

An Efficient Repository Model for Online Software Education

Won Joo Lee*, Yuncheol Baek**, Byung Seok Yang***

Abstract

In this paper, we propose an efficient repository model for online software education. The software education of app development consists of 7 stages: coding & debugging, submit, collaboration, review, validate, deployment, certification. Proposed repository model supports all 7 stages efficiently. In the coding & debugging stage, the students repeat coding and debugging of source. In the submit stage, the output of previous process such as source codes, project, and videos, are uploaded to repository server. In the collaboration stage, other students or experts can optimize or upgrade version of source code, project, and videos stored in the repository. In the review stage, mentors can review and send feedbacks to students. In the validate stage, the specialists validate the source code, project, and the videos. In the deployment stage, the verified source code, project, and videos are deployed. In the certification stage, the source code, project, and the videos are evaluated to issue the certificate.

▶ Keywords : Repository Model, Online Software Education, Coding & Debugging, Deployment,

1. Introduction

최근 4차 산업혁명으로 각 산업분야에서 소프트웨어(SW)의 중요성이 증가함에 따라 국내외적으로 소프트웨어 교육에 대한 관심이 높아지고 있다. 특히 국내에서는 2016년 교육부와 미래창조과학부에서 소프트웨어(SW) 연구·선도학교로 900개 학교를 선정하여 SW 인력 양성 사업을 추진 중이다. 국내 초·중·고교의 수는 초등학교 5,798개교, 중학교 3,204개교, 고등학교 2,344개교를 포함하여 총 11,526개교이다[1]. 소프트웨어(SW) 교육 선도학교는 초등학교 382개교, 중학교 196개교, 고등학교 107개교로 총 685개교이며, 2015년에 이어 2차년도

계속 지정 연구학교 및 선도학교 215개교를 포함한 900개교이다. 이는 국내 초·중·고교의 약 7.8% 수준으로 매우 부족한 상황이다.

초·중·고교의 소프트웨어(SW) 교육은 교과 시간 외에 창의적 체험 활동(자율, 동아리, 봉사, 진로활동), 자유학기 등을 활용하여 다양한 교육 프로그램을 운영하고 있다. 이러한 다양한 소프트웨어 교육 프로그램을 효율적으로 운영하기 위해서는 현재의 정보교사의 수로는 부족하다. 각 학교에서는 산업체 인사 및 대학 교수의 재능 기부를 통하여 정보교사의 부족 문제를 해결

• First Author: Won Joo Lee, Corresponding Author: Yuncheol Baek

*Won-Joo Lee (wonjoo2@inhac.ac.kr), Dept. of Computer Science, Inha Technical College.

**Yuncheol Baek (ybaek@smu.ac.kr), Dept. of Computer Science, SangMyung University.

***Byung Seok Yang (fstory97@spri.kr), Software Convergence Policy Lab., Software Policy & Research Institute.

• Received: 2016. 11. 11, Revised: 2016. 11. 25, Accepted: 2016. 12. 05.

• This work was supported by the 2015 Broadcasting Communication Policy Research Program funded by the Broadcasting Communication Development Fund of the Ministry of Science, ICT and Future Planning.

하고 있다. 또한, 농어촌 및 산간지역, 섬 지역 등의 지역적 문제와 IT 환경 부족 및 인적 문제로 인해 소프트웨어(SW) 교육을 받지 못함에 따라 소프트웨어(SW) 교육의 기회 불균등 문제점이 발생할 것이다. 소프트웨어 선도학교에 선정되지 못한 초·중·고교의 학생들은 소프트웨어(SW) 교육에 소외되는 두려움으로 인해 사교육에 의존할 가능성이 증가하고 있다.

향후 소프트웨어 교육에 대한 요구가 지속적으로 증가함에 따라 정보교사의 부족, 기회 불균등, 다양한 교육 콘텐츠 부족 등으로 인한 문제점도 증가할 것이다.

이러한 문제점을 해결하기 위한 하나의 방법으로 본 논문에서는 온라인 소프트웨어 교육을 위한 레파지토리(Repository) 모델을 제안한다. 이를 위해 2장에서 온라인 소프트웨어 교육의 사례를 알아본다. 3장에서는 소프트웨어의 교육 유형에 따른 레파지토리 모델을 분석하여 장단점을 파악한다. 4장에서 효율적인 온라인 소프트웨어 교육을 위한 레파지토리 모델을 제안한다. 그리고 마지막 5장에서 결론을 맺는다.

