

SMARTPHONE TO MICROCONTROLLER COMMUNICATION ANDROID FRAMEWORK FOR STUDENTS

SUSTAV ZA IZRADU ANDROID APLIKACIJE ZA KOMUNIKACIJU S MIKROUPRAVLJAČEM NAMIJENJEN STUDENTIMA

Petar Zubak¹, Tomislav Novak², and Stipe Predanić²

¹*Styria Media Group AG, Zagreb, Croatia*

²*University of Applied Sciences, Zagreb, Croatia*

Abstract

Aim of this paper is to describe the development and the use of an Android application that is intended to be used by electrical engineering students. By using this application, students can set their own arbitrary graphical interface, connect to a microcontroller through Bluetooth protocol and manage the microcontroller through graphical interface. This process is explained thoroughly and through examples. This application is valuable to students because it simplifies the process of making an embedded project that communicates with a smartphone. In many electrical engineering student projects - regarding aforementioned thematic - it is useful to achieve smartphone to microcontroller communication. Unfortunately, students studying electrical engineering sometimes lack the programming experience in the smartphone software development.

Keywords: *Bluetooth, microcontroller, Android*

Sažetak

Cilj ovog rada je opisati razvoj i korištenje aplikacije namijenjene za Android platformu koja olakšava studentima elektrotehničkog smjera ostvarivanje komunikacije između pametnog telefona i mikrokontrolera. Korištenjem ove aplikacije studenti mogu izgraditi aplikaciju proizvoljnog grafičkog sadržaja, povezati se s mikroupravljačem i komunicirati s istim. Proces ostvarivanja navedenog detaljno je objašnjen u radu. Ova aplikacija je korisna studentima jer pojednostavljuje izradu sustava koji zahtijeva navedenu vrstu komunikacije. U mnogim elektrotehničkim studentskim projektima – koji

su vezani uz gorenavedeno područje – korisno je ostvariti mogućnost komunikacije između pametnog telefona i mikroupravljača. Nažalost, studenti elektrotehničkog smjera dosta često nemaju adekvatno iskustvo programiranja aplikacija za pametne telefone.

Ključne riječi: *Bluetooth, mikroupravljač, Android*

1. Introduction

1. Uvod

It is not difficult to foresee the near future of the technical aspect of human environment. Using various interfaces and communication technologies physical objects can interact with digital world. Lately, a variety of terms have been used for this revolution. Sometimes it is described as Smart grids, Smart metering, Smart Living, Smart Cities, Machine to machine communication or with the paradigm called the Internet of Things. Microcontrollers accompanied with additional peripherals play a great role in this development. Abbreviation M2M sometimes stands for: Man-to-Machine, Machine-to-Man, Machine-to-Mobile and Mobile-to-Machine. In general, M2M refers to communication between embedded devices, smart sensors and actuators, mobile devices without, or with limited, human intervention [1]. This revolution implies the ability of physical objects to communicate with the digital world. These physical objects collect the information from their surroundings or control some aspect of the system. The “smart” part of these objects often consists of some sort of microcontroller and

adequate peripheral hardware – sensors, actuators, memory, etc. Popularization of the aforementioned technical field caused the increase in popularity of microcontroller based projects among students.

Often, student projects include the need for some form of smartphone interaction with the underlying hardware system. Smartphone is used either as a simple information indicator, a controlling device or even as an intermediary device to achieve Internet connectivity. To realize this, the communication between smartphone and microcontroller needs to be established. Bluetooth is a common and easy way to accomplish this goal. It is a wireless communication protocol that is characterized by low-power consumption, short range and low-cost transceiver available in each communicating device. As shown in [3] systems that include a microcontroller and a smartphone that can communicate over a Bluetooth protocol are, also, very cost effective and, because of that, adequate for student projects.

Student, who would like to realize a project of this scope, should possess the knowledge of electrical engineering and electronics, embedded programming and a certain level of mobile software development expertise. Sometimes, students do not have adequate level of knowledge in all of the above-mentioned fields.

The rest of the paper is organized as follows. Section II introduces the problem addressed by this paper. Section III describes the possible solutions to the stated problem. In section IV, one possible solution is proposed. Section V highlights some future improvement possibilities of the proposed solution. The paper is concluded with section VI.

2. Current problem

2. *Trenutni problem*

Often electrical engineering or mechatronics students lack the programming knowledge and experience in field of the smartphone software development. Shortage of this kind of knowledge is not a consequence of student reluctance, but simply a side effect of the university study program.

