Enchanting Education from Student Input

Preparing students to envision and develop in an Internet of Things world

Torben Svane, Zhu Minling
Computer School
Beijing Information Science and Technology University
Beijing, P.R. China
torben@svane.se, zhuminling@bistu.edu.cn

Lars-Olof Johansson, Esbjörn Ebbesson
School of Information Technology
Halmstad Univeristy
Halmstad, Sweden
lars-olof.johansson@hh.se, esbjorn.ebbesson@hh.se

***Abstract*—Searching for "The Internet of Things" may render thousands of hits in academic databases but a challenge still remains: to let students envision as well as work with the concept in a practical way. Designing "Things for the Internet" will not only require skills in engineering and electronics but also some understanding of digital service design and business models. This paper reports on work in progress between Belgium, Sweden and China. Students with different education backgrounds and levels hand over work (ideas) to others, to develop further. Currently, there are nine open-platform exercises to use as starting points for student work and discussions. Exercises are in English and free to download and use, as are some of the lectures related to them. So far, they are used in a digital service design class and in an introductory course in embedded systems.**

Keywords—Internet of Things, product design, service design, embedded systems, student engagement, international cooperation, Ardiuno, classroom exercises, enchanted objects, SMILE, Bloom, Belgium, China, Sweden.

I. Introduction

Creating new courses which enable crossovers between different subjects, different study levels and even different continents brings around many challenges. Formulating texts, enabling both overview in understanding and a possibility to investigate in great detail (but without the explicit need to do so) are but some issues which must be dealt with. This paper reports on work initiated from course development around the Internet of things (IoT) concept [1] with the aim to (a) make the notion very concrete through students' hands-on experiences and (b) point at the variety of skills needed in this development.

A third goal was to let students create input to others with a different background [2]. This has at least in part exposed them to a possible future scenario of project work in an increasingly international environment.

II. The classes: a breif overview

The Belgian students who do projects can have a variety of assignments, in Belgium and internationally with a traditional setup of workplace experiences. They are assessed on how well they have completed their tasks, and write reports on their work and experiences. The Thomas More university college sends out numerous students on such projects; it is their final assignment before graduation but not as such part of this design.

The Swedish class is called "Physical materials and IT" and was initially given to final-year students in the Digital Design and Innovation program (in Information Systems) at Halmstad University. The idea of the class is to move beyond systems and let the students have a hands-on experience of actual, although simple, electronics assembly. The aim is to understand in greater detail how such technologies work and then use knowledge from lectures and previous classes "to look at technology and extrapolate ideas". Discussions will deal with new ways to use the tested technologies – and new potential services which could stem from them. Student experiences are later used in a (graded) retrospective paper in the class.

The Chinese class is an introductory class in embedded systems, mainly targeting graduate students. "Embedded systems" is here seen as a foundation theory, a theoretical base to aid students in discussions and problem solving. The course combines theory with engineering practice and plays an important role in the computer and electronic information class curriculum system at Beijing Information Science and Technology University.

The class has a strong practicality orientation so that students have to combine comprehensive design abilities of software and hardware simultaneously, which many times is seen "difficult" by students. The setup engages students in a project-driven teaching method, where the projects (in teaching) all rely on a strong application background. At the same time, the teaching methods promote progression in terms of an ability to complete project implementations, as explained below. Adding external input will contribute further to this progression.

For the students, the whole project is firstly analyzed in detail ("decomposed"). This will lead to a set of subprojects, where some knowledge components are generated. Then, different concepts and knowledge areas required are introduced, from an overview to a deeper understanding. As for all subprojects (the components), an atomistic-to-holistic process is executed – from the part to the whole – until each project is completed. This final phase – the fourth layer, presented in the next section – could add useful input to the Swedish students, at that time in their final (undergraduate) year.

III. Work

The development started in the spring of 2014, when the Swedish university wanted a Belgian project student to create a number of exercises to be used in a design class for Swedish

978-1-5090-5920-1/17/$31.00 ©2017 IEEE

final-year students the coming autumn. The exercises turned out very well and were later discussed with colleagues in China. In that discussion, the idea came up to use Swedish student reports as input in China and later use Chinese feedback in Sweden.