II. The Case of Online Software Education

1. 국내 온라인 SW 교육 사례

국내 온라인 SW 교육은 주요 이러닝 서비스 업체(메가스터디, 웅진씽크빅, 크레듀, 아이넷스쿨, 비상교육, 정상제이엘에스, 디지털대성, 청담러닝, 에듀박스, 유비온 등)들을 중심으로 주로 OA나 자격증 위주의 컴퓨터 활용 교육 중심으로 이루어져 왔다. 하지만 최근에는 비주얼 프로그래밍(Scratch, Entry, Kodu 등)과 텍스트 프로그래밍(C, C#, Java, Python 등)과 같이 SW 개발 및 구현을 위한 프로그래밍 언어 중심의 교육이 진행되고 있다. 또한, 아두이노(Arduino), 로봇 등과 같이 직접 체험해 보는 피지컬컴퓨팅이 활발하게 진행되고 있다. 최근 국내에서 진행되는 프로그래밍 중심의 주요 온라인 SW 교육에 살펴본다.

1.1 Korea SW(<http://koreasw.org/>)

Korea SW(<https://koreasw.org/>)는 정보통신산업진흥원과 미래인재연구소가 공동으로 운영하는 온라인 교육 서비스로 Scratch, Entry, Python 등의 프로그래밍과 피지컬컴퓨팅 관련 짧은 동영상들을 제공하는 레파지토리를 운영한다[2].

1.2 소프트웨어야 놀자(<http://www.playsw.or.kr/>)

초·중등 학생을 위한 대표적인 온라인 SW 교육 사이트인 소프트웨어야 놀자(<http://www.playsw.or.kr/>)는 아이들이 쉽게 이해하고 재미있는 창의적 도구로 소프트웨어를 경험하고, 학부모와 선생님들에게는 소프트웨어 교육의 장기적인 가이드

라인을 제시하기 위해 시작한 네이버 주관 온라인 사이트이다[3]. 이 사이트에서는 컴퓨팅사교육, 스크래치수업, 아두이노 수업 등의 소프트웨어 관련 교재와 엔트리를 활용한 실습 가능 동영상 강의와 SW 관련 된 ebs-BBC동영상들을 제공하고 있으며, 선생님들을 위한 교육지도서가 있어 수업시간에 활용이 가능하다.

1.3 OLC(<https://www.oss.kr/>)

대학생 또는 제작자 대상의 온라인 SW 교육 사이트중의 하나인 OLC(Open Source Software Learning Community Center)센터는 공개SW 전문인력 양성을 목적으로 구 지식경제부가 주최하고, 정보통신산업진흥원(NIPA)과 공개소프트웨어협회(KOSSA)가 공동주관하는 개방형 소프트웨어 교육센터이다[4]. OLC센터는 체계적인 양질의 커리큘럼으로 공개 SW 개발인력 부족 문제를 해소하고 계층별, 기능별 전문 인력 양성 기반을 구축하고 있다. 학생, 현직 개발자, IT관련 종사자 등 공개 SW에 관심 있는 누구나 참여가 가능하며, 특성화된 우수 콘텐츠의 수강에서부터, 사용자가 직접 제작한 교육 콘텐츠로 유·무료의 강의실을 개설할 수 있는 온라인 오픈마켓 서비스를 제공하는 레파지토리를 운영한다.

1.4 코다임(codigm, <http://codigm.com/>)

코다임(codigm, <http://codigm.com/>)은 클라우드 인프라에 대한 깊은 이해와, 웹 애플리케이션 개발에 대한 적지 않은 노하우로 국내를 넘어서 글로벌 서비스를 표방하는 기술 기반 스타트업이다. 코다임은 언제 어디서나 SW를 배우고, 개발할 수 있는 클라우드 코딩 서비스인 구름(goorm)을 개발하고 서비스하고 있다[5].

클라우드 코딩 서비스 구름은 구름 EDU를 이용하여 초등학교, 중학교, 고등학교, 대학교를 비롯한 다양한 교육 기관에서 프로그래밍 언어를 효율적으로 교육할 수 있다. 특히 구름은 실습용 컴퓨터의 성능 노후화 및 한 대의 컴퓨터를 여러 명이 사용하여 유지 관리가 어려운 환경, 그리고 컴퓨터마다 파편화된 실습용 개발 도구의 버전 등은 일관된 프로그래밍 교육이 어려운 환경에서 SW 교육에 유용하다[6]. 구름 EDU를 활용하면, 온라인에서 자동으로 코딩 문제를 출제하고 채점을 해줄 뿐만 아니라, 수업 시간 이외에도 학교·학원·집 구분 없이 언제 어디서나 항상 일관된 교육 환경을 제공한다. 강의자 도구를 지원하여 오프라인 수업에서도 활용 가능하며, 실제 교과서와 연동되어 온·오프라인에서 연속성 있는 코딩 교육이 가능하다. 구름의 특징은 다음과 같다.