This scenario is present at undergraduate and specialist graduate professional study of electrical engineering at the University of Applied Sciences

in Zagreb. Even though students can attend optional workshops regarding aforementioned thematic, the knowledge and experience gained are not sufficient for some common student smartphone based projects.

Suppose that there was an electrical engineering embedded system student project of a smartphone controlled garage door. To achieve this, student needs to be able to make adequate motor control circuitry and to program the embedded microcontroller – which often implies knowledge of C programming language and a certain set of libraries. Also, student needs to make a smartphone application capable of Bluetooth communication with the corresponding graphical interface.

If a student is studying a university module which does not cover all the areas needed for such a project, he needs to invest a significant amount of time to learn them. Sometimes, those areas are not something that students will need in their professional career. For instance – it is unlikely for a mechatronics student to need the knowledge of software development for mobile platforms.

This paper deals with the concrete issue of electrical engineering and mechatronics students who lack the mobile platform software development knowledge and experience, but want to make a project that requires it.

3. Possible solutions

3. *Moguća rješenja*

Solution to the problem mentioned in the previous section would be to outsource the mobile platform software development part to some other student. Outsourcing the part of a project to a student of some other university module often presents an administrative overkill for a small student project.

Another solution would be to use some third party mobile application for Bluetooth communication. Unfortunately, most of the free mobile applications offer basic functionality – sending and receiving character based messages. Also, these applications do not offer the possibility to adapt the graphical interface to the specific project.

Solution presented in this paper is based on a mobile application that has an ability to accept a formatted textual file as input and build a

graphical interface based upon the content of the file. Elements of the graphical interface can be used to send or receive messages via Bluetooth protocol. Flexibility of this solution lies in the fact that the file has a predefined format. Hence, to build a Bluetooth enabled mobile application with an arbitrary user interface, a file of a given format needs to be composed and passed on to the application. This application is valuable to students because students do not need to know how to program Android applications, but they can still make a smartphone controlled system. This process is explained thoroughly and through examples. It is also described how to connect with another device in the application via Bluetooth protocol.

This paper does not cover the implementation of Bluetooth protocol on the microcontroller side of a system, since that is usually not an issue to electrical engineering and mechatronics students.

4. Solution proposition

4. Prijedlog rješenja

Students who want to make a system that can be controlled using Android based smartphone over a Bluetooth protocol do not need to have any knowledge in Android mobile platform programming. They need application, as proposed in this paper, which can take as input a definition of a needed graphical interface. The application can pair the smartphone, over Bluetooth protocol, with another Bluetooth device (e.g. a microcontroller) and send it a message (identification) specific to the graphical component the user is interacting with. Identifications associated to different graphical components are also defined in the input file.

On the microcontroller side of the system, students can implement a Bluetooth based communication (e.g. using popular HC-05 module) and associate different actions based upon the message received from the smartphone. Overall product can be a system that can be controlled with an Android based smartphone.

Text-based, language-independent data interchange format known as JavaScript Object Notation (JSON) is used as a format of the input file for the proposed system. It is a lightweight, human-readable set of formatting rules for the portable representation of structured data [2].

4.1 Application layout definition

4.1 Defniranje grafičke strukture aplikacije

Visual structure of a graphical user interface is defined by a layout. Layout is a graphical element that provides a way of grouping other visual elements in a predefined manner. For example, Linear Layout groups elements in a vertical or a horizontal direction.

A certain layout is defined in the input file by a JSON object (defined in [2]) containing three name/value pairs:

- “views” – contains the JSON array of graphical elements (also known as View objects in Android platform development) that are contained in the defined layout [2].

```

{
  "views": [],
  "properties": [
    {
      "value": "up",
      "type": "string",
      "name": "text"
    },
    {
      "value": "1",
      "type": "int",
      "name": "id"
    }
  ],
  "widget": "Button"
},
{
  "views": [],
  "properties": [
    {
      "value": "2",
      "type": "int",
      "name": "id"
    }
  ]
},
{
  "views": [],
  "properties": [
    {
      "value": "true",
      "type": "boolean",
      "name": "clickable"
    },
    {
      "value": "100",
      "type": "int",
      "name": "max"
    },
    {
      "value": "3",
      "type": "int",
      "name": "id"
    }
  ],
  "widget": "SeekBar"
},
{
  "properties": [
    {
      "value": "vertical",
      "type": "string",
      "name": "orientation"
    }
  ],
  "widget": "LinearLayout"
}

```

Figure 1 Sample JSON file structure

Slika 1 Primjer strukture JSON datoteke

- **“properties”** – JSON array of JSON objects which define properties of the layout (height, width, color, etc.). Available properties for a certain layout can be found in.
- **“widget”** – a single string value defining the name of a layout, e.g.: LinearLayout, RelativeLayout, etc.

