

# TouchDevelop: Create Rich Mobile Apps on Touch Devices

Nikolai Tillmann    Michał Moskał    Jonathan de Halleux  
Sebastian Burckhardt    Tom Ball    Judith Bishop  
Microsoft Research, One Microsoft Way, Redmond WA 98052, USA  
{nikolait,micmo,jhalleux,sburckha,tball,jbishop}@microsoft.com

## ABSTRACT

We are experiencing a technology shift: Powerful and easy-to-use mobile devices like smartphones and tablets are becoming more prevalent than traditional PCs and laptops. Mobile devices are going to be the first and, in less developed countries, possibly the only computing devices which virtually all people will own and carry with them at all times. In this tutorial, participants will learn about developing software directly on their mobile devices. The tutorial is based on TouchDevelop, a modern software development environment that embraces this new reality, treating mobile devices as first-class software development machines, instead of relying on legacy development models built around PC. TouchDevelop comes with typed, structured programming language that is built around the idea of only using a touch-screen as the input device to author code. Access to the cloud, flexible user interfaces, and access to sensors such as accelerometer and GPS are available as a first-class citizens in the programming language. TouchDevelop is available as a web app on Windows tablets, iOS, Android, Windows PCs and Macs, and as a native app on Windows Phone.

## Categories and Subject Descriptors

D.3.0 [Programming Languages]: General; D.2.3 [Software Engineering]: Structured programming; D.2.3 [Software Engineering]: Program editors; D.2.6 [Programming Environments/Construction Tools]: Integrated environments; D.2.11 [Software Architectures]: Languages

## General Terms

Languages

## Keywords

mobile devices, Web IDE, smart phone, tablet, touch-based entry

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MOBILESoft '14, June 2-3, 2014, Hyderabad, India

Copyright 2014 ACM 978-1-4503-2878-4/14/06 ...\$15.00.

## 1. OBJECTIVES

The goal of this tutorial is to show to mobile software engineering researchers how software authoring for mobile devices can evolve beyond PC-centric legacy approaches. The tutorial embraces a new mobile-device-centric concept around a cloud-based modern programming language and programming environment.

Participants will gain the following skills and knowledge:

- how to use mobile devices to create mobile applications, without the use of a PC
- a simplified approach to designing user interfaces and interacting with cloud-based data
- how to easily share applications, and communicate with users
- how to gather and analyze crowd-sourced insights data about one's applications
- how to deal with a heterogeneous environment of devices and operating systems.
- what opportunities, limitations and challenges exist when developing mobile applications without a PC
- how to leverage software development on mobile devices for learning and teaching of software engineering

## 2. TOPICS TO BE COVERED

The tutorial is based on TouchDevelop[6], a novel programming environment and language that makes it possible to write applications directly on mobile devices, using touch to its full advantage, and without the need for a separate PC. TouchDevelop is freely available as a web app on PC, Mac, iOS, Android, and as a native app on Windows Phone. Since the first release in April 2011, TouchDevelop for Windows Phone has been downloaded more than 400,000 times, and over 160,000 registered users have published more than 70,000 scripts and exported over 800 apps to the Windows Store and Windows Phone Store.

The tutorial will cover the following topics:

- We motivate the idea of programming on a mobile device itself.

All design decisions of the TouchDevelop programming environment are motivated by one main principle: the programs are created on a mobile device, and they are consumed on a mobile device. By a *mobile device*

we understand a pocket-sized computing device with a touchscreen. This principle has large implications on the design of the language, the program editor, and the execution environment.

As a typical laptop does not fit the above definition of a mobile device, it is not a primary development device, but TouchDevelop will still work on a laptop using a keyboard and a mouse instead of a touchscreen.

- We describe the novel TouchDevelop language.

