TouchDevelop
Programming on the Go
TouchDevelop
Programming on the Go

R Nigel Horspool
University of Victoria

Nikolai Tillmann
Microsoft Research
# Contents

**Preface**

- Who this book is for  xi
- Background to the book   xii
- Other learning materials  xiii
- Acknowledgments     xiii
- Important websites      xiii
- The TouchDevelop Team  xiii

**List of figures** xvii

**List of tables** xix

**Chapter 1  Introduction to TouchDevelop** 1

1.1 Computers want to be programmed  1
1.2 What is TouchDevelop?    2
1.3 The TouchDevelop ecosystem   4
1.4 History and Future   6
1.5 Platforms     7
1.6 The scripting language  9

**Chapter 2  The Scripting Language** 11

2.1 Introduction – the language flavor  11
2.2 Datatypes and variables   15
2.3 Expressions     22
2.4 Statements     28
2.5 Actions     31
2.6 Events     34
2.7 Pages     36
2.8 Creating library scripts   36
Chapter 3  The Wall – using the screen  37
  3.1  Output – the writing on the wall  37
  3.2  Input of values from the touchscreen  42
  3.3  Updating the wall’s content  43
  3.4  Events on the touchscreen  45
  3.5  Pushing and popping pages  49
  3.6  Titles and subtitles  49
  3.7  Wall buttons  50
  3.8  On-demand creation of output  52

Chapter 4  The Web  53
  4.1  URLs and webpages  53
  4.2  Downloading and uploading files  58
  4.3  Downloading structured data  62
  4.4  REST guidelines and web requests  69

Chapter 5  Audio  73
  5.1  Music  73
  5.2  Sounds  79
  5.3  Microphone  81

Chapter 6  Camera, Graphics and Video  83
  6.1  Camera  83
  6.2  Working with pictures  87
  6.3  Static graphics drawing and display  93
  6.4  Playing videos from the internet  96

Chapter 7  Sensors  97
  7.1  The sensors  97
  7.2  Sensor-driven events  98
  7.3  Accelerometer  99
  7.4  Compass  104
  7.5  Gyroscope  105
7.6 Motion

Chapter 8 Interactions
8.1 Social messages
8.2 Locations, places, maps
8.3 Emails
8.4 Phone Calls
8.5 2D barcodes
8.6 SMS messages (WP8 only)
8.7 Calendar and appointments (WP8 only)
8.8 Contacts (WP8 and Android only)

Chapter 9 Game Board
9.1 Introduction
9.2 The Board datatype
9.3 The Sprite datatype
9.4 The Sprite Collection datatype
9.5 Touching and board events
9.6 Debugging games

Chapter 10 UI with Boxes and Pages
10.1 Page Overview
10.2 Box Overview
10.3 Examples of Boxes and Pages
10.4 Working with Pages
10.5 Live Editing of the User Interface
10.6 API Support for Boxes and Pages

Chapter 11 Authenticating Web Services
11.1 Registering your app
11.2 Authenticating
11.3 Libraries
11.4 Advanced topics
Appendix A Editing TouchDevelop Scripts

A.1 The starting point 164
A.2 The editing steps 166
A.3 Additional steps 173
A.4 More advanced editing features 174

Appendix B TouchDevelop Services

B.1 bazaar 178
B.2 box 178
B.3 collections 180
B.4 colors 180
B.5 contract 182
B.6 invalid 182
B.7 languages 184
B.8 locations 184
B.9 maps 185
B.10 math 185
B.11 media 187
B.12 phone 188
B.13 player 188
B.14 senses 189
B.15 social 190
B.16 tags 191
B.17 tile 191
B.18 time 192
B.19 wall 192
B.20 web 194