As mentioned above, each property can be defined with a JSON object. JSON property object contains three name/value pairs:

- **“value”** – represents the value of a property.
- **“type”** – string which defines the data type of a property (e.g.: string, int, float, dimen, color).
- **“name”** – defines the property that wants to be set (e.g. : layout_height, layout_width, orientation, gravity, id, etc.)

4.2 Supported graphical input controls

4.2 Podržane grafičke kontrole

Proposed system supports three different input controls: Button, ToggleButton and a SeekBar.

Each of these input control elements can be defined the same way as the JSON layout object. Unlike the JSON layout object, JSON input control object will have an empty “views” value (i.e. empty JSON array) – since the control elements usually do not contain other input control elements. Also, “properties” value of JSON input control object can contain different JSON property objects – depending on input control typ.

The most important property of an input control object in the proposed system is the id property. Id property is used for the identification of a control in the bluetooth communication which is explained in detail later in this section.

4.3 Example

4.3 Primjer

Essentially, a typical JSON input file contains one JSON layout object which subsequently contains multiple JSON input control objects inside its “views” name/value pair. Each of the JSON input control objects will be assigned a unique identification using the id property.

Figure 1 illustrates an example of a simple JSON input file defining a Button, ToggleButton and a SeekBar placed vertically

one below the other. Figure 2 depicts the produced graphical interface.

It can be seen in the Figure 2 that there are 4 additional graphical elements in the user interface which are not defined in the input file. The first element is a switch button at the top of graphical interface used for scanning and connecting to another Bluetooth enabled device. The second is a text area intended for displaying messages received via Bluetooth protocol (i.e. messages received from the microcontroller). There is, also, a text box and an additional button intended for sending arbitrary textual messages to the microcontroller.

For user to set his desired graphical interface layout, he needs to load a JSON file that he earlier created and saved on the local storage of his Android device. Choosing the “Load” option under the “Settings” menu the user will be able to see a screen as shown in Figure 3. After choosing the desired JSON file, application will load a new graphical interface or an error message will be displayed to user. Error can occur if the chosen file is not formatted correctly or the file cannot be opened.

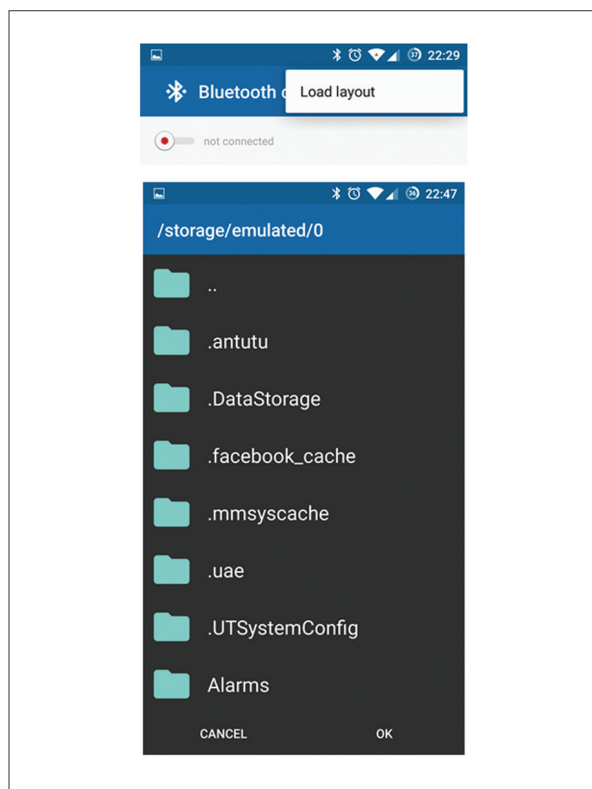


Figure 2. Loading layout screen

Slika 2. Prozor za odabir JSON datoteke

4.4 Mobile-to-machine communication

4.4 *Mobile-to-machine communication*

As mentioned earlier in this paper, this application can pair with another Bluetooth device and exchange information with it.

Pressing the switch in the upper left corner of the application displays a list of available Bluetooth devices to the user. Selecting one of the devices from the list establishes a connection to the device via Bluetooth protocol.

Also, as stated earlier in this section, each graphical control is assigned a unique id. Assuming that a connection to another device is established, by pressing a graphical control the aforementioned id of the pressed control is sent to the connected device along with the current state of the control. States of Button and a ToggleButton are 0 and 1. When Button is pressed application sends its id and a value 1 (as long as it is pressed), and 0 when it is released. When ToggleButton is pressed the application sends its id and a value 1 if it is on, and 0 if it is off. When SeekBar control is shifted application sends its id along with the current value of the SeekBar position.