TouchDevelop aims at making programming easier and accessible for non-expert programmers, a topic that has been studied widely for decades [3]. TouchDevelop uses a semi-structured code editor [5]; the editor is not strictly structured, as it allows arbitrary token-strings at the expression-level, but each token is treated as an atom again. All of the structured editing in TouchDevelop is motivated by the ability to easily edit the code on a touchscreen, where elements are typically selected with a big thumb, and not for the sole purpose of preventing syntax errors. While the TouchDevelop language is edited graphically, it is not a Visual programming language [4], since the interaction with the graphical elements is quite limited.

When creating a mobile application, a large amount of time is typically spent on creating graphical user interfaces, and interfacing with cloud service to retrieve and store data. The TouchDevelop language comes with a built-in way of specifying declarative user interfaces [1], and also with a built-in concept of persistent and cloud-based data [2].

- We give an overview over the TouchDevelop environment: the program management, the code editor, the runtime environment, the debugger, the unit testing framework.

Programs can be easily shared with other TouchDevelop participants, and they can also be exported as Windows Store and Windows Phone Store apps.

Once an author publishes a program, the TouchDevelop environment collects certain runtime information from all users, in particular code coverage information, profiling data showing real-world performance characteristics, and crash information (if any). All of this data is publicly available.

- In an interactive exercise together with the tutorial participants, we will develop a series of applications in TouchDevelop.

The developed applications will illustrate the wide range of possibilities on mobile devices, including: games with graphics and sounds, and also a cloud-based chat application for multiple users.

- We will discuss opportunities and challenges of teaching and learning in the context of our proposed programming environment and language for mobile devices.

Today's classrooms are not well equipped for the transition yet, as traditional PCs and laptops are still prevalent and adoption of mobile devices by schools is slow. However, some students may already own personal mobile devices, and many more will soon. It is likely

that we will find a heterogeneous environment with incompatible phone platforms supporting vastly different novel programming environments.

### 3. PRESENTATION APPROACH

The participants will use either the TouchDevelop Web App or the native TouchDevelop app for Windows Phone. A reasonably fast WiFi connection available to all participants is required. We expect all participants to bring some suitable modern device, which could be a Windows Phone, Windows tablet, iPhone, iPad, Android device. A laptop running Windows or MacOS would also be sufficient.

The hands-on activities include:

- Exercise to create a simple game
- Exercise to create a distributed chat application that stores data in the cloud
- Exercise to create a virtual classroom, manage participants, access and analyze data

### 4. TARGET AUDIENCE

This tutorial is aimed at researchers of software engineering for mobile devices. We also welcome researchers of programming languages, execution environments, mobile computing, as well as computer science educators who are interested in introducing programming on mobile devices early in their teaching approach.

### 5. PREREQUISITE KNOWLEDGE

No theoretical knowledge of mobile computing is required. Practical knowledge of some traditional programming language and environment is suggested. Practical knowledge of how to use some kind of mobile device is suggested.

### 6. REFERENCES

- [1] S. Burckhardt, M. Fähndrich, P. de Halleux, J. Kato, S. McDirmid, M. Moskal, and N. Tillmann. It's Alive! Continuous Feedback in UI Programming. In *Proceedings of the 34th International ACM Conference on Programming Language Design and Implementation (PLDI)*, June 2013.
- [2] S. Burckhardt, M. Fähndrich, D. Leijen, and B. P. Wood. Cloud Types for Eventual Consistency. In *Proceedings of the 26th European Conference on Object-Oriented Programming (ECOOP)*, June 2012.
- [3] C. Kelleher and R. Pausch. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Comput. Surv.*, 37:83–137, June 2005.
- [4] J. V. Nickerson. Visual programming. In *PhD thesis*. New York University, 1994.
- [5] T. Teitelbaum. The cornell program synthesizer: a syntax-directed programming environment. *SIGPLAN Not.*, 14:75–75, October 1979.
- [6] N. Tillmann, M. Moskal, J. de Halleux, and M. Fähndrich. Touchdevelop: programming cloud-connected mobile devices via touchscreen. In *Proceedings of the 10th SIGPLAN symposium on New ideas, new paradigms, and reflections on programming and software*, ONWARD '11, pages 49–60, 2011.