(传感器)模块	198
B.6 Screen Arrangement(屏幕布局)控件	201
B.7 LEGO MINDSTORMS(乐高机器人)控件	204
B.8 Other stuff(其他)控件	213
B.9 Not ready for prime time(不成熟)控件	226

第 1 章

chapter 1

Android 与 App Inventor

App Inventor 是一个基于网络、可拖曳的 Android 程序开发环境,它将枯燥的代码变成一块一块的拼图,使 Android 软件开发变得简单有趣,使不懂编程的用户也可以开发出属于自己的 Android 应用程序。通过本章的学习,读者将了解 Android 系统的特点和发展趋势,掌握 App Inventor 的起源、优势和不足,了解互联网上的一些 App Inventor 学习资源。

本章学习目标:

- 了解 Android 系统的起源和发展趋势;
- 掌握 App Inventor 的优势和不足;
- 了解 App Inventor 学习资源。

1.1 Android 简介

Android 是 Google 发布的基于 Linux 平台的开源手机操作系统。Android 一词的本义是“机器人”,国内多称为“安卓”或“安致”。Android 最初应用在智能手机和平板电脑上,是第一个完整、开放、免费的手机操作系统。Android 界面如图 1.1 所示。



图 1.1 Android 界面

Android 由安迪·罗宾 (Andy Rubin) 创建于 2003 年, 于 2005 年被 Google 收购。2007 年 11 月 5 日, Google 公司正式向外界展示了这款名为 Android 的操作系统。

Android 是当今世界上应用最广泛的智能手机平台。截止到 2012 年 10 月, Android 的应用软件已有约 70 万个, 在 Google Play 的软件下载量达到了 250 亿次。2012 年 9 月的统计数据表明, 每天的 Android 设备激活数量已经超过了 130 万台, 总激活的设备量超过 4.8 亿。Android 在全球智能手机操作系统市场所占的份额为 76%, 在中国市场的占有率高达 90%。

Android 应用得如此广泛, 与其自身的特点是分不开的。首先, Android 是一款开源的手机操作系统, 有效地缩短了开发周期, 降低了开发成本。用户可以免费下载 Android 源代码, 并在原系统的基础上进行二次开发, 创造具有个性化的 Android 系统。Android 为各种应用程序提供平等的性能支持, 能够满足用户对不同应用的需求, 用户可以下载自己喜欢的软件到手机上, 也可更改手机的界面或图片的浏览方式, 使自己的手机与众不同。借助谷歌在互联网运营方面的优势, 谷歌地图、邮件和搜索等服务直接内置在 Android 系统中, 作为用户与互联网之间的重要纽带, 增强了手机的实用功能。

Android 系统最初只是为智能手机所设计, 但随着应用领域的不断拓展, Android 系统被广泛应用在平板电脑、电视、手表、眼镜、冰箱、耳机和跑步机等设备中, 使人们的生活变得越来越智能化。

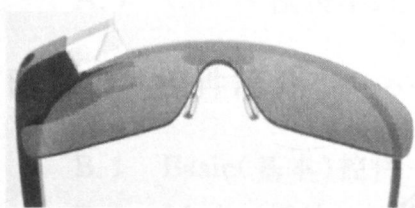


图 1.2 谷歌眼镜

“谷歌眼镜”是谷歌公司在 2012 年 4 月发布的一款“扩展现实”眼镜产品, 如图 1.2 所示, 可以语音拍照、视频通话和辨别方向, 也可以访问互联网信息、处理文字信息和电子邮件。眼镜的右眼镜片上安装了一个微型

投影仪和一个摄像头, 投影仪用以显示数据, 摄像头用来拍摄视频和照片, 再通过传感器进行存储和传输, 而操控模式可以是语言或触控。

基于 Android 系统的 i'm Watch 智能手表, 可以显示时间和天气预报, 还可以显示短信息和联系人等, 并可与其他 Android 系统手机连接, 如图 1.3 所示