- 손쉬운 강의 교재 제작 가능 (기존 강의안 활용, PDF, 슬라이드 공유 연동, 신규 제작을 위한 기능 제공)
- 다양한 SW 교육 환경 제공(C/C++, Java, Python 등 총 9종 언어)
- 손쉬운 학생 관리를 위한 교수자 모드 제공 (학생 등록, 성적

관리, ..)

- 프로그래밍 시험 서비스 및 자동채점 기능 제공
- 소스코드 유사도 검사 기능 제공

2. 국외 온라인 SW 교육 사례

해외 온라인 SW 교육은 교육대상과 수준에 따라 다양하게 진행하고 있다. 해외 온라인 SW 교육 현황에 대한 분석 결과는 그림 1과 같다[7].

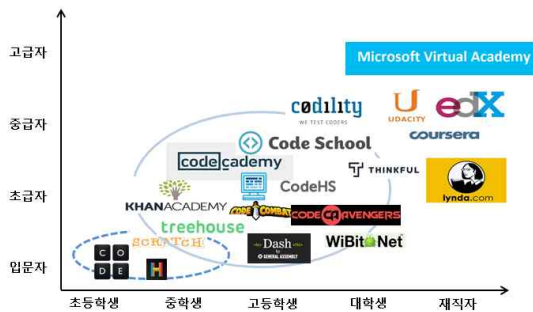


Fig. 1. Online software education status

2.1 Code.org

Code.org는 영유아와 초등학생들을 대상으로 입문 수준의 비주얼 프로그래밍 교육을 제공한다[8]. Code.org는 유명인사가 직접 나와 소프트웨어에 대한 중요성과 SW교육을 장려하는 캠페인형 동영상상을 제공한다. 또한 입문을 위한 튜토리얼 교육과 교사들에게 교육 자료를 제공하는 온라인 교육 서비스이다. code studio는 학생들이 구현한 앱과 그림을 저장하고 있으며, 해당 그림이나 스크린 샷을 클릭하면 실행 결과를 볼 수 있다.

2.2 Codility.com

Codility.com은 초중급자를 대상으로 다양한 프로그래밍 언어(C, C++, C#, Go, Java, JavaScript, Lua, Objective-C, PHP, Pascal, Perl, Python, Ruby, Scala, VB.NET)에 대한 코딩 테스트와 알고리즘을 공부하고 문제풀이를 할 수 있는 프로그래밍 교육서비스를 제공한다[9].

2.3 Udacity.com

2011년 스탠포드대학에서 시작한 Udacity.com은 Coursera, 에드엑스와 함께 세계 3대 MOOC 사이트로 알려져 있다[10]. Udacity 설립자 세바스찬 스룬(Sebastian Thrun) 교수가 처음 개설한 컴퓨터공학 강의에 160,000 여명의 학생이 등록한바 있다. Udacity에서는 초급에서 고급까지 다양한 강의를 제공하고 있다. 특히 특정 과목과 프로젝트를 이수하면 나노 학위(Nano degree)를 수여하는 서비스를 제공하고 있다. Udacity.com에서 제공하는 12개의 Nanodegree 프로그램은 2015년 12월 현재 Android Developer, Full Stack Web Developer, Front-End Web Developer, iOS Developer 등 12개의 Nanodegree 프로그램을 제공하고 있다.

III. Repository Model Analysis

소프트웨어 교육은 교재, 도구, 결과물의 형태에 따라 표 1과 같이 분류할 수 있다.

Table 1 Programming Type for SW education

Type	Characters	Case
Visual Programming	programming combine block which have code execute command function easy and simple using GUI graphic	Scratch, Entry, Kodu
Text Programming	Code based programming know computer architecture and library	Python, C, C#, Java
Physical Computing	HW and SW convergence Hardware control software using	Robot programming, Arduino

1. Visual Programming

비주얼 프로그래밍 유형으로 대표적인 것은 스크래치(Scratch) 프로그래밍이다. 스크래치는 MIT 미디어랩의 Lifelong Kindergarten Group 에서 운영하는 프로젝트이며 무료로 제공된다. <https://scratch.mit.edu/> 사이트에 접속하면 스크래치 초기 화면을 볼 수 있다. 스크래치는 학습자가 창의적으로 생각하기, 논리적으로 추론하기, 다른 사람들과 협동하는 능력을 배우는데 도움을 준다[11]. 또한 다양한 인터랙티브 스토리, 게임, 애니메이션 등을 만들 수 있고, 만든 작품을 스크래치 온라인 커뮤니티를 통하여 다른 학습자들과 공유할 수 있다.

