Fostering Engineering Ingenuity and Self-determination in Computer Science Undergraduate Students

Sorin Ilie, Alexandru Mitrica, Alecsandru Sperila, Cristian Totolin, Marius Neghina and Marcu Marian
University of Craiova
Department of Computers and Information Technology
Blvd. Decebal nr. 107, Craiova, Romania
ilieviorelsorin@gmail.com

Abstract

Computer Science undergraduate students arrive in the first year with great expectations of themselves. However they soon realize that being the next disruptive innovator takes a lot of knowledge, inspiration and hard work. This realization chips away at their determination until they accept their fate as just another developer. Last year we conducted an experiment in maintaining student enthusiasm for innovation, aimed at first year students. We created multiple teams, lured them with easily programmable Arduino projects, then asked them to make the projects work together and finally implement an android app to control those projects. This generated a complex skill set for the students and fostered team work and self determination towards innovation. One such project was the Smart Hotel project, awarded with three international prizes, had 5 public presentations, was described in multiple interviews on TV and radio and is being updated with pro-activity and artificial intelligence.

1. Introduction

The engineering–inclined students love to tinker with hands-on projects. Taking advantage of this fact for educational purposes became much easier when we received a sponsorship from Intel: 20 arduino-compatible microcomputers. We asked for volunteers and 7 students showed up eager to play with the devices: 6 first year students and one senior, all having minimal programming and networking knowledge.
The first order of business was to find one or more projects that the students would be happy to work on, so we organised a brainstorming session. After one week of individual research, all students proposed small projects in the domain of home automation.
They organised themselves in teams of two. Within a week they were presented with their own arduino-compatible device, a breadboard and a variety of sensors, actuators and basic electric parts.
The next month, students worked hard on their applications showing a great deal of self-determination but still needed the presence of someone with more experience to guide them. Soon we realised they find the solutions they are looking for themselves so we stopped trying to help them but used the “granny teaching method” presented here by professor Sugata Mitra [17,18]. By using this method, the “teacher” is supposed to constantly ask questions then praise the student when he solves the problem.
Once hooked on experimenting with their devices we presented them with the challenge of merging the projects by using Ethernet connections, web services and web interfaces.
The student’s work envisions a new way of interacting with hotel rooms by simply interacting with a mobile app that can control door locks, lighting, plugs etc. Access can be given to guest-users and maintenance staff to a room for a given amount of time, even at predefined intervals. At all times the server can log all door accesses, presence, power consumption, and so on for inspection.
To summarize, the teaching strategy was: i) attracting students with fun toys, ii) allowing them to set the initial goals (self determination), iii) high support with planning and initial implementation (leadership through example), iv) a daunting challenge meant to improve teamwork (purpose), v) tutorials about new technologies like web services, android, object oriented programming, versioning, networking (self improvement) vi) validation through prizes.

2 Research phase

The students were asked to find similar approaches, tutorials and help with the component shopping. They found the following:
Philips's Hue Connected Bulb[1] uses a central module that plugs directly into a wireless router and translates signals between a Wi-Fi-connected smartphone and the

In: A. Bădică, M. Colhon (eds.): Proceedings of the 2015 Balkan Conference on Informatics: Advances in ICT

bulbs. The bulbs cost from $79 to $89 a piece and a kit containing one hub and three bulbs costs $199.
The Belkin WeMo Switch + Motion[2] is a relatively low price device which has a plug-and-play configuration that will appeal to many budding home automations. Belkin's WeMo Switch+ controls your plugs via Wi-Fi.
SmartThings[3] is another approach to a smart environment, they use a hub connected to a router which can control multiple modules. The modules connect to the hub via Wi-Fi and are controlled using an Android or iOS device.
Smart Room Control Solution[4] from Distech Controls is a different approach to the smart room problem. They use different modules which require a complicate installation compared to other solutions. The wiring is complicated and modules have to be carefully mounted on the fuse box.
Control4[5] from Exzel Smart Home, is an award-winning product that can help homeowners manage their home in a smart and integrated way. From turning music on, choosing movies, to controlling the lighting, setting room temperature, controlling sprinklers, monitoring the front door camera to see if anybody is there, setting and monitoring fire alarms and security alarm systems. All of this can be done by one single user interface; be it a touch screen panel, iPhone, iPad, iPod, or universal remote control.
At the end of the research, phase two of the 7 students quit. One realised that he is not happy building such devices and made a career change towards medicine the next year. The other felt he was slowing down his teammate so we directed him towards tutorials and invited him back when he feels he is prepared.

3 Design and implementation

The challenge was to provide home automation that is available as a website (see the high level functionality image in Figure 1). This allows the hotel staff to grant access and control of a room to a guest and manage the room statistics and logs. When granted access, a guest can use any and all room functionality until the end of his/her stay directly from the mobile device.