联想智能电视尺寸达到了 55 英寸, 运行最新的 Android 4.0 Ice Cream Sandwich 系统, 内置 500 万像素摄像头, 支持 SD 卡插槽, 如图 1.4 所示。



图 1.3 i'm Watch 智能手表



图 1.4 联想 K91 智能电视

三星 Android 冰箱是一台有着内置应用程序的冰箱,功能包括显示照片、播放音乐和给家人留便条等,三星 Android 冰箱还有一个专门用来除霜以及改变温度的应用程序,如图 1.5 所示。

Admiral Touch 耳机同样是基于 Android 系统,配备了一块 2.4 英寸的彩色触摸屏,用户可以用它玩游戏和看电影,这款耳机支持 2.4G 和蓝牙通信,具备 7.1 虚拟环绕音效,如图 1.6 所示。



图 1.5 三星 Android 冰箱



图 1.6 Admiral Touch 耳机

1.2 App Inventor 起源

Android App Inventor 曾是谷歌实验室的一个子计划,于 2010 年 7 月推出,是一款所见即所得的 Android 应用创建器,它允许没有编程知识的用户,通过拖曳特定的应用程序行为模块来创建 Android 应用。2011 年 8 月,谷歌公司将该项目的源代码对外开放,并于 2012 年 1 月将该项目移交给麻省理工学院移动学习中心(The MIT Center for Mobile Learning),由麻省理工学院的 Hal Abelson 教授领导开发,并与 2012 年 3 月向互联网用户开放使用,更名为 MIT App Inventor,如图 1.7 所示。

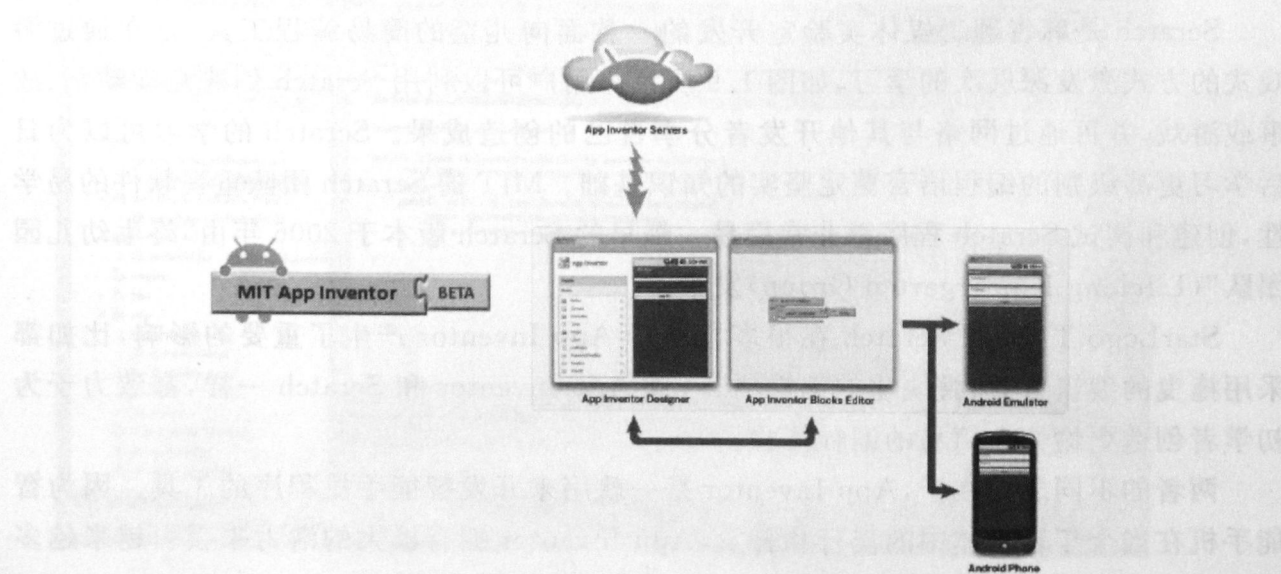


图 1.7 MIT App Inventor



App Inventor 的研发目标是“使人们在移动通信的世界里成为创造者，而不仅是消费者”。App Inventor 创造的拼图开发方式，简化了复杂的程序编码过程，极大地提升了学习者对软件编程的兴趣，并为初学者创造了一个轻松的开始。