1.1 스크래치(Scratch) 레파지토리 분석

스크래치(<https://scratch.mit.edu/>) 사이트에서는 아래와 같은 항목들을 볼 수 있다. 그림 2의 최근 프로젝트는 학생들이 최근에 제작한 스크래치 프로젝트를 등록하는 기본 공간이다.

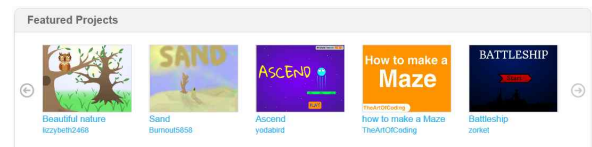


Fig. 2. Featured Projects

그림 3의 최근 스튜디오는 학생들이 제작한 스크래치 프로젝트를 등록하고, 다른 학생들과 의견 교환을 통하여 더욱 발전 방향으로 진행할 수 있는 공간이다.



Fig. 3. Featured Studios

그림 4의 큐레이트 된 프로젝트는 Manager 및 Curator 들이

등록 스크래치 프로젝트에 대하여 의견을 등록하는 공간이다.

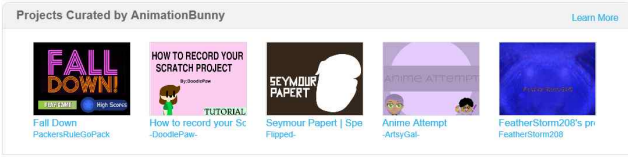


Fig. 4. Projects Curated by AnimationBunny

그림 5의 리믹스중인 프로젝트는 등록된 스크래치 프로젝트를 공유하여 새로운 아이디어를 접목하고, 수정하여 더욱 발전 방향으로 진행할 수 있는 공간이다.

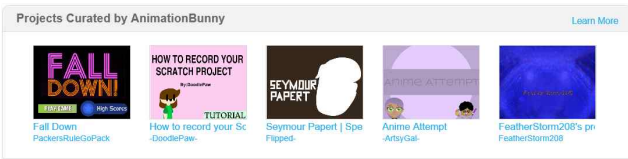


Fig. 5. What the Community is Remixing

그림 6의 사람들이 좋아하는 프로젝트는 학습자들로부터 많은 추천을 받은 스크래치 프로젝트들을 모아둔 공간이다.

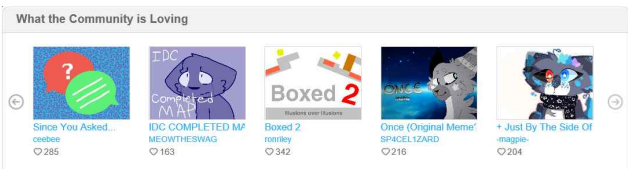


Fig. 6. What the Community is Loving

그림 6에서 ♥(이 프로젝트가 좋아요) 버튼을 클릭하면 추천 수가 증가한다.

2. Text Programming

텍스트 프로그래밍 유형으로 대표적인 것은 Java, C, C#, 파이썬(Python) 프로그래밍이다. 텍스트 프로그래밍은 아래 그림 7과 같이 C#언어의 IDE 초기 화면을 볼 수 있다.

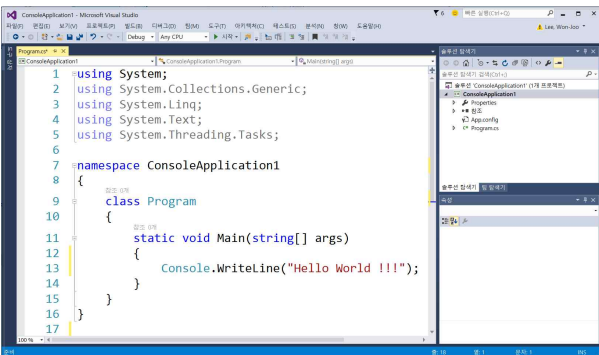


Fig. 7. IDE Start Page

그림 7에서 “Console.WriteLine(“Hello World !!!”);” 코드를 입력하고, 구문(Syntax) 에러를 찾기 위해 아래 그림 8과 같이 빌드(B) 메뉴에서 [솔루션 빌드] 메뉴를 클릭하여 빌드(Compile) 과정을 수행한다.



Fig. 8. Build menu

빌드(Compile) 과정 수행한 후에 실행하기 위해 아래 그림 9과 같이 디버그(D) 메뉴에서 [디버그하지 않고 시작] 메뉴를 클릭하여 실행한다.



Fig. 9. Debug menu

2.1 텍스트 프로그래밍 레포지토리 분석

텍스트 프로그래밍 레포지토리로 대표적인 예는 GitHub (<https://github.com/>) 이다[12]. GitHub는 git을 사용하는 프로젝트를 지원하는 웹 기반의 호스팅 서비스이다. GitHub에서는 다음과 같은 중요한 요소를 가진다.