The design has hence developed over time as new ideas have been introduced but in that process, pedagogical discussions have also played an important role for creating a sound, meaningful and theoretically aware arrangement.

In trying to visualize the concept, a snowball metaphor was introduced. Snowballs become bigger as more layers are added. In the same manner, the initial exercises "grow" as new ideas and new perspectives (from other disciplines) are put on.

A. The first layer: creating exercises

The entire process starts when Belgian final-year (undergraduate) students in ICT and electronics arrive for their spring project placement in Sweden. One student is assigned to work on new exercises aiming at demonstrating additional possibilities of IoT services/interaction. Both physical and logical functions (e.g. audio, timers) as well as interaction issues addressing SMILE [3] technologies are discussed. The student may also suggest own exercises. Suggestions from other teachers and students at the three institutions are also welcome.

B. The second layer: testing and thinking

When new exercises are tested and validated (electronically and for text clarity) the Arduino™ workshop "gadget boxes" are prepared. The kits contain equipment/parts needed to assemble, test and modify a specific exercise. Swedish students studying digital design and innovation will then use some of the boxes in their class the next semester. In their exercises, they investigate functionality, change code to observe different effects and discuss how new digital services and business models may be applied. They are examined not on handling the electronic setups *per se* but on discussions (in written reports) how the tested functions could be used in other applications and services.

C. The third layer: cross-continent, -level, -discipline

The next step is carried out in China where graduate students in an embedded systems class can discuss system architecture and design setups stemming from the Swedish students' ideas (with an option to add further to the concepts). Here, the challenges will focus more on programming and on how to actually construct devices whereas work in Sweden uses off-the-shelf electronics which can be purchased in most hobby stores. The Swedish students are also provided with programming code.

Another challenge for the graduate students is found in the international and interdisciplinary dimension: input reports are all in English, by non-engineers. Taking on ideas from others (with a very different background) is part of the teaching design and resembles well what students may face after graduation.

D. The fourth layer: feedback, in later stages

At time of writing, a second cycle of development has started. The first batch of exercises have been used in Sweden and distributed in China. Swedish student (idea) papers on how the technology tested could be packaged into new services are finished. The exercises have been discussed in China for the embedded systems class (with Chinese and Swedish teachers). An idea which emancipated from a lecture on IoT and business models to other (non-engineering) Chinese students has now turned into a new exercise, as part of new packages for 2016.

The final feedback step is not fully implemented yet, as explained in the design section. The timing will be better synchronized between Sweden and China with both classes held in in the autumn, as Swedish students will still be around when Chinese students can give feedback on their ideas.

E. Future developments

The 2015/2016 exercises formed the base for Swedish students in autumn 2016. There will not be any new exercises for 2017 as there is not any Belgian project student available. The ambition is now to add additional exercises in 2018. Some more "gadget boxes" will increase the options (for the Swedish students) to decide on what technologies to investigate in more detail and hence let interests guide their choices. In that process, it will of course also bring new challenges to the Chinese students – and indeed, to all teachers involved.

IV. EXAMPLES

This section will give three examples of exercises through the first phases. As paper length is restricted, descriptions will be fairly short for examples B and C. A list of links to free-to-use exercises and lectures is available for download [4].

The Swedish applications will usually start from a practical problem (to be tested with the Arduino "gadget boxes") whereas Chinese exercises may start from solving more underlying problems (e.g. in programming).

A. Using timers

The timer exercise is combining a speech recognition app which via Bluetooth sends information to Arduino (which in turn displays the timing on a display). A second exercise (simulating a microwave oven) reads the timer setting and finds a suitable music file which will play for that time. This will keep the user better informed of the elapsed time (compared to the traditional silence for e.g. 2 minutes, and then a sound).