Appendix C TouchDevelop Datatypes

C.1 Appointment 197
C.2 Appointment Collection 197
C.3 Board 198
C.4 Boolean 199
C.5 Camera 199
C.6 Color 200
C.7 Contact 200
C.8 Contact Collection 201
C.9 DateTime 202
C.10 Form Builder 203
C.11 Json Builder 203
C.12 Json Object 204
C.13 Link 204
C.14 Link Collection 205
C.15 Location 205
C.16 Location Collection 206
C.17 Map 206
C.18 Matrix 207
C.19 Message 208
C.20 Message Collection 209
C.21 Motion 210
C.22 Number 210
C.23 Number Collection 211
C.24 Number Map 211
C.25 OAuth Response 212
C.26 Page 213
C.27 Page Button 213
C.28 Page Collection 213
C.29 Picture 213
C.30 Picture Album 215
C.31 Picture Albums 216
C.32 Pictures 216
C.33 Place 216
C.34 Place Collection 217
C.35 Playlist 218
C.36 Playlists 218
C.37 Song 218
<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>D.1</td>
<td>Supported Browsers</td>
<td>231</td>
</tr>
<tr>
<td>D.2</td>
<td>General Features</td>
<td>231</td>
</tr>
<tr>
<td>D.3</td>
<td>Supported Sensors and Devices</td>
<td>232</td>
</tr>
<tr>
<td>D.4</td>
<td>Support for Services/Resources</td>
<td>233</td>
</tr>
<tr>
<td>D.5</td>
<td>Support for Created Apps</td>
<td>233</td>
</tr>
<tr>
<td>E.1</td>
<td>The sample program</td>
<td>235</td>
</tr>
<tr>
<td>E.2</td>
<td>The back button, undo and mistakes</td>
<td>235</td>
</tr>
<tr>
<td>E.3</td>
<td>The editing example</td>
<td>236</td>
</tr>
<tr>
<td>E.4</td>
<td>Additional steps</td>
<td>241</td>
</tr>
<tr>
<td>E.5</td>
<td>Refactoring code into a new action</td>
<td>243</td>
</tr>
</tbody>
</table>
Preface

The sales figures for smartphones continue to rise exponentially. Tablet computers are showing a similarly phenomenal adoption rate and are replacing laptop computers in many areas of life. We can imagine a time when nearly everyone is carrying around a powerful computer in the form of a smartphone or a tablet. The term *mobile device* is used to cover such devices. Typically, an *app* (an application program) for a smartphone or tablet has to be developed in a PC and transferred to the mobile device later. But does it have to be that way? The TouchDevelop project at Microsoft Research has proved that the answer is No. TouchDevelop is a programming environment that runs on all mobile devices. It allows a script to be developed on a mobile device, or on a PC, and to be run on any mobile device or a PC. After releasing the app in 2011 when it was available only for the Windows Phone, the overwhelming response was a big surprise: more than 200,000 users downloaded the app and they published more than 10,000 scripts written entirely on phones. Since then, TouchDevelop has been made available in a form that runs on PC, Mac and Linux platforms, and on iPad, iPhone, iPod Touch and Android devices. TouchDevelop is truly a portable development environment for creating portable apps.

Who this book is for

Mobile devices represent the latest in technology. Furthermore, many students actually own their own smartphone. High school teachers and college or university instructors love the idea of using the latest technology to engage their students. While they may be experts in the field of teaching programming, many teachers appreciate guidance on how to navigate a complex app like TouchDevelop: its visual program editor is designed for touchscreens and uses different editing paradigms from a traditional keyboard-based text processor. Another opportunity and challenge is how to make use of some of the sensors that a modern mobile device has to offer.

This book has much to offer to both both teachers and self-starting students who are learning how to program on their own. For teachers, it walks in detail through all of the screens of the app, and it points out similarities and differences of the TouchDevelop language compared to other programming languages that the teacher might already be familiar with. For students and
enthusiasts, the book can serve as a handy reference which they keep next to the device they are using – it is particularly useful when that device has a small screen. The book systematically addresses all programming language constructs, starting from the very basic constructs such as variables and loops. The book also explores many of the phone sensors and data sources which make creating apps for mobile devices so rewarding.

If you are new to programming with TouchDevelop, or if you have not yet worked on touchscreen devices, we suggest that you read the book starting from Chapter 1. If you are already familiar with the basic paradigm of the TouchDevelop programming environment, then feel free to jump ahead to the later chapters that address particular topic areas.