App Inventor 的模块编辑语言是基于 Open Blocks 项目的研究成果创建的，而 Open Blocks 项目与 StarLogo TNG、Scratch 这两个项目联系紧密。对 App Inventor 模块编辑器的界面构想，在很多方面借鉴了 StarLogo TNG 和 Scratch 的研究成果，下面分别对这两个项目进行介绍。

StarLogo TNG(The Next Generation)是一种基于主体的仿真语言，由麻省理工媒体实验室和教师教学计划共同研发，其设计主要目的是针对计算机教育，可以用来模拟分散式控制系统的行为，如图 1.8 所示。StarLogo TNG 能够利用开放式图形库提供 3D 视野，并运用模块图形语言来增强易用性和易学性。



图 1.8 StarLogo TNG

Scratch 是麻省理工媒体实验室开发的一款面向儿童的简易编程工具，旨在通过游戏式的方式激发深层次的学习，如图 1.9 所示。用户可以利用 Scratch 创建互动动画、故事或游戏，并可通过网络与其他开发者分享自己的创造成果。Scratch 的学习可以为日后学习更高级别的编程语言奠定坚实的知识基础。MIT 的 Scratch 团队重视软件的易学性，创建和调试 Scratch 程序都非常简易。最早的 Scratch 版本于 2006 年由“终生幼儿园团队”(Lifelong Kindergarten Group)发布。

StarLogo TNG 和 Scratch 在很多方面对 App Inventor 产生了重要的影响，比如都采用拖曳的编辑方式、模块化的编辑语言，且 App Inventor 和 Scratch 一样，都致力于为初学者创造更愉快和简易的编程体验。

两者的不同之处在于，App Inventor 是一款用来开发智能手机程序的工具。因为智能手机在当年轻群体中的流行和普及，App Inventor 拥有极大的潜力来吸引越来越多的年轻人从事软件开发工作，并运用计算思维分析和解决问题。