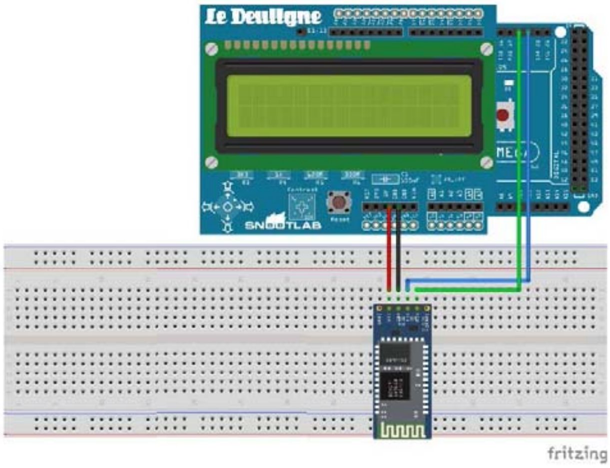
The instructions have been compressed here, but the text is complete. Instructions are better "pedagogically formatted" in the real exercises (but thereby, also more space-consuming).

1. SETTING UP THE PROJECT
Parts you will need:

Arduino Mega, USB type A to B cable, LCD screen Hitachi 2x16, HC-06 Bluetooth module, Breadboard, assorted cables, male-female wired.
Software needed: Arduino IDE.
Set up the hardware following the displayed schematic (on next page).

2. SETTING UP AND STARTING THE SOFTWARE

Click on "File" then "Open". Navigate to the .info file and open it. Once it is open it's time to upload the software to the Arduino. First plug in the device and give it a few seconds. Now you can open the Serial Monitor. Do so by clicking on the magnifying glass on the right side of the program. After this opens you can upload the software to the device by clicking the right pointing arrow on the left of the program.



After compilation and upload the software starts running by itself. If you get an error that tells you that the COM port is unavailable then close the Serial Monitor. Unplug and plug the device back in. This should resolve the issue. Study the software to get an idea of what is happening. The comments should help you along the way. If you don't understand a command: Google is your friend!

3 CONTROLLING THE PROJECT
The Voice Timer is very easy to control. First install the AMR_Voice app on your Android device. Use this link to get it:

https://play.google.com/store/apps/details?id=robotspace.simplelabs.amr_voice

Open the app and give it a second to turn your Bluetooth on. Once this is done, open the context menu and press "Connect Robot". Wait until you see "HC-06" appear. Select this connection and use "1234" as PIN. Once you are connected you can press the microphone button and say your sentence. The sentence should have the form of "[Device name] [Timer name] 00 minutes 00 seconds". Timer name is "Timer" Some examples would be:

• Timer potatoes 10 minutes
• Timer 10 minutes 20 seconds
• Timer rice 1 minute 2 seconds
• Timer 80 seconds

For troubleshooting it might be useful to type the command. You do this in the setup function. The command then runs as soon as the program starts. It would look something like this:

```
searchSentence ("*timer potatoes 10 minutes#")
```

Remember the special characters at the start and end of the sentence!

Voice control, for timers and other uses, triggered many ideas in the Swedish reports. Egg timers and "reminder" units which would use audio to remind the user, and setting remote timers e.g. to heat up the car in winter via a text message to the mobile-connected car heater were some.

Timer programming is often one of the first exercises to do on e.g. TI devices. Combined with assignments to connect input and output devices, more advanced timer projects (and coding) would be useful in China, and helpful as feedback in Sweden.

B. *Using thresholds*

The Swedish exercise uses a pulse sensor and an MP3 shield to play audio (a voice saying "relax…"). The scenario is a device, which will help people relax if their pulse gets too high. A similar exercise was meant to mimic music selections during skiing; when you go down-hill your pulse is fast and you want music with a high BPM (beats per minute). When in the ski-lift going up again, music should be slower as the pulse is down.

In reports, Swedish students suggested many alternatives to the use of pulse sensing, e.g. in pens during tests, or in cars, if the driver becomes too relaxed (indicating a risk for sleeping).

Programming thresholds and triggering related actions may be simple for students in embedded systems but with plenty of real-world uses to relate to, they may become more relevant and inspirational for further discussing different uses/applications.

C. *Using RFID devices*

Inspired by an MIT project from 1999 [5] students use an RFID reader to light up LEDs (depending on the specific RFID tag), and play audio from an SD card. The exercise aims at widening students' understanding of (in particular) RF technologies and (at least, for beginners) more complex technical setups.

Among the suggestions are "book cards" which will read a book/bedtime story, and ideas which will be turned into a new exercise next year: a "shopper automation" setup, inspired by what is now known as *amazon go* [6]. This exercise will also involve teachers who teach marketing at Halmstad University.