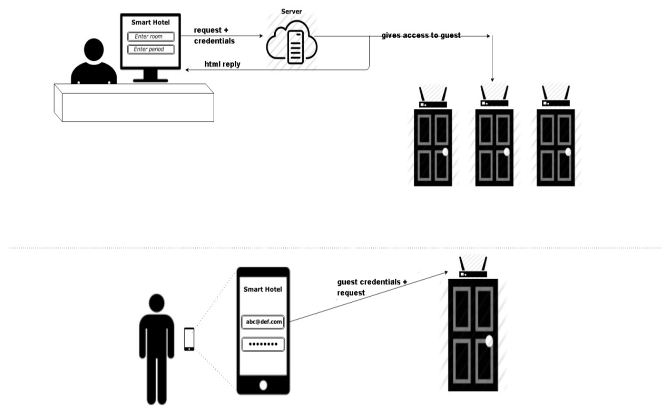

Figure 1: Giving control of a room to a registered user

3.1 Architecture

The system we designed for the students was composed of a central server, one embedded controller per hotel room, one mobile device for each staff member or guest and interconnecting Ethernet or Wi-Fi networks.
The embedded hotel room controller implements all interactions with the physical elements of the room: power plugs, lighting, door lock, motion sensor (see figure 2 for the block diagram). The hotel server communicates via Ethernet with all controllers using basic socket communication, but also waits for requests from mobile devices and reports from room controllers. This is done via a Room Management Module. All functionality of this module as well as administrative functionality and user profile management is offered to other applications using REST web services [16].

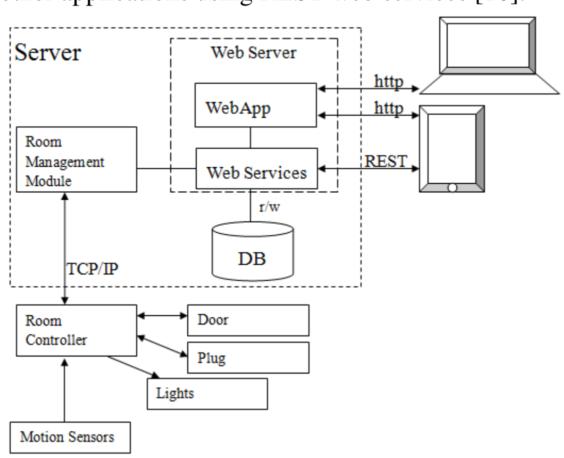

Figure 2: System architecture

3.2 Android mobile device

The personal mobile device is used to identify the user and exercise control over the assigned room in a meaningful way to the guest or staff member. We chose to implement this initial version of the project using Android mobile devices[11] and the Java programming language [14].

One of the issues we had to tackle was the security of login over a Wi-Fi connection. We approached this challenge by implementing an access token with an expiration date. The token t obfuscates the actual text password p using the SHA1 algorithm[12], and incorporates the time of the access request t0 using an internet time service. In the following equation an access token t is generated from p and t_0.

$$t = SHA1\ (SHA1(p),\ t0) \qquad (1)$$

Access token t is only valid x seconds after creation after which the server will not recognize it any longer. The server can determine if the token is expired because t0 is attached to the command along with the token, username and command. The server can then use the hash of the user's password from the database and the time to generate a token. This type of token eliminates sniffing attacks.

With this info one of the students had to implement the whole Android application. The student was provided with a basic tutorial including a live coding session.

3.3 Server

This software module runs on a physical server and is the central entity of the system. The software baseline on which we built this module is: Windows 8 operating system [7], a Tomcat [10] webserver and a MySQL database server [6] all running on the same machine.

The rooms and users are entries in the database, and the Rooms and Rent servlets are used to create the web interface needed to assign control of a room to a guest. DbManager class controls and populates the database access object classes RoomsDAO and UserDAO. UserLogin is the login page for the web application. It uses class PassCheck to verify the validity of a password. Services Login and Register, implemented as servlets, are necessary for the Android application to invoke functionality.

The HotelDriver class implements the protocol used to interact with the room controller (Arduino-compatible module[8]). Every method of the HotelDriver is offered as a service to the mobile application after adding a user authentication layer. We also added a subscription service on the controller for the motion sensor allowing the room controller to provide notifications when movement is detected in the room.

Two students were given this description and a tutorial including a bit of live coding and were asked to build a working service-based app.

3.4 Room Controller

The room controller is implemented on an Intel Galileo (Arduino-compatible) module [8]. The operating system on which we built this software was Yocto Linux[9]. We implemented an Ethernet server that communicates with the web server using an Ethernet. Each room must be assigned a static IP.