This book is written from the perspective of a person developing their code using a browser. All screenshots and navigation instructions refer to the TouchDevelop Web App running in a browser and is applicable to all platforms except the Windows Phone. Only Appendix E, which covers the editor on the Windows Phone, uses screenshots and instructions specific to the Windows Phone.

This book is available online as well as being publish in print form by APress. Please email touchdevelop@microsoft.com to give feedback.

**Background to the book**

This edition of the book is the result of the year-long evolution of earlier book versions, incorporating feedback from tutorials and lectures given by the authors. The first version of the book was produced as limited edition of 75 copies for the ACM SIGCSE Conference in Raleigh, NC, March, 2012. That book was based on the recently released Version 2.6 of TouchDevelop. An updated copy of the book with 1000 copies, based on Version 2.10, was printed in January 2013. This book was made available via a Creative Commons Licence and put on the Amazon Bookstore as well as the TouchDevelop website. Much of the contents of the second book were also applicable to the Web App version of TouchDevelop, though all the screenshots were still of a phone. This third version has been retargeted at the Web App version of TouchDevelop.
Other learning materials

On the TouchDevelop website, you can also find extensive videos, tutorials and slides to help you learn and teach TouchDevelop. Just tap (or click) on the large tile labeled “Docs” under the “Chat and Learn” heading once you have logged in to the TouchDevelop website to find these learning resources.

Comments are very welcome. To contact the TouchDevelop team or the authors, you can

- Send email to touchdevelop@microsoft.com
- Post on https://facebook.com/touchdevelop
- Post on the forum in the app

Acknowledgments

As the TouchDevelop community grows, we are finding that we are learning from everyone who engages in the project – students at Hackathons, academics who write papers, and most of all developers of the amazing apps in the bazaar. Thanks to all of you.

Important websites

https://www.touchdevelop.com
https://www.facebook.com/TouchDevelop
http://research.microsoft.com/touchdevelop

The TouchDevelop Team

Thomas (Tom) Ball is a principal researcher and research manager at Microsoft Research, Redmond, widely known for his work in program profiling, software model checking, program testing, and empirical software engineering. Ball is a 2011 ACM Fellow for "contributions to software analysis and defect detection." Since becoming a manager at Microsoft, he has nurtured and grown research areas such as automated theorem proving, program testing and verification, and empirical software engineering. He
holds a B.A. in Computer Science from Cornell University and a M.S. and Ph.D. from the University of Wisconsin-Madison.

Judith Bishop is Director of Computer Science at Microsoft Research, based in Redmond, USA. Her role is to create strong links between Microsoft’s research groups and universities globally, through encouraging projects, supporting conferences and engaging directly in research. Her expertise is in programming languages and distributed systems, with a strong practical bias and an interest in compilers and design patterns. She is the author or editor of 17 books on programming languages. She has a PhD from the University of Southampton, UK in Computer Science.

Sebastian Burckhardt is a Researcher at Microsoft Research. He was born and raised in Basel, Switzerland. His research interests revolve around the general problem of programming concurrent, parallel, and distributed systems conveniently, efficiently, and correctly. More specific interests include consistency models, concurrency testing, self-adjusting computation, and the concurrent revisions programming model. After a few years of industry experience at IBM, he earned his PhD in Computer Science at the University of Pennsylvania.

Juan Chen is a Researcher in the RiSE group at Microsoft Research Redmond. Her main research areas include compilers, programming verification, and type systems. She has worked on certifying compilers for object-oriented languages, and design and implementation of a functional programming language for specifying and verifying program properties. She has a PhD in Computer Science from Princeton University.
Jonathan 'Peli' de Halleux is a Software Engineer in the Research in Software Engineering group at Microsoft Research. Peli also volunteers at the local high school to teach mobile computer science. From 2004 to 2006, he worked in the Common Language Runtime (CLR) as a Software Design Engineer in Test in charge of the Just In Time compiler. He has a PhD in Applied Mathematics from the Catholic University of Louvain, Belgium.