- 버전 관리
- 소스 코드 저장
- 협업(Collaboration)
- GitHub에서 개발자는 그림 10과 같이 프로젝트를 만들고, 웹 페이지를 통해 열람, 수정 등이 가능하다.

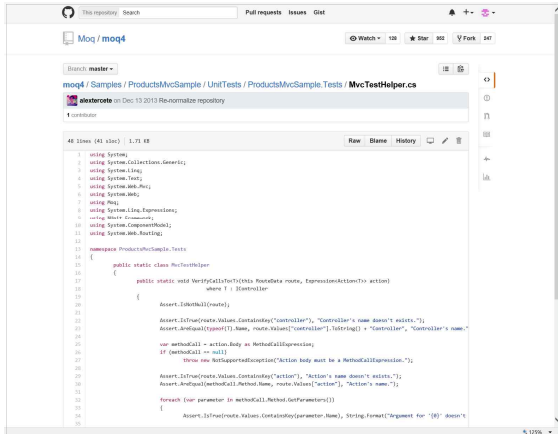


Fig. 10. GitHub source code

그림 10에서 fork 부분의 숫자 247은 fork 횟수이다. fork는 프로젝트 복사 횟수를 의미한다. 개발자가 fork 하면 개발자의 계정에 현재 작성된 버전이 그대로 복사가 되고, 수정한 버전을 생성하여 계속 사용할 수도 있다. 또한, [Pull request] 메뉴를 클릭하여 rails 관리자에게 수정한 부분을 반영해 달라고 요청할 수 있다. [Watch] 메뉴는 프로젝트의 수정을 지속적으로 관찰하고 수정이 발생하면 notification을 받을 수 있다.

GitHub를 SW 교육에 활용한 예를 들어 보자. 그림 11과 같이 1~10사이의 2의 배수 합을 구하는 프로그램을 구현하는 예를 들어보자.

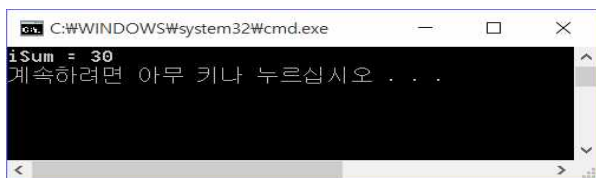


Fig. 11. Execution result

위의 그림 11의 결과를 얻기 위해 한 학생이 그림 12와 같이 소스코드를 작성했다면 다른 학생이 그림 12의 소스코드 14~17 라인을 그림 13과 같이 최적화 할 수 있다.

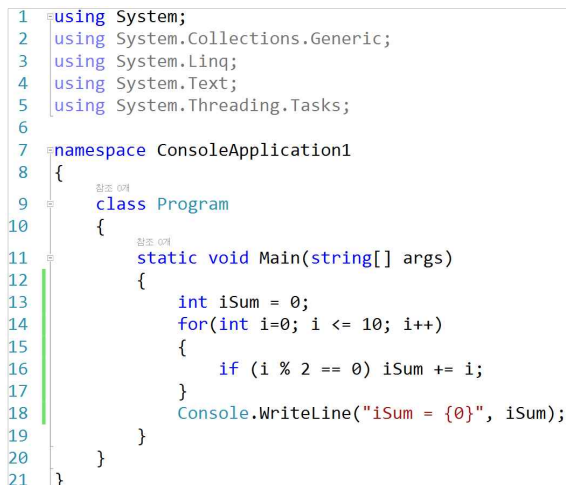


Fig. 12. C# source code

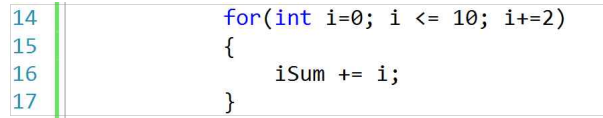


Fig. 13. Optimized C# source code

그림 13과 같이 최적화 한다면 for문의 실행 횟수를 1/2로 줄일 수 있고, if (i % 2 == 0)문을 제거하여 프로그램의 실행 속도를 높일 수 있을 것으로 예상된다.

3. Physical computing

피지컬 컴퓨팅은 HW와 SW가 융합된 형태로 HW를 제어하는 SW 프로그래밍을 구현하는 로봇, 아두이노 프로그래밍 등을 예로 들 수 있다. 그림 14의 Hamster의 사양은 표 2와 같다 [13].