It should be emphasized that by knowing the above stated protocol it is possible to realize a Mobile-to-microcontroller communication. At the microcontroller side it can be detected which control in the application was used by parsing each message received from the mobile device. Furthermore, an action corresponding to each control can be taken.

5. Possible future improvements

5. *Moguća buduća poboljšanja*

As already mentioned, proposed system is only intended for the Android platform based mobile

devices. Adaptation of the system to other mobile platforms should be realized.

A student using the system must create and populate the input file manually. Desirable upgrade would be to create an application for the input file production.

6. Conclusion

6. *Zaključak*

In recent years trends have changed in a way that more and more electrical engineering student projects require smartphone interaction with their targeted system. This paper covers the issue of electrical engineering and mechatronics students who lack the mobile platform software development knowledge and experience, but who want to realize a project that requires it. Solution presented in this paper is based on a mobile application that has an ability to accept a formatted textual file as input and build a graphical interface based upon the content of the file. Application is able to pair with another Bluetooth device and communicate via Bluetooth protocol. Every graphical interface element can be assigned a unique ID. Such a solution is flexible, since students do not need to know anything about mobile platform software development – just a predefined format of an input file. Also, this paper, explains in great detail how to create the aforementioned input JSON file, and how to connect to another Bluetooth device. This application significantly eases and shortens the realization of human to machine Bluetooth based communication.

7. Reference

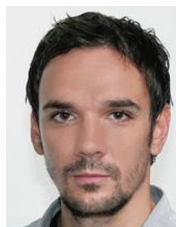
7. *References*

- [1] V. Galetić, I. Bojić, M. Kušek, G. Ježić, S. Dešić, D. Huljenić, “Basic principles of Machine-to-Machine communication and its impact on telecommunications industry” MIPRO 2011, Opatija, Croatia, pp. 380-384, May, 2011
- [2] D. Crockford, “The application/json Media Type for JavaScript Object Notation (JSON),” July 2006. [Online]. Available: <https://tools.ietf.org/html/rfc4627>.
- [3] “Layouts | Android Developers;,” 21 2 2016. [Online]. Available: <http://developer.android.com/guide/topics/ui/declaring-layout.html>. [Accessed 21 2 2016]

AUTORI · AUTHORS**Petar Zubak**

Petar Zubak rođen je u Zagrebu 03.05.1989 godine. Osnovnu i srednu školu pohađao u Zagrebu. Preddiplomski studij završio na Tehničkom veleučilištu u Zagrebu 2016. godine, smjera

Komunikacijska i računalna tehnika. Tokom studija sudjeluje na natjecanju u izradi mobilnih aplikacija MC2 u sklopu Tehničkog veleučilišta u Zagrebu. Tu osvaja 2. mjesto, i kao rezultat toga od 2014. godine, uz fakultet počinje raditi u Styriji kao Android developer. 2016. godine, završava studij te počinje raditi u Agenciji 404.

**Tomislav Novak**

Tomislav Novak rođen je u Zagrebu 22.07.1985. godine. Osnovnu i srednu školu pohađao u Zagrebu. Preddiplomski i diplomski studij završio na Fakultetu elektrotehnike

i računarstva u Zagrebu 2010. godine, smjer Telekomunikacije i informatika. Radi kao predavač na Tehničkom veleučilištu u Zagrebu od 2011. godine. Od 2012. godine honorarno radi kao instruktor objektno orijentiranog programiranja i Jave na NetAkademiji.

Korespodencija

tnovak@tvz.hr

**Stipe Predanić**

Stipe Predanić, dipl.ing. rođen 1980. godine u Šibeniku, diplomirao je 2005. na Fakultetu elektrotehnike i računarstva u Zagrebu. Nakon prvih poslova kao web developer, od 2006. radi

na Elektrotehničkom odjelu Tehničkog veleučilišta u Zagrebu. Najprije kao asistent, a kasnije kao predavač i viši predavač iz područja digitalnih sklopova, programiranja, razvoja web sustava te ugradbenih računalnih sustava. Uz to je i instruktor na NetAkademiji za Cisco CCNA program, a posjeduje i licencu za Mikrotik Academy trenera. U slobodno vrijeme bavi se rekreativnim trčanjem i radioamaterizmom, osobito radiogoniometrijom.

Korespodencija

spredanic@tvz.hr