# Multidisciplinary Skill Assessment for Embedded Software Development Education via a Robot Contest

Harumi Watanabe
Tokai University
Tokyo, Japan
harumi@wing.ncc.u-tokai.ac.jp

Mikiko Sato
Tokai University
Tokyo, Japan
mikiko.sato@tokai.ac.jp

Masafumi Miwa
Tokushima University
Tokushima, Japan
miw@tokushima-u.ac.jp

Makoto Imamura
Tokai University
Tokyo, Japan
imamura@tsc.u-tokai.ac.jp

Shintaro Hosoai
Change Vision, Inc.
Fukui, Japan
shintaro.hosoai@change-vision.com

Nobuhiko Ogura
Tokyo City University
Tokyo, Japan
ogura@tcu.ac.jp

Hiroyuki Nakamura
Kyushu University
Fukuoka, Japan
nakamura@qito.kyushu-u.ac.jp

Kenji Hisazumi
Kyushu University
Fukuoka, Japan
nel@slrc.kyushu-u.ac.jp

## ABSTRACT

This article introduces a multidisciplinary skill assessment for learning embedded software development. In the industry, software engineers and mechanical engineers have to communicate with each other, and hence, these engineers need to have multidisciplinary skills. For learning such skills, we need to give an occasion to work with different field students. To this end, we have been organizing a robot contest with embedded software development education. One of the goals of the contest is to inculcate multidisciplinary skills. However, we have not clarified the contribution. Thus, we construct a multidisciplinary skill assessment map based on the experiences gained through these contests. The map consists of (1) integration skill, (2) performing skill, and (3) cross-understanding skill.

## CCS Concepts

• **Computer systems organization → Embedded systems;** Robotics; • **Social and professional topics methodologies →** Computing education programs;

**Keywords** embedded system development; education; software engineering; robotics; project-based learning

## 1. INTRODUCTION

[1]The difficulty of training for embedded software development is providing support to learn multidisciplinary skills. In the industry, software engineers and mechanical engineers have to communicate with each other to build a new product together. To support the education for achieving multidisciplinary skills, we should make an opportunity to communicate with different field students. In a single university curriculum, we cannot provide such an opportunity. Additionally, students need to be motivated to learn broad skills. Thus, we should select educational material that will keep them motivated. To this end, we have been organizing a contest involving robots called Embedded System Symposium Robot Challenge (ESS-RC) for the past fourteen years. This contest has been providing embedded software development education for graduate and undergraduate students. The students participating in this contest belong to different fields from several universities, and they develop a robot system. Robot developments are well known effective educational materials [7, 11]. In particular, since robots are attractive for many people [4], we believe that they help sustain students' motivation.

However, even though we have applied these attractive materials, the issue of multidisciplinary skills is not easy to solve. Naturally, within a limited period of students' undergraduate course, we cannot educate them sufficiently every field. Thus, we need to provide them with multidisciplinary skills in stages. In our education, we divide multidisciplinary skills into three individual skills: integration, performing, and cross-understanding. To evaluate these skills, we consider an assessment map for ESS-RC. This article introduces an introductory seminar in the contest and evaluates the seminar on the map. The style of this seminar is project-based learning (PBL).

With regard to PBL, many works deal with robots [1,2,3,5,6,8,11]. In several works, the target students are undergraduate and graduate students learning embedded systems [3,5,9,11]. Several works address multidisciplinary skills [5, 10, 11]. In these works, students were part of robotics and mechatronics courses. On the other hand, our students were part of software science, robotics, and mechatronics courses.

## 2. MULTIDISCIPLINARY SKILLS

The section illustrates a multidisciplinary skill assessment map. We divide multidisciplinary skills into three individual skills: integration, performing, and cross-understanding. Each skill has a goal, which includes sub goals and activities. The goal is what students should learn. The activity is how to learn the goal. Table 1 shows the goals and activities in Spring School.

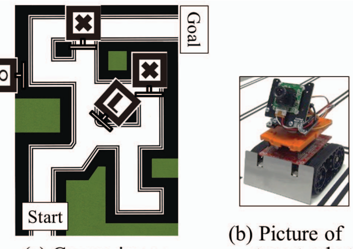
## 3. EVALUATION

As mentioned previously, we provide an introductory seminar called Spring School, which is the first learning course towards ESS-RC. The main game of ESS-RC is an auto-cruising game, in which the teams compete to see whose robot is faster and whose

*ICSE '18 Companion, May 27-June 3, 2018, Gothenburg, Sweden*

ACM ISBN 978-1-4503-5663-3/18/05.
https://doi.org/10.1145/3183440.3195059

robot correctly follows the given issues of the AR (Augmented Reality)- markers. Figure 1 shows the course image and the robot in the auto-cruising game. In the following, we evaluate Spring School. Seventy-eight students participated in the Spring School program, and their fields were computer science and control mechanics. The students were divided into 19 groups, with each group having 3 or 5 students. To evaluate the multidisciplinary skill assessment map, we checked the final presentations and the comments of five teachers. The following discusses each skill.



(a) Course image. (b) Picture of target robot.

**Figure 1. Materials used in the auto-cruising game.**

(1) Integration skill: Most students seemed to achieve step 2 of the goal. They measured the physical quantities of robot behavior and identified quantities for robot behavior. However, nearly half of the groups did not achieve the result or the model of control theory.