Manuel Fähndrich is a Senior Researcher in the RiSE group at Microsoft Research in Redmond. He works on programming language design, static type systems, program analysis and verification, as well as runtime techniques and optimizations. His past and current project involvements include the Singularity OS and Sing# language, CodeContracts for .NET, and TouchDevelop. He has a PhD from the University of California, Berkeley.

Nigel Horspool is a professor of computer science at the University of Victoria. His main focus for research and teaching has been programming languages and compilers, though his main claim to fame is a string searching algorithm. He is the author or co-author of three books, which cover the C language, Unix and the C# language. He is currently the co-editor of the journal 'Software: Practice and Experience'. He has a PhD from the University of Toronto, Canada in Computer Science.

Michał Moskal is a Researcher in Redmond. He is in the RiSE group working on software verification, automated theorem proving, and programming languages. He works on a formal verifier for concurrent C programs called VCC, while also taking on other projects including Boogie intermediate verification language, SPUR tracing JIT, and DKAL authorization engine. He has a PhD from the University of Wrocław, Poland.
Arjmand Samuel works with the academic community to foster research and collaborations in the devices and services research areas. He leads the mobile and cloud computing research and outreach for Microsoft Research (Project Hawaii and TouchDevelop). His recent research interests are in software architectures and programming paradigms for devices of all shapes and forms (TouchDevelop and HomeOS). He has published in a variety of publications on topics of security, privacy, location aware access control, and innovative use of mobile technology. Samuel has a Ph.D. in Information Security from Purdue University.