Fig. 14. Hamster and USB to BLE Bridge dongle

그림 14의 USB to BLE Bridge dongle은 블루투스 4.0 BLE 패킷을 시리얼 통신 패킷으로 변환해 주는 장치이다. PC와 Hamster를 블루투스로 연결할 때는 페어링 과정을 통한다. 페어링 과정은 쉽고 빠르며 여러 대의 Hamster전원이 켜져 있어도 자신의 Hamster 만을 찾아 연결한다.

Table 2. Hamster specification

Item	Characters
Communication Type	• Bluetooth 4.0 BLE (connection range < 15m)
Battery	• Lithium-Ion • Charge 30 minute • Operation: the average 1 hour • Stand-by: maximum 12 hour
Charging terminal	• micro USB
PC connection	Serial communication with USB dongle
Smartphone connection	Bluetooth 4.0 BLE
Sensor	• proximity sensor x 2, • Line sensor x 2, • LED sensor x 2, • Tri-Axial Accelerometer, • Photo Resistor, • Inner Temperature sensor

그림 15의 Hamster 주행을 제어하는 SW 프로그래밍은 파이썬, 스크래치, 엔트리, 자바스크립트, 자바, C/C++, 등의 다양한 프로그래밍 언어로 구현할 수 있다.

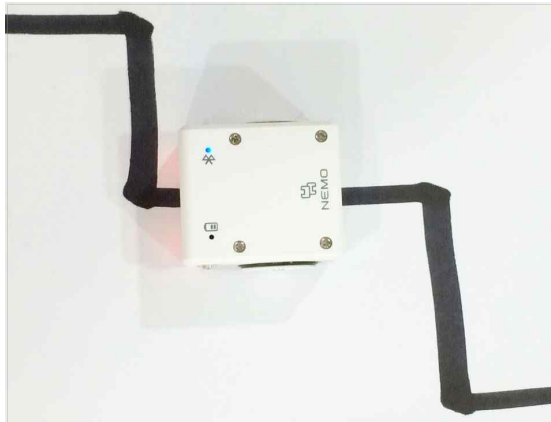


Fig. 15. Hamster line tracer

그림 16은 파이썬 프로그래밍 언어를 사용하여 Hamster의 주행을 제어하는 소스 코드이다[14].

```

from roboid.hamster import *

hamster = Hamster()

while True:
    if hamster.read(Hamster.LEFT_FLOOR) > 50:
        hamster.write(Hamster.LEFT_WHEEL, 0)
        hamster.write(Hamster.RIGHT_WHEEL, 30)
    else:
        hamster.write(Hamster.LEFT_WHEEL, 30)
        hamster.write(Hamster.RIGHT_WHEEL, 0)
    
```

Fig. 16. Python source code for Hamster control

3.1 피지컬 컴퓨팅 레파지토리 분석

융합 기반의 프로그래밍은 하드웨어와 소프트웨어가 융합된 형태로 하드웨어를 제어하는 소스 코드가 동반되어야 한다. 기본적으로 텍스트 프로그래밍의 레파지토리에 하드웨어의 동작을 확인할 수 있는 동영상 파일이 필요하다.

IV. The Proposed Repository Model

소프트웨어 교육의 유형에 따라 그 결과물의 형태는 표 3과 같이 분류할 수 있다.

Table 3. Result type according to software education

SW education	Result type
Visual Programming	<ul style="list-style-type: none"> Project
Text Programming	<ul style="list-style-type: none"> Project Source code
Physical Computing	<ul style="list-style-type: none"> Project Source code Video file

표 3에서와 같이 소프트웨어 교육의 결과물인 프로젝트, 소스코드, 동영상 등을 레파지토리에 저장할 수 있다. 따라서 스크래치 레파지토리와 GitHub 레파지토리를 융합한 모델이 바람직할 것으로 판단한다.

1. 온라인 SW 교육 레파지토리 제안

본 논문에서 제안하는 소프트웨어 교육 레파지토리 모델은 그림 17과 같다.

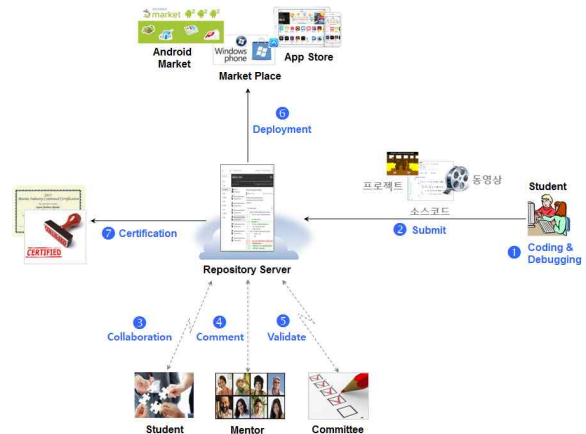


Fig. 17. The proposed online software repository

소프트웨어 교육에서 App 개발 및 배포 과정은 그림 18과 같다.