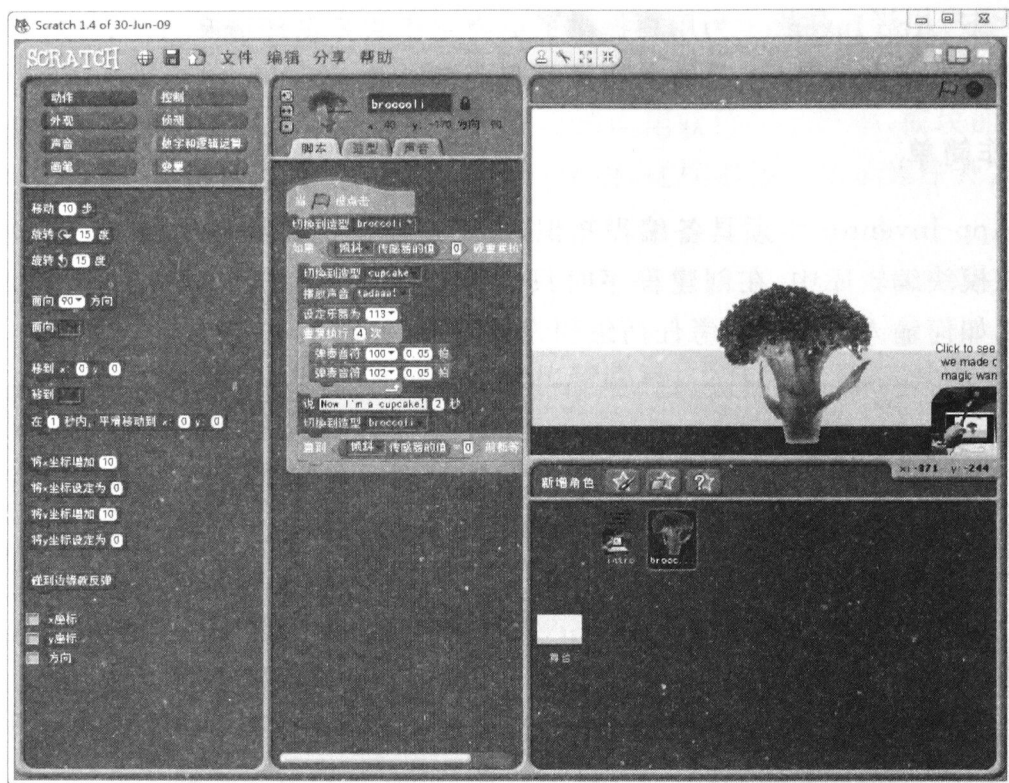


图 1.9 Scratch

1.3 App Inventor 的优势与不足

目前比较流行的 Android 开发方式是使用 Eclipse 编写 Java 代码, Eclipse 集成开发环境如图 1.10 所示。使用代码进行程序开发是目前较为成熟且普遍的方法, 这种开发方式对开发人员的开发知识和经验具有一定的要求, 对于刚刚接触程序开发或者没有程序开发经历的用户来说, 使用代码开发是一件较为困难的事情。

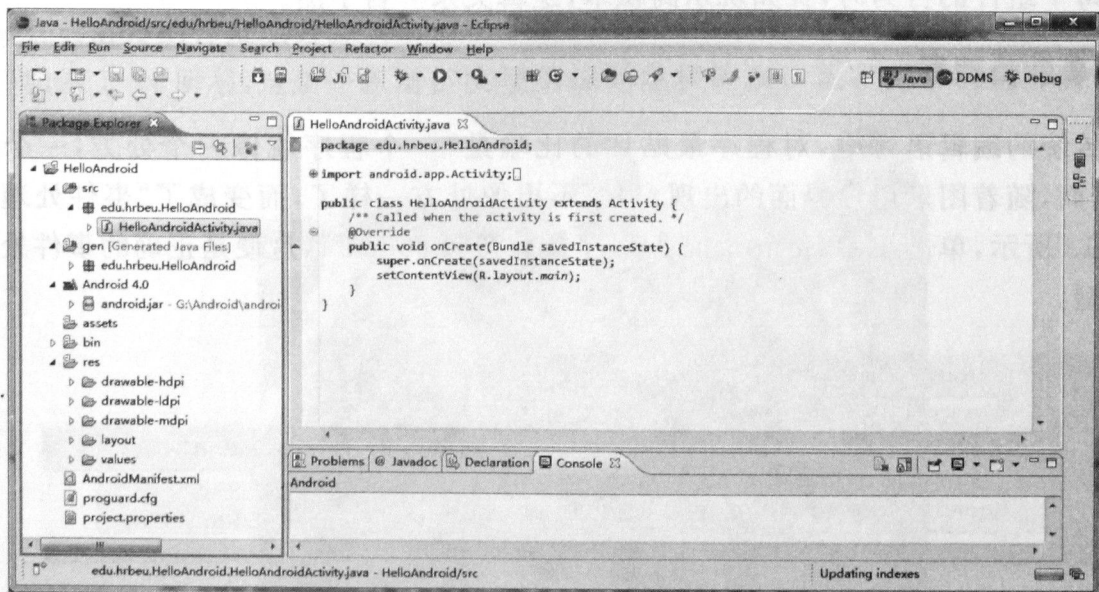


图 1.10 Eclipse 集成开发环境

相比之下, App Inventor 为用户提供了更为便捷的开发环境和方法, 具有操作简单、可视化、模块化、事件置顶、正确性高和便于调试等优点。

1. 操作简单

使用 App Inventor 无须具备编程知识, 也不需要记忆和编写代码, 程序的组件和功能都存储在模块编辑库中, 在创建程序时只需将其拖曳到编辑区域进行组合即可, 用户不需要记忆如何输入指令或参考任何编程设计手册, 如图 1.11 所示。