This setup combines many technologies not available with the TI devices but can be simulated graphically on displays. The Chinese university has now also acquired a set of the used Arduino devices, which will enable their students to explore further on setups and ideas from Swedish input when relevant.

V. TEACHING DESIGN PERSPECTIVES

Creating an arrangement which ranges over multiple continents, disciplines and levels will not only call for long-time contacts but also for common beliefs in teaching and student delivery. Much of the setup has been inspired by concepts such as Bloom's taxonomy and Problem-based learning (PBL). Both frameworks acknowledge the importance of student engagement on many levels and (as with PBL) an often-showed enthusiasm for real-world problem solving; important goals in any engineering or creative discipline. China also looks into CDIO (see *C*).

A. *Bloom's revised taxonomy*

Bloom's learning framework [7] (revised by Krathwohl [8]) consists of two dimensions (knowledge and cognitive process). The separation guide educators in creating instructions and assessments. In the knowledge dimension, there are four categories: factual, conceptual, procedural, and metacognitive knowledge. The separation between conceptual and procedural knowledge is especially interesting as it highlights "skills and abilities". The cognitive process dimension has six categories: to remember, understand, apply, analyze, evaluate and create. These categories are applicable to both conceptual and procedural knowledge. The snowball design addresses both cognitive processes and knowledge categories. For the Swedish students, learning is manifested in the doing and duplication of what is written in the workshop exercises, which leads to an understanding of concepts (how technology can be applied). From these exercises, students apply what they have learned, regarding procedural knowledge (change parameters/programming), and analyze the outcomes.

They also evaluate possibilities of the technologies and discuss new designs/services which could use the tested functions.

B. Problem-based learning (PBL)

From early works [9] to recent reports [10], problem-based learning (PBL) has been a useful approach to action-oriented and student-active learning. As a student-centered pedagogy it encourages students to learn about a subject through the experience of solving open-ended problems found in trigger materials.

The snowball design draws upon ideas from PBL, both in terms of the delivery (input) of open-ended problems in given trigger materials (the workshop cases for the Swedish students and the reports which later can be used in China) and through an action-driven approach to exploring problems and suggesting various solutions (rather than merely delivering answers).

The setup is by no means "pure" PBL. The Swedish class also includes lectures unrelated to engineering, information systems or design, e.g. on business models [11] and service blueprints [12] and calls for student engagement in course content, following KBUD [13], Knowledge By User Demand.

C. Conceive-Design-Implement-Operate (CDIO)

Originally an MIT initiative, the CDIO network of educational institutions has spread around the globe. Although neither of the classes involved in the snowball setup uses a CDIO-style syllabi [14], many of the ideas in that template (e.g. technical knowledge and reasoning, personal and professional skills, and interpersonal skills) are addressed in, and between, classes. PBL and CDIO are sometimes seen as two quite different approaches but studies have also shown how the two combined can be complements and reinforce engineering education [15].

D. Other sources of inspiration

The importance of training students to work in international teams has been evaluated in many studies. An ability to work in teams, together with well-developed communication skills both at an international and intercultural level are among such skills being investigated [16]. A revised ACM/AIS model curriculum for undergraduate degrees in Information Systems [17] likewise acknowledges such needs. Cross-disciplinary designs, including student work between different subjects have also been reported [18]. A more theoretical paper (taking a starting point in social constructivism and reflective learning) [19] used *Cooperation technology* as a theme for designing multi-faceted tasks and outputs in "expanding" group settings (from course groups to local-community and finally, to international groups).

In the sources mentioned, there is however not any bridging between undergraduate and graduate levels. The subjects are also more closely related, whereas the design presented in this paper is between behavioral science and engineering students.

VI. DESIGN CHANGES

During the two years when steps gradually have been implemented, changes have made for a better fit between groups, and better synchronization for feedback. The fourth phase is not in place yet, as course times have been changed, in Sweden and in China. Autumn 2017 will see the first cycle of the final phase.