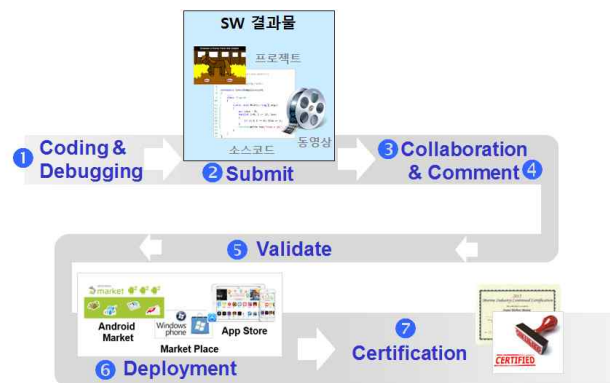


Fig. 18. App development and deployment process

① Coding & Debugging

학습자들이 소스를 코딩하고, 디버깅을 반복하는 과정으로 앞에서 설명한 클라우드 코딩 서비스 구름(goorm)과 같은 환경 제공이 필요하다.

② Submit

Coding & Debugging 과정의 결과물인 소스코드, 프로젝트, 동영상 등을 레파지토리 서버에 업로드하는 과정이다. 이 과정에서는 스크래치, 엔트리 등과 같이비주얼 프로그래밍 결과물은 스크래치, code.org의 레파지토리와 같은 기능 제공이 필요하다. 하지만 Java, Python 등과 같은 텍스트 프로그래밍 결과

물은 클라우드 코딩 서비스 그룹(goorm)과 codility.com의 소스코드 테스트와 같은 환경 제공이 필요하다. 또한, 로봇, 아두이노 등과 같은 피지컬 컴퓨팅 결과물은 스크래치, code.org의 레파지토리와 같은 기능과 클라우드 코딩 서비스 그룹(goorm)과 codility.com의 소스코드 테스트와 같은 환경이 융합된 기능을 제공하는 환경이 필요하다.

③ Collaboration

Collaboration 과정은 레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상에 대하여 다른 학습자 또는 전문가들이 참여하여 최적화 하거나 버전 업그레이드 하는 과정이다. 이 과정은 다른 개발자들과 팀을 조직하고 슬라이드/PDF 공유, 채팅 기능 뿐만 아니라 소스 코드의 실시간 동시 편집도 가능한 클라우드 코딩 서비스 그룹(goorm)이 제공하는 협업 기능이 필요한 과정이다.

④ Review

레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상에 대하여 전문가 또는 멘토들이 Review하는 과정이다. 이 과정에서는 Apple의 AppStore, Microsoft의 Windows Store와 같이 등록된 앱에 대하여 다른 개발자 또는 이용자들이 review할 수 있는 기능이 필요하다.

⑤ Validate

레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상에 대하여 전문가 위원들이 검증하는 과정이다. 이 과정에서는 Apple의 AppStore, Microsoft의 Windows Store와 같이 등록된 앱에 대하여 전문가(개발자, 기획자, 등) 들이 검증할 수 있는 기능이 필요하다.

⑥ Deployment

검증된 소스코드, 프로젝트, 동영상을 배포하는 과정이다. 레파지토리에 등록된 많은 소스코드, 프로젝트, 동영상을 검증하여 우수한 품질의 소스코드, 프로젝트, 동영상을 다른 개발자들이 사용할 수 있도록 공개 배포하는 기능이 필요하다.

⑦ Certification

평가를 통하여 자격증을 수여하는 과정이다. 이 과정에서는 Codility.com에서 제공하는 코딩 테스트와 알고리즘 테스트와 같은 과정을 통과하고, Udacity.com에서 제공하는 Nanodegree 프로그램과 같은 교육 과정을 이수하면 자격증을 수여하는 기능이 필요하다.

V. Conclusions

소프트웨어 교육은 교재, 도구, 결과물의 형태에 따라 비주얼 프로그래밍, 텍스트 프로그래밍, 피지컬 컴퓨팅으로 분류할 수 있다. 비주얼 프로그래밍은 GUI 기반으로 단순하고 쉬운 프로그래밍으로 명령어를 수행하는 코드를 가진 블록을 조합하여 프로그래밍한다. 대표적인 비주얼프로그래밍은 스크래치, 엔트리, 코두 등이다. 텍스트 프로그래밍은 코드 기반의 프로그래밍으로 컴퓨터구조와 라이브러리를 알아야 한다. 대표적인 텍스트 프로그래밍은 Python, C, C#, Java 언어 등이다. 피지컬 컴퓨팅은 하드웨어와 소프트웨어가 융합된 형태로 하드웨어를 제어하는 SW를 구현한다. 대표적인 피지컬 컴퓨팅은 Arduino, 로봇 프로그래밍 등이다. 이러한 소프트웨어 교육에서 강의자료, 프로젝트 결과물, 소스코드, 동영상 등을 저장하여 다른 학습자들과 공유할 수 있는 기능을 제공하는 것이 레파지토리이다.

