

UML과 LabVIEW OOP를 기반으로 한 내장형 프로그래밍 교육의 적용 방안 및 분석

정덕길^{1*} · 정민포² · 조혁규² · 노영욱³

The Application and Analysis of the Embedded Programming Education Based on UML and LabVIEW OOP

Deok-Gil Jung^{1*} · Min-Po Jung² · Hyuk-Gyu Cho² · Young-Uhg Lho³

^{1*}Department of Computer Science, Dong-eui University, Pusan 614-714, Korea

²Department of Cyber Police & Science, Youngsan University, Gyeongnam 626-790, Korea

³Department of Computer Education, Silla University, Pusan 617-736, Korea

요 약

컴퓨터 언어를 배우는 학습자는 텍스트 기반으로 된 프로그래밍을 매우 어렵다고 생각하며, 객체 지향 프로그래밍 학습에서도 어려움을 호소한다. 이러한 문제를 해결하기 위한 방법이 객체 지향 시각 프로그래밍을 프로그래밍 교육에 도입하는 것이다. 이 논문에서는 프로그래밍 교육에서 이러한 문제를 해결하기 위해 객체지향 모델링을 지원하기 위해 UML을 도입하고 시각 프로그래밍 방법론으로서 객체지향 프로그래밍을 지원하는 LabVIEW OOP를 사용하여 학습자들에게 프로그래밍 교육을 수행하는 방법을 제시한다. 또한, 이 교육 방법론을 내장형 프로그래밍 교육에 적용하는 방안을 제안하며, 제시된 프로그래밍 교육 방법에 대해 설문조사를 실시하여 교육적인 효과를 분석한다.

ABSTRACT

The students who learn the computer languages think it very difficult to program using the text-based programming languages and also to learn the object-oriented programming. Therefore, we introduce the object-oriented visual programming into the programming education. And, to solve the this problem in the programming education, in this paper, we introduce the UML to support the object-oriented modeling and to teach the students using the LabVIEW OOP that supports the object-oriented visual programming. And, we propose to apply it on the embedded programming education. Also, we analyze the educational effect of the proposed education methodology of programming based on the student survey.

키워드 : UML, LabVIEW OOP, 프로그래밍 교육, 내장형 프로그래밍, RFID

Key word : UML, LabVIEW OOP, Programming Education, Embedded Programming, RFID

접수일자 : 2015. 02. 02 심사완료일자 : 2015. 02. 16 게재확정일자 : 2015. 03. 02

* **Corresponding Author** Deok-Gil Jung(E-mail:dgiung@deu.ac.kr, Tel:+82-51-890-2351)

Department of Computer Science, Dong-eui University, Pusan 614-714, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2015.19.3.708>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

컴퓨터 언어를 배우는 학습자는 다양한 분야에 적합한 프로그래밍 언어를 배우고 텍스트 기반으로 된 프로그래밍이 매우 어렵다고 생각한다. 이러한 문제를 쉽게 풀기 위한 방법이 프로그램을 시각적으로 표현하는 것이다. 기존의 시각 프로그래밍 방법으로 사용되는 Visual C++, Visual Basic, Delphi와 같은 프로그래밍 언어에서 외부 인터페이스는 시각적 컴포넌트로 표현되고 컴포넌트의 작동에 대해서는 텍스트 기반으로 표현한다. 이러한 프로그램을 배우는 학습자들은 컴포넌트 작동에 대한 텍스트 프로그래밍에 대해 어려워하고 있으며 프로그래밍을 싫어하는 한 요소가 되었다[1]. 대학의 프로그래밍 언어 교육 현장을 살펴보면, 텍스트 방식의 주입식 교육을 수행하는 것이 일반적이다. 이는 대부분의 학생들이 프로그래밍 문제를 해결하기 위한 본질을 생각하기 전에 프로그래밍 문법의 어려움으로 인해 중도 포기를 하는 계기가 되었다[2].