VII. CONCLUSIONS

The design where students deliver input to entirely different groups will be further explored. There are interesting challenges to all involved and the approach has already triggered creative ideas with commercial potential – in a time where the outlooks for IoT business opportunities indeed seems promising [20]. As with most testing of new arrangements, there are still a number of challenges to face. The two most central are a) the synchronization of classes (what time of year) and b) the evaluation of future feedback and possible cooperation in communication from China to Sweden (the final feedback loop).

REFERENCES

[1] D. Rose, Enchanted Objects: Innovation, Design, and the Future of Technology, 2015, Scribner Book Company, New York.

[2] D. Boud, R. Cohen, and J. Sampson, Learning from & with each other, 2001, Kogan Page Ltd., London, U.K.

[3] T. Svane, "Get your students to SMILE! Exploring emerging interaction technologies", IEEE ITHET 2013 (Antalya, Turkey), pp. 1-6, DOI 10.1109/ITHET.2013.6671064.

[4] Download links from http://hh.mywr.net/FIE2017/FIE2017_LINKS.pdf.

[5] Project is found at http://tangible.media.mit.edu/project/musicbottles/.

[6] Info URL: https://www.amazon.com/b?ie=UTF8&node=16008589011.

[7] A. Nunes Rodrigues, and S. C. dos Santos, "A systems approach to managing learning based on Bloom's revised taxonomy to support student assessment in PBL", IEEE FIE 2013, DOI 10.1109/FIE.2013.6684978.

[8] D. R. Krathwohl, "A revision of Bloom's taxonomy: an overview", Theory into practice, 41:4, Revising Bloom's taxonomy, autumn 2002, pp.212-218.

[9] J. R.Savery, and T. M. Duffy, "Problem based learning: An instructional model and its constructivist framework", Educational Technology, 35:5 (Sept/Oct 1995).

[10] P. Dolog, L. Leth Thomsen, and B. Thomsen, "Assessing problem-based learning in a software engineering curriculum using Bloom's taxonomy and the IEEE software engineering body of knowledge", ACM Trans. Comput. Educ. 16:3, May 2016, DOI: http://dx.doi.org/10.1145/2845091.

[11] A. Osterwalder, and Y. Pigneur, Business model generation: a handbook for visionaries, game changers, and challengers, 2010, J Wiley & Sons, Hoboken, NJ, USA.

[12] S. Fließ, and M. Kleinaltenkamp, "Blueprinting the service company", Journal of Business Research, 57:4, pp. 392-404, 2004. DOI 10.1016/S0148-2963(02)00273-4.

[13] T. Svane, C. Aderklou, L. Fritzdorf, and J. Hamilton-Jones, "Knowledge by user demand and self-reflection: new models for teaching and assessment in edutainment software design", IEEE FIE 2001 (Reno, NV, USA), DOI 10.1109/FIE.2001.963834.

[14] F. Crawley, "Creating the CDIO syllabus, a universal template for engineering education", IEEE FIE 2002 (Piscataway, NJ, USA), pp. F3F-8 - F3F-13 (vol.2), DOI 10.1109/FIE.2002.1158202.

[15] K. Edström, and A. Kolmos, "PBL and CDIO: complementary models for engineering education development", European Journal of Engineering Education, 39:5, pp. 539-555, 2014, DOI 10.1080/03043797.2014.895703.

[16] N. Escudeiro, P. Escudeiro, A. Barata, and C. Lobo, "Enhancing students team work and communication skills in international settings", IEEE ITHET 2011 (Izmir, Turkey), DOI 10.1109/ITHET.2011.6018683.

[17] ACM/AIS 2010 curriculum can be downloaded from http://www.acm.org/education/curricula/IS%202010%20ACM%20final.pdf (accessed 170426).

[18] J. Polutnik, M. Druzovec, and T. Welzer, "Interdisciplinary projects – cooperation of students of different study programs", IEEE EAEEIE 2013 (Chania, Greece), DOI 10.1109/EAEEIE.2013.6576532.

[19] S.C. dos Santos,et.al., " Applying PBL in software engineering education", CSEET 2009, 22nd Conference on Software Engineering Education and Training (Hyderbad, India), DOI 10.1109/CSEET.2009.39.

[20] Ericsson Mobility Report February 2016, with Q4 2016 update. Find latest pdf (free download) at https://www.ericsson.com/mobility-report.