Nikolai Tillmann is a Principal Research Software Design Engineer, Microsoft Research. His main areas of research are program authoring on mobile devices, program analysis, testing, optimization, and verification. He started the TouchDevelop project, which enables end-users to write programs for mobile devices on mobile devices. He also leads the Pex project, in which he develops together with Peli de Halleux a framework for runtime verification and automatic test case generation for .NET applications based on parameterized unit testing and dynamic symbolic execution. Nikolai has a Dipl. Inf. in Computer Science from TU Berlin, Germany.
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-1</td>
<td>The TouchDevelop ecosystem</td>
<td>4</td>
</tr>
<tr>
<td>1-2</td>
<td>Viewing metadata of a script</td>
<td>5</td>
</tr>
<tr>
<td>1-3</td>
<td>Editing a script</td>
<td>6</td>
</tr>
<tr>
<td>2-1</td>
<td>The ‘new songs’ script (/okzc)</td>
<td>13</td>
</tr>
<tr>
<td>3-1</td>
<td>Simple output, normal and reversed order</td>
<td>37</td>
</tr>
<tr>
<td>3-2</td>
<td>Displaying a string using a TextBox</td>
<td>37</td>
</tr>
<tr>
<td>3-3</td>
<td>Displaying composite values</td>
<td>39</td>
</tr>
<tr>
<td>3-4</td>
<td>Prompting for input</td>
<td>42</td>
</tr>
<tr>
<td>3-5</td>
<td>An updatable textbox (/censaair)</td>
<td>43</td>
</tr>
<tr>
<td>3-6</td>
<td>Updating text using a board (/wkoxnasz)</td>
<td>44</td>
</tr>
<tr>
<td>3-7</td>
<td>Using tap wall events</td>
<td>45</td>
</tr>
<tr>
<td>3-8</td>
<td>Using sprite events (/akmcnpux)</td>
<td>47</td>
</tr>
<tr>
<td>3-9</td>
<td>Title and subtitle example</td>
<td>49</td>
</tr>
<tr>
<td>3-10</td>
<td>The ‘Question Mark’ page button</td>
<td>50</td>
</tr>
<tr>
<td>4-1</td>
<td>Posting a Webpage Link to the wall</td>
<td>55</td>
</tr>
<tr>
<td>4-2</td>
<td>Posting a link to an image on the wall</td>
<td>56</td>
</tr>
<tr>
<td>4-3</td>
<td>Downloading a text file</td>
<td>58</td>
</tr>
<tr>
<td>4-4</td>
<td>Picture download</td>
<td>59</td>
</tr>
<tr>
<td>4-5</td>
<td>Snapshot of a weather webpage</td>
<td>62</td>
</tr>
<tr>
<td>4-6</td>
<td>Weather data in JSON format</td>
<td>64</td>
</tr>
<tr>
<td>4-7</td>
<td>Accessing Twitter with a library</td>
<td>66</td>
</tr>
<tr>
<td>4-8</td>
<td>Weather data in JSON format</td>
<td>67</td>
</tr>
<tr>
<td>5-1</td>
<td>The ‘new songs’ script (WP8 and Android only)</td>
<td>77</td>
</tr>
<tr>
<td>6-1</td>
<td>A simplified camcorder script (/xbhl)</td>
<td>83</td>
</tr>
<tr>
<td>6-2</td>
<td>Computing brightness</td>
<td>88</td>
</tr>
<tr>
<td>6-3</td>
<td>Blending two pictures</td>
<td>90</td>
</tr>
<tr>
<td>6-4</td>
<td>Using the draw ellipse method</td>
<td>92</td>
</tr>
<tr>
<td>7-1</td>
<td>A simple pedometer program (/jbpv)</td>
<td>98</td>
</tr>
<tr>
<td>7-2</td>
<td>Accelerometer orientation</td>
<td>100</td>
</tr>
<tr>
<td>7-3</td>
<td>Accelerometer colors simplified (script /tbcb)</td>
<td>101</td>
</tr>
</tbody>
</table>
Figure 7-4: Magnetic compass script (script /drvu) .............................................. 103
Figure 7-5: Methods of the Motion type ................................................................. 106
Figure 7-6: Yaw, pitch and roll ........................................................................... 106
Figure 8-1: Examples of 2D barcodes..................................................................... 115
Figure 8-2: Sending an SMS message .................................................................. 116
Figure 9-1: Example script: a moving ball (/nyuc) .............................................. 124
Figure 10.1: Page Example 1 (/bhugenw) ............................................................. 140
Figure 10.2: (a) Result from Page Example 1 ....................................................... 141
Figure 10.3: Page Example 2 (/hnimxaiw) ............................................................. 143
Figure 10.4: Result of Running Page Example 2. ................................................ 144
Figure 10.5: Page Example 3 (/wrsonnwh) ............................................................. 145
Figure 10.6: Translation produced by Page Example 3 ......................................... 146
Figure 10.7: Icons for User Interface Editing .......................................................... 148
Figure 11-2: Using Facebook Library ...................................................................... 156
Figure A-1: The rotor program /gtbd ................................................................. 158
Figure A-2: The first few script templates ............................................................. 160
Figure A-3: The editor webpage ........................................................................... 161
Figure A-4: The left keypad .................................................................................. 164