이 논문에서는 이러한 문제를 해결하기 위해 논리적 사고를 표현하면서 객체지향 모델링을 지원하기 위해 UML[3-6]을 도입하고, 텍스트 프로그래밍 요소를 시각 프로그래밍 요소로 대체하기 위해 객체지향 개념을 지원하는 LabVIEW OOP[7,8]를 사용하여 학습자들에게 프로그래밍 교육을 수행하는 방법을 제시한다. 또한, 제시된 프로그래밍 교육 방법론을 사용하여 내장형 프로그래밍 교육에 적용하는 방안을 제시하기 위하여, RFID[9] 태그 불량 검출 기능을 수행하는 불량검출 시스템을 UML을 이용하여 객체지향 기법으로 설계하고 UML로 설계된 모델링을 객체지향을 지원하는 시각 언어인 LabVIEW OOP로 구현하는 방안을 제시하며, 설문조사를 실시하여 교육적인 효과를 분석한다.

II. UML과 LabVIEW OOP를 활용한 프로그래밍 교육 방안

프로그래밍 교육 현장에서 시각 언어인 LabVIEW OOP와 텍스트 언어인 JAVA를 이용한 프로그래밍 언어 교육에 대한 적용 단계는 그림 1과 같다.

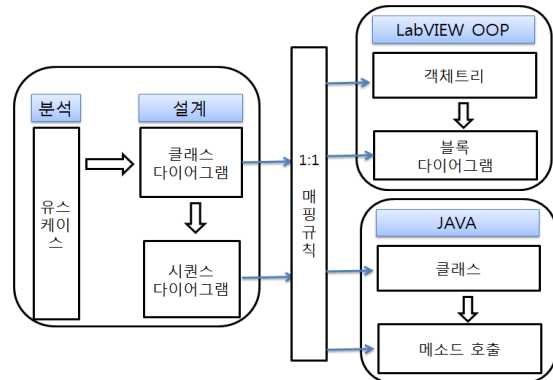


그림 1. UML 다이어그램과 LabVIEW OOP 클래스의 매핑 흐름도

Fig. 1 Mapping flow diagram between UML diagram and LabVIEW OOP class

- (1) UML로 설계된 유스케이스(Use Case)[10]를 제시한다.
- (2) 설계된 유스케이스를 이용하여 UML의 클래스 다이어그램을 설계한다.
- (3) 설계된 클래스 다이어그램을 이용하여 요구조건에 적합한 시퀀스 다이어그램(Sequence Diagram)[11]을 설계한다.
- (4) 설계된 UML의 클래스 다이어그램을 사용하여 LabVIEW OOP의 프로젝트 탐색기에 1:1 매핑 규칙으로 클래스를 추가하여 객체 트리를 구성한다. JAVA[12]에서도 1:1 매핑 규칙으로 클래스를 텍스트로 추가한다.
- (5) 설계된 UML의 클래스 다이어그램을 사용하여 LabVIEW OOP에서 생성된 클래스에 변수와 메소드를 추가한다. JAVA에서 생성된 클래스에 변수와 메소드를 추가한다.
- (6) 설계된 UML의 시퀀스 다이어그램을 사용하여 LabVIEW OOP에서 생성된 메소드 VI를 사용하여 블록 다이어그램에서 와이어링(wiring)을 진행한다. JAVA에서 생성된 클래스 내의 메소드를 사용하여 시퀀스 다이어그램에 맞게 객체간의 메소드를 호출한다.
- (7) 요구사항에 따라 화면을 설계하기 위해, LabVIEW OOP에서 프론트 패널을 설계한다. JAVA에서는 화면을 설계하기 위해서는, AWT와 같은 별도의 라이브러리를 사용해야 한다.

III. RFID 태그 불량 검출 시스템의 적용

실시간 태그 불량 판단 시스템을 설계하기 위해, RFID 태그 불량 검출 시스템의 필수 요구사항을 반영하여 UML을 사용한 최상위 유스케이스 다이어그램을 그림 2와 같이 제시한다.