본 논문에서는 온라인 소프트웨어 교육을 위한 레파지토리 모델을 제안하였다. 소프트웨어 교육에서 App을 개발하고 배포하는 과정은 Coding & Debugging, Submit, Collaboration, Review, Validate, Deployment, Certification 등의 7개 과정으로 분류할 수 있다. 제안한 레파지토리 모델은 이러한 7개 과정을 효율적으로 지원한다. Coding & Debugging 과정에서는 학습자들이 소스를 코딩하고, 디버깅을 반복한다. Submit 과정에서는 Coding & Debugging 과정의 결과물인 소스코드, 프로젝트, 동영상 등을 레파지토리 서버에 업로드 한다. Collaboration 과정에서는 레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상에 대하여 다른 학습자 또는 전문가들이 참여하여 최적화 하거나 버전을 업그레이드 한다. Review 과정에서는 레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상에 대하여 멘토들이 멘토링 한다. Validate 과정에서는 전문가 위원들이 레파지토리 서버에 업로드 된 소스코드, 프로젝트, 동영상을 검증한다. Deployment에서는 검증된 소스코드, 프로젝트, 동영상을 배포한다. Certification 과정에서는 검증된 소스코드, 프로젝트, 동영상을 평가하여 자격증을 발급한다.

기존의 온라인 SW 교육 레파지토리들은 App을 개발하고 배포하는 과정에서 Validate, Certification 부분에 취약점이 있다. 특히 평가를 통하여 자격증을 수여하는 Certification 과정에서는 다양한 평가 요소와 평가 방법이 필요하다. 따라서 향후에는 소프트웨어 교육의 프로젝트 결과물을 평가하는데 필요한 평가 요소와 평가 방법 도출에 대한 연구가 필요하다고 판단된다.

REFERENCES

- [1] <http://cesi.kedi.re.kr/index>
- [2] <http://koreasw.org/>
- [3] <http://www.playsw.or.kr/>
- [4] <http://olc.oss.kr>
- [5] <http://codigm.com/>
- [6] <https://www.goorm.io/>
- [7] B. S. Yang, H. Y. Kil, and Y. M. Kim, "Effective Online Software education System for elementary and Secondary School," SPRi Research report, Broadcast communication policy research 2015-promotion-055, Jan. 2016
<https://spri.kr/post/13668>
- [8] <https://code.org/>
- [9] <https://codility.com/>
- [10] <https://www.udacity.com/>
- [11] <https://scratch.mit.edu>
- [12] <https://github.com/>
- [13] <http://www.hamster.school/>
- [14] http://hamster.school/ko/tutorial/python/line_tracer_one_sensor.jsp

Authors



Won Joo Lee received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Hanyang University, Korea, in 1989, 1991 and 2004, respectively.

Dr. Lee joined the faculty of the Department of Computer Science at Inha Technical College, Incheon, Korea, in 2008, where he has served as the Director of the Department of Computer Science. He is currently a Professor in the Department of Computer Science, Inha Technical College. He has also served as the Vice-president of The Korean Society of Computer Information and the Editor-in-Chief for the Journal of The Korean Society of Computer Information. He is interested in parallel computing, internet and mobile computing, and cloud computing.



Yuncheol Baek received the B.S., M.S. and Ph.D. degrees in Computer Science from Seoul National University, Korea, in 1988, 1990 and 2005, respectively.

He joined the faculty of the Department of Computer Science at Sangmyung University, Seoul, Korea, in 1996, where he has served as the dean of the Software College. He is currently a Professor in the Department of Computer Science, Sangmyung University. He had visited Princeton University as a research fellow from 2005 to 2007. He is a member of ACM, IEEE Computer Society, KIISE and KIPS. He is interested in performance and security of cloud computing system, digital forensics and computing for secure society.



Byoung Seok Yang received a bachelor's degree in 2004 from Soongsil University in Computer Science.

Since then, he has developed mobile phone software in Infracore and has developed OCR engine research and development and various web services until 2010 by Naver, a Korean portal site.

In 2011, he was responsible for Naver's corporate strategy and service planning and management. Since 2014 in SPRi, he has been conducting research on developers and software education policies and recently on virtual reality related policy work.

In addition, he is a famous blogger writing various IT subject.