Figure A-5: The right keypad ............................................................................... 164
Figure A-6: Add above and add below buttons ..................................................... 166
Figure A-7: The running script ............................................................................ 168
Figure A-8: Selecting the first line ................................................................. 170
Figure A-9: Marking the first line to extract ......................................................... 171
Figure A-10: Marking the last line to extract ...................................................... 171
Figure A-11: Naming the extracted code ............................................................. 171
Figure E-1: The rotor program /cqxk ................................................................. 236
Figure E-2: Getting started screenshots .............................................................. 237
Figure E-3: Editing the first line ........................................................................... 239
Figure E-4: Running the script ............................................................................ 242
List of tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Special symbols used in scripts</td>
<td>12</td>
</tr>
<tr>
<td>2-2</td>
<td>The Value types</td>
<td>15</td>
</tr>
<tr>
<td>2-3</td>
<td>Reference types provided by the API</td>
<td>17</td>
</tr>
<tr>
<td>2-4</td>
<td>Regular collection types</td>
<td>19</td>
</tr>
<tr>
<td>2-5</td>
<td>Special collection types</td>
<td>20</td>
</tr>
<tr>
<td>2-6</td>
<td>Operators</td>
<td>26</td>
</tr>
<tr>
<td>2-7</td>
<td>Events</td>
<td>33</td>
</tr>
<tr>
<td>2-8</td>
<td>Gameboard events</td>
<td>34</td>
</tr>
<tr>
<td>3-1</td>
<td>Display of media values</td>
<td>39</td>
</tr>
<tr>
<td>3-2</td>
<td>Display of social values</td>
<td>40</td>
</tr>
<tr>
<td>3-3</td>
<td>Display of web values</td>
<td>40</td>
</tr>
<tr>
<td>3-4</td>
<td>Prompting for input</td>
<td>41</td>
</tr>
<tr>
<td>3-5</td>
<td>Tap wall events</td>
<td>45</td>
</tr>
<tr>
<td>3-6</td>
<td>Methods of the Page Table datatype</td>
<td>50</td>
</tr>
<tr>
<td>4-1</td>
<td>Converting URLs</td>
<td>53</td>
</tr>
<tr>
<td>4-2</td>
<td>Creating web links</td>
<td>54</td>
</tr>
<tr>
<td>4-3</td>
<td>Uploading/downloading to websites</td>
<td>57</td>
</tr>
<tr>
<td>4-4</td>
<td>Methods of Web Request datatype</td>
<td>69</td>
</tr>
<tr>
<td>4-5</td>
<td>Methods of Web Response datatype</td>
<td>70</td>
</tr>
<tr>
<td>5-1</td>
<td>Supported music formats</td>
<td>71</td>
</tr>
<tr>
<td>5-2</td>
<td>Accessing media resources (WP8 and Android only)</td>
<td>72</td>
</tr>
<tr>
<td>5-3</td>
<td>Using songs and song albums (WP8 and Android only)</td>
<td>73</td>
</tr>
<tr>
<td>5-4</td>
<td>Methods of player resource for songs</td>
<td>74</td>
</tr>
<tr>
<td>5-5</td>
<td>Methods of Sound datatype</td>
<td>78</td>
</tr>
<tr>
<td>6-1</td>
<td>Methods for using the camera(s)</td>
<td>81</td>
</tr>
<tr>
<td>6-2</td>
<td>Methods of the Camera datatype</td>
<td>82</td>
</tr>
<tr>
<td>6-3</td>
<td>Methods of Picture Album and Pictures datatypes (WP8 and Android)</td>
<td>85</td>
</tr>
<tr>
<td>6-4</td>
<td>General Picture methods</td>
<td>86</td>
</tr>
<tr>
<td>6-5</td>
<td>Colorizing/intensity picture effects</td>
<td>89</td>
</tr>
</tbody>
</table>
Table 6-6: Drawing methods of the Picture datatype 91
Table 7-1: Sensing methods of the senses service 95
Table 7-2: Sensor events 96
Table 8-1: Messaging methods of the social service 108
Table 8-2: Extra methods of the Message Collection datatype 109
Table 8-3: Methods of the locations service 110
Table 8-4: Methods of the maps service 111
Table 8-5: Methods of the Map datatype 112
Table 8-6: Methods for handling phone calls 113
Table 8-7: Barcode generation methods 114
Table 8-8: Methods of the Appointment datatype 117
Table 8-9: Methods for accessing and creating contacts 117
Table 9-1: Methods to create a board 121
Table 9-2: Methods of Board datatype: appearance 122
Table 9-3: Methods of Board datatype: creating / accessing sprites 122
Table 9-4: Methods of Board datatype: obstacles / boundaries 124
Table 9-5: Methods of Board datatype: forces / animation 125
Table 9-6: Methods of Sprite datatype: visual attributes 126
Table 9-7: Methods of Sprite datatype: position / velocity 128
Table 9-8: Methods of Sprite datatype: mass, friction, elasticity 130
Table 9-9: Methods of Sprite datatype: additional features 130
Table 9-10: Additional or modified Sprite Set methods 132
Table 9-11: Touch methods of the Board datatype 133
Table 10-1: General Methods of box Service 150
Table 10-2: Text Handling Methods of box Service 150
Table 10-3: Layout Methods of box Service 151
Table 11-1: General methods related to OAuth 2.0 155