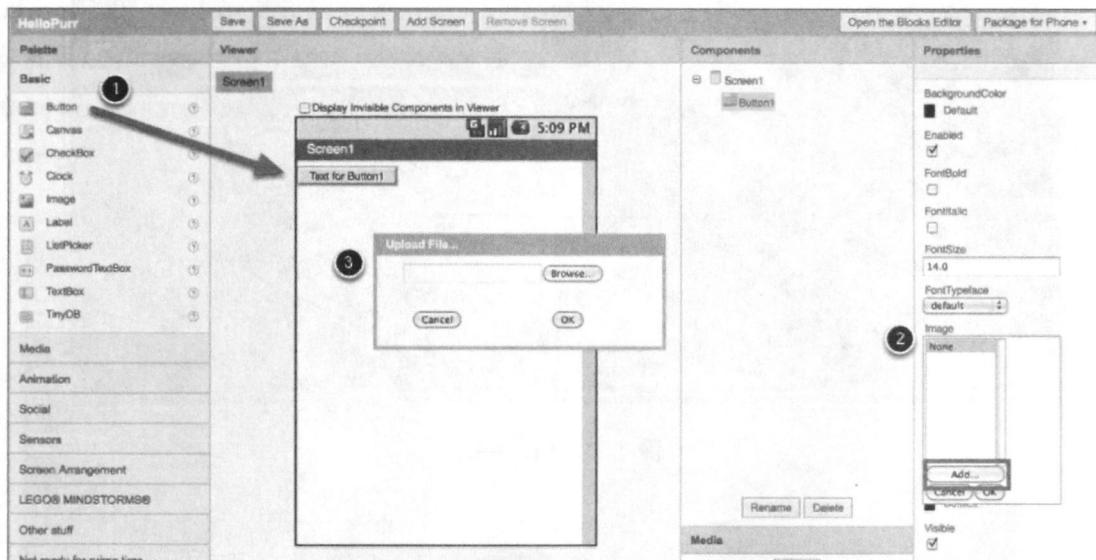


图 1.11 界面编辑器拖曳组件完成界面开发

2. 可视化和模块化

在 App Inventor 中, 不仅用户界面开发是可视化和模块化的, 程序逻辑的开发也是如此。在图 1.12 中, 模块被分为不同的类别, 并且标记成不同的颜色, 执行不同的动作。在设置每个组件的行为时, 犹如玩乐高积木, 逻辑关系一目了然。

3. 事件处理器

在传统的编辑语言中, 对程序最贴切的比喻是“一个程序就像一个处方, 一个说明书”。然而, 随着图形用户界面的出现, 程序不再像处方一样了, 而变成了“事件处理器”。如图 1.13 所示, 单击按钮 Button 1 时, 程序播放音频 sound 1, 这便是正确的事件处理器概念模型。

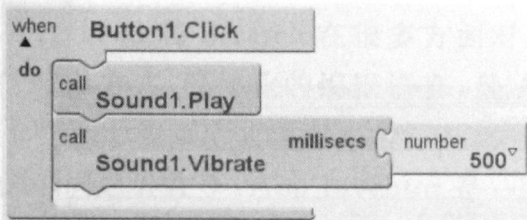


图 1.12 App Inventor 模块化编辑语言

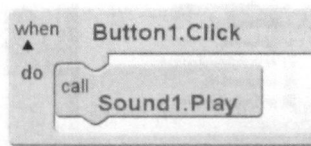


图 1.13 事件置顶

对于 App Inventor 来说,一个应用程序便是一套事件处理器。当用户想要设计一个按钮被单击后的效果时,只需拖曳出按钮的单击事件模块,并把单击后要发生的行为模块放置在按钮单击事件模块中就可以了。在设计应用软件的过程中,模块的每个功能行为都预先设计好,并摆放在开发环境中供用户使用,这样不但简化了程序开发工作,也使整个编程过程显得分外清晰。