(2) Performing skill: All groups achieved step 2 of the goal. They addressed the KPT by using appropriate keywords. However, eight groups received negative comments from teachers. The comments were related to role sharing, process scheduling, and goal setting.

(3) Cross-Understanding skill: All students easily cleared step 2 of the goal, because the achievement evaluates if students performed the exercise.

# 4. CONCLUSION

This article introduced multidisciplinary skill assessment for learning embedded software development. The students who took participated in this assessment were part of an undergraduate course, and they aim of becoming embedded system engineers. The education helps to solve the problem of learning multidisciplinary skills. In the assessment, we regard multidisciplinary skills as (1) integration skill, (2) performing skill, and (3) cross-understanding skill. Each skill of the map contains a goal with three stages and the activity also has three steps. We discussed the activity based on the final presentations of the introductory seminar. Since we can distinguish the learning levels on the map, we believe that the map is appropriate for evaluating the skills. In future work, we will update the skill assessment map and education contents based on the results attained this year. Moreover, we are planning to extend the map to other skills.

# REFERENCES


[1] S. Akayama, S. Kuboaki, K. Hisazumi, T. Futagami, and T. Kitasuka. 2013. Development of a Modeling Education Program for Novices Using Model-driven Development. In *Proceedings of the Workshop on Embedded and Cyber-Physical Systems Education (WESE '12).* ACM, Article 4, 8 pages.

[2] D. J. Cappelleri and N. Vitoroulis. 2013. The Robotic Decathlon: Project-Based Learning Labs and Curriculum Design for an Introductory Robotics Course. *IEEE Transactions on Education* 56, 1 (Feb 2013), 73–81.

[3] N. Correll, R. Wing, and D. Coleman. 2013. A One-Year Introductory Robotics Curriculum for Computer Science Upperclassmen. *IEEE Transactions on Education* 56, 1 (Feb 2013), 54–60.

[4] G. Dudek and M. Jenkin. 2010. *Computational Principles of Mobile Robotics.* Cambridge University.

[5] H. Hassan, C. Dominguez, J. M. Martinez, A. Perles, J. V. Capella, and J. Albaladejo. 2015. A Multidisciplinary PBL Robot Control Project in Automation and Electronic Engineering. *IEEE Transactions on Education* 58, 3 (Aug 2015), 167–172.

[6] H. H. Huang, J. H. Su, and C. S. Lee. 2013. A Contest-Oriented Project for Learning Intelligent Mobile Robots. *IEEE Transactions on Education* 56, 1 (Feb 2013), 88–97.

[7] O. O. Ortiz, J. A. Pastor Franco, P. M. Alcover Garau, and R. Herrero Martin. 2017. Innovative Mobile Robot Method: Improving the Learning of Programming Languages in Engineering Degrees. *IEEE Transactions on Education* 60, 2 (May 2017), 143–148.

[8] L. Pinto, R. Rosa, C. Pacheco, C. Xavier, R. Barreto, V. Lucena, M. Caxias, and C. Maurccio. 2009. On the Use of SCRUM for the Management of Practical Projects in Graduate Courses. In 2009 39th *IEEE Frontiers in Education Conference.* 1–6.

[9] D. Rus. 2006. Teaching robotics everywhere. *IEEE Robotics Automation Magazine* 13, 1 (March 2006), 15–94.

[10] Z. Shiller. 2013. A Boom-Up Approach to Teaching Robotics and Mechatronics to Mechanical Engineers. *IEEE Transactions on Education* 56, 1 (Feb 2013), 103–109.

[11] J. Shin, A. Rusakov, and B. Meyer. 2015. Concurrent Software Engineering and Robotics Education. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, Vol. 2. 370–379.


**Table 1. Multidisciplinary skill assessment map**

| | | Integration skill | Performing skill | Cross-understanding skill |
|---|---|---|---|---|
| Definition | | Integrate different types of knowledge acquired on individual lectures into a practical system development. | Develop practically a system based on the knowledge acquired. | Communicate with students from different backgrounds. |
| Goal | Step 1 | **Awareness:** Understand where we need control theory in embedded software. | **Knowledge:** Learn basic schedule management. | **Interest:** Take an interest in acquiring knowledge of a different field. |
| | Step 2 | **Quantity:** Identify quantities for robot behavior. | **Experience:** Perform a small project. | **Try:** Solve issues of a different field. |
| | Step 3 | **Theory:** Understand some fundamental theories for moving robots. | **Postmortem:** Reflect on the project and make a plan for the next project. | **Question:** Ask a question about a different field. |
| Activity | Step 1 | **Trial and error:** Build a program without using theories. | **Knowledge:** Conduct a role-playing lesson by using a simple project management framework: Scrum. | **Interest:** Receive lectures on both individual fields and a general field for bridging them. |
| | Step 2 | **Observation:** Measure physical quantities of robot behavior. | **Experience:** Do basic schedule management on a small PBL. | **Try:** Try a small development that includes the essence of both fields. |
| | Step 3 | **Adaptation:** Build a program using a theory. | **Postmortem:** In the presentation, talk about "Keep, Problem, Try" in the five days' development. | **Question:** In the presentation, ask a question about a different field. |