We can now outline a schematic of the interaction protocol between the mobile device, the server and the room controller (see figure 3 for the sequence diagram[13]). We identify three types of interactions: i) mobile->server interactions, ii) mobile->room controller interactions via server and iii) room controller -> server interactions.

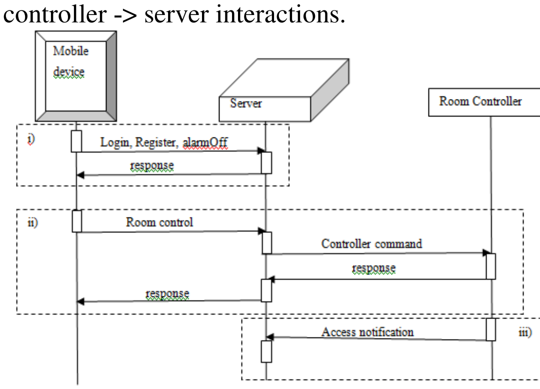

Figure 3: Interactions Schematic

4 Hardware room controls

The initial projects implemented by the students were the room controllers for lights power sockets and access.

4.1 Door

Pressing the "Open Door" button in the mobile app invokes to the central server via Wi-Fi which in turn sends an Ethernet packet to the room controller that opens the door.

We built an automatic door that also has a normal metal key in case of blackouts (see figure 4). We used a relay to operate the door. We also connected a 5V output port on the door lock and an input port on the door frame so we can detect when the door is shut.

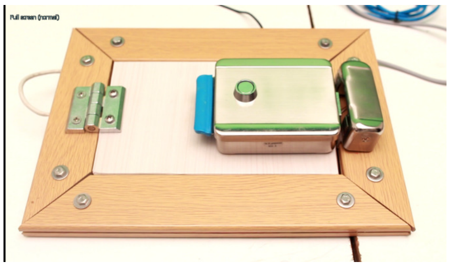

Figure 4: Demo door

4.2 RGB Lighting

RGB LEDs can emit any color of light depending on three analog command circuits. This allows the user to set the mood in the hotel room as he desires. The user can choose any color using the android app where he sets 3 different values for the Red, Green and Blue hues. The preferred colors can also be saved on the Server. This command can be given via internet from virtually anywhere.

4.3 Plug

The smart plug functionality allows the user to turn a device connected to it on or off by pressing the SmartPlug button on the Android app. The app will send a message to the central server via Wi-Fi which will send a message to the Controller and activate or deactivate the relay. By using a current sensor the SmartPlug can read how much current is being used. In order to control the power plug we use a relay much like the ones used for the door. Both the relay and the sensor are connected in series with the power plug.

4.4 Motion Sensor

Movement detection sets off a notification. The room controller will read input from the PIR and forward it to the server via the alarm subscription service when presence is detected so that the server can log it.

5.Collaboration

The students used Google Docs to synchronize tasks and module inputs and outputs. In the beginning most interactions were done through their advisory professor. But in just one month of working this way they self organised using Trello1 as a work management platform, git for code versioning. They also realised that I was actually using “granny teaching method” on them without really providing any explicit answers so they instinctively moved to rubber ducky and pair debugging. In this programming approach they explain their code to each other or to imaginary partners until they find the problem. All students were required to attend each tutorial even if it had no immediate benefit to their portion of the project for collaboration purposes.

Students stated that that most of the initial motivation was generated by the fact that they believed in the project initially and this was fostered by financial and educational support from their teachers. They also state that their enthusiasm only grew when they were faced with seemingly insurmountable tasks that they successfully completed: databases, object oriented programming, mobile computing, web applications, networking, embedded computing and electronics. They loved becoming better problem solvers. They enjoyed cannibalizing old electronics to find the parts they needed.

This team expected their teachers to be all knowing at the beginning soon they realised that we have limitations. But they still needed a mentor present at all times in the room.

The project investment was about 500 euros and the teacher time spent on this project alone was about one and a half man-months.

6..Outcomes

The project was awarded third place in the ESDC[2] international contest (Shanghai China), third place in the national contest IMSmart[3] (Bucharest Romania) and fourth place in the international BringITon[4] workshop (Iasi Romania). The Smart Hotel project was also presented in the Intel Romania 10 year anniversary

[1] Trello task management system https://trello.com/

[2] ESDC 2014 international contest http://nuedc.sjtu.edu.cn/EN/show.aspx?info_lb=20&info_id=66&flag=20

[3] IMSmart 2014 national contest http://imsmart.ro/

[4] BringITon Iasi 2014 http://bringiton.info.uaic.ro/en/

event, and Researcher Night in the University of Craiova. One of the students also won first prize in a locally organised embedded contest “how to electronics”.