4. 正确性高且便于调试

在代码式编程过程中,出现错误后信息比较隐秘,无法简单地遏制错误的发生。而 App Inventor 的模块编辑语言可以从一开始就限制编程的出错几率。例如,如果选择了一种参数模块槽,便无法将其他类型的参数模块与其拼接,这样便降低了参数设置错误的几率。App Inventor 允许相匹配的模块进行拼接,这个特点在一定程度上保证了编程的正确性。

如果编程过程中出现了错误,可以利用 App Inventor 的回收站,用户只要将错误的组件直接拖曳进去便可删除,这比起代码开发方式中对错误的修补要方便简捷得多。在应用程序的开发过程中,用户可以随时在自己的 Android 设备上或模拟器上进行调试,发现的错误可以随时进行修改。

当然,App Inventor 也有它的不足之处。目前 App Inventor 开发出来的程序,只能通过下载安装在手机上,或者下载到计算机上,但不能发布到 Google Play 上供他人下载。App Inventor 目前还不能产生 Java 代码,因此也无法将开发完成后的程序导入 Eclipse 进行再次开发。上述 App Inventor 的不足之处,谷歌官方正在积极解决,希望不久的将来 App Inventor 可以更加成熟、易用。

1.4 App Inventor 作品展示

本节介绍使用 App Inventor 制作出来的手机软件作品,以便使读者能够直观地了解 App Inventor 的开发能力。

如图 1.14 所示,这是一款增扩实境的国际象棋游戏。玩家用手机选择自己的团队



图 1.14 增扩实境国际象棋游戏

和开棋的位置,利用室外的开阔地作为棋盘,然后他们充当棋子来回移动。游戏通过 GPS 来记录每个玩家的移动轨迹,并在手机屏幕上展示玩家在虚拟棋盘上的位置。

美国海军陆战队上士 Chris Mstzger,利用 App Inventor 开发了一款应用软件,可以帮助海军陆战队士兵摧毁在战场上发现的炸药,如图 1.15 所示。

在海地,人道主义开源软件项目利用 App Inventor 开发了两款软件,帮助了那里的人道主义救援人员实地记录降雨量和物价的变化,如图 1.16 所示。



图 1.15 弹药检查软件



图 1.16 统计软件

阿拉巴马州劳伦斯郡高中的学生用 App Inventor 开发了一款物种检查软件,用来记录野猪的出没,如图 1.17 所示。这款软件所记录的数据,将帮助科学家了解野猪入侵的问题。

Google 图书搜索软件在搜索时,用户可以输入书籍的全名或书名的关键词,然后根据用户所输入的内容显示最相关的书籍信息,图 1.18 是该软件的运行界面。



图 1.17 物种检查软件

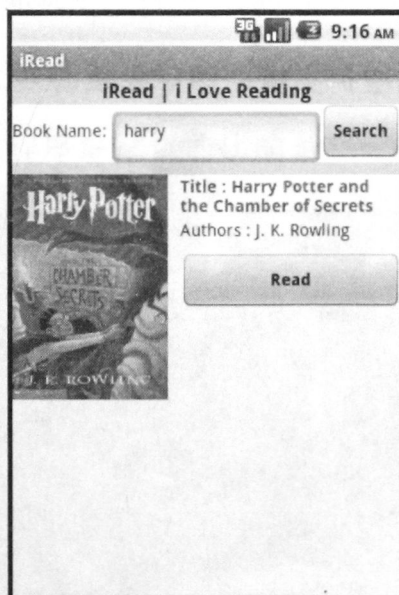


图 1.18 Google 图书搜索软件