After the first contest was won one of the students accepted an internship and started rejecting further tasks on the principle of lack of personal support and enterprise-like efficiency. He immediately became a part time employee at a local software firm.

7. Future Work

There are two directions to be analyzed: the future work of the team members and the future development of this hands-on way of educating students.

8. Student team

The future work on the Smart Hotel project consists of: voice operation, modularization for private use and proactive artificial intelligence. Since the last public presentation the project has already become voice operated by invoking the services via open source voice recognition software, however, it is unclear if this is more useful than the mobile app. Modularization poses challenges on self-organization for easy configuration. While the proactive artificial intelligence is meant to automatically adjust actuators to reach a preferred environment status.

The next step in the students’ training is slowly removing teachers from the equation in order to generate self reliance. The students no longer need teacher presence to get tasks done but they require it in order to actually attempt to start working. For that reason we started limiting teacher interaction to one short scrum meeting a week. Most of the tasks are assigned and reported on using Trello.

Premature workforce integration of the students seems detrimental to their studies. We generally see a decrease in time and effort dedicated to tasks when students start working, even part time. For this reason we created connections with several private sector organizations with real life problems that need to be solved and are willing to support the students financially for their efforts. This way we can make sure they keep improving their knowledge base and also become self sufficient.

9. Education style

During this experiment we identified a few problems with the educational process: not all students are made to be computer scientists and the sooner they find out if they like it, the better, the topic of computer science is ever changing to the point that it takes a very long time to update contents of teaching materials, students must be included in projects that are worthwhile for them then supported in implementing them.

10. Conclusion

This experiment had the purpose of turning students into programmers in the shortest possible time using hands-on challenges. The skills they learned were applied in the spirit of self-determination and entrepreneurship. Soon they were approached by prospective buyers and human resource representatives of local programming firms. All the students accepted internships but half of them also work on their own projects such as artificial intelligence for home automation, indoor localization systems and easy to program embedded systems. The programming games and tutorials found by the students and reviewed by their mentors are now part of the optional bonus point challenges presented to first year students.

11. Acknowledgement

This work was partially supported by the strategic grant POSDRU/159/1.5/2/133255. Project ID 133225(2014), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007- 2013.

References

[1] Philips's Hue reference, accessed February 2015 http://www.usa.philips.com/e/hue/hue.html

[2] The Belkin WeMo Switch + Motion reference , accessed February 2015 http://www.belkin.com/us/p/P-F5Z0340/

[3] SmartThings reference, February 2015 http://www.smartthings.com/

[4] Smart Room Control Solution by Distech Controls reference, accessed February 2015 http://www.distech-controls.com/en/csa/products/smart-room-control-solution/

[5] Control4 from Exzel Smart Home reference, accessed February 2015

http://www.exzelsmarthome.com/products/homecontrol/index.html

[6] MySql refference manual http://dev.mysql.com/doc/refman/5.7/en/ accessed June 2014

[7] Windows 8 User manual , accessed June 2014, http://www.microsoft.com/en-us/download/details.aspx?id=39055

[8] Intel Galieo online manual, accessed June 2014 http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html

[9] Yocto Linux documentation, accessed June 2014 https://www.yoctoproject.org/documentation

[10] Apache Tomcat Server documentation http://tomcat.apache.org/tomcat-7.0-doc/

[11] Android Developer Manual http://developer.android.com/ , accessed June 2014

[12] Schneier on Security: Cryptanalysis of SHA-1, accessrd June 2014 https://www.schneier.com/blog/archives/2005/02/cryptanalysis_o.html

[13] Unified Modeling Language User Guide, The (2 ed.). Addison-Wesley. 2005. p. 496. ISBN 0321267974.

[14] Ian F. Darwin : Java Cookbook, 3rd Edition, ISBN 978-1-4493-3704-9, publisher O'Reilly Media, 2014

[15] Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson. 2001.Introduction to Algorithms (2nd ed.). McGraw-Hill Higher Education.

[16] Leonard Richardson, Sam Ruby: RESTful Web Services Web services for the real world, ISBN: 978-0-596-52926-0, publisher O'Reilly Media, 2007

[17] Mitra S: Beyond the Hole in the Wall , Technologi Education Design talk publication Book: https://books.google.ro/books?id=GheifdC3VjUC, Presentation: http://www.ted.com/talks/sugata_mitra_shows_how_kids_teach_themselves

[18] Mitra, S.: The Hole in the Wall: Self-Organising Systems in Education, isbn 9780070617872, 2006, https://books.google.ro/books?id=2cjaAAAAMAAJ