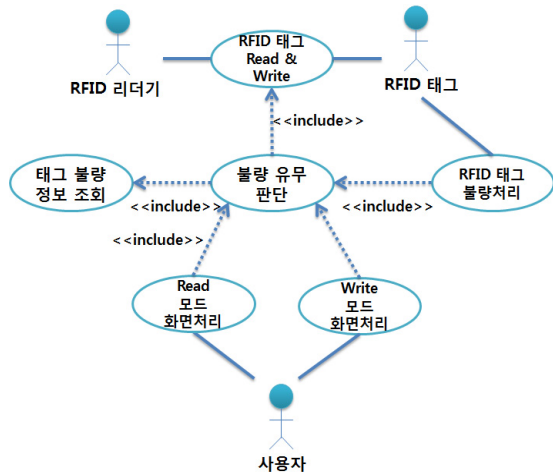


그림 2. 태그 불량 검출 유스케이스 다이어그램
Fig. 2 Usecase diagram of tag defect detection

3.1. UML의 클래스/시퀀스 다이어그램

그림 3에서는 불량 태그 검출 시스템에서 사용되는 클래스들을 정의하고 클래스들 간에 존재하는 정적인 관계를 표현할 수 있는 클래스 다이어그램을 보여주고 있다.

RFID 태그 불량 검출 시스템의 중요 클래스는 frmMain, Sensor, ImagePanel, DBWrapper, Options, Validation으로 분류된다. Sensor 클래스는 Machine_Controller와 RFID_Reader 클래스인 두 개의 서브클래스를 가지고 RFID Reader와 다른 디바이스들 사이의 통신 기능을 담당한다. ImagePanel 클래스는 태그, 센서, 오류 정보 정보를 보여주는 기능을 담당한다. DBWrapper 클래스는 태그 정보와 시스템의 옵션 정보를 저장하는 기능을 담당한다. Options 클래스는 GUI 세팅으로 태그 속성 정보와 통신 정보와 같은 옵션을 설정하는 기능을 담당한다. Validation 클래스는 불량판정을 결정하는 기능을 담당한다.

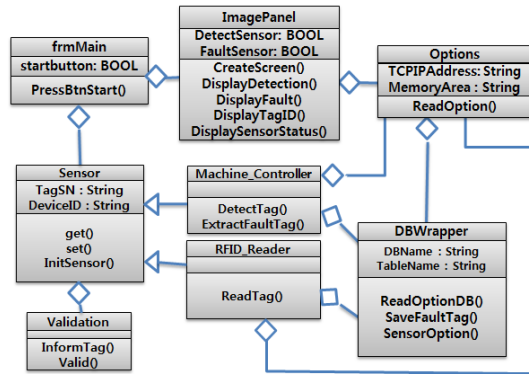


그림 3. 태그 불량 검출 시스템의 클래스 다이어그램
Fig. 3 Class diagram of tag defect detection system

클래스 다이어그램이 완성되고 나면 UML의 시퀀스 (Sequence) 다이어그램을 개발한다. 시퀀스 다이어그램은 RFID 불량검출 시스템 내의 객체들 사이의 메시지를 보여준다. 그림 4는 RFID 불량검출 시스템에 대한 시퀀스 다이어그램을 보여준다.

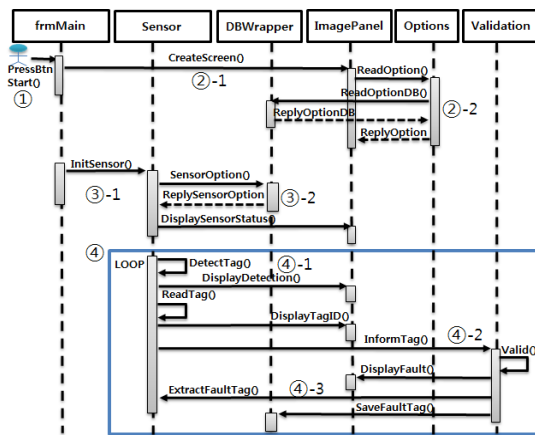


그림 4. RFID 불량 검출 시스템에 대한 시퀀스 다이어그램
Fig. 4 Sequence diagram of RFID defect detection system

① 테스트 시작

RFID 불량 태그에 대한 테스트를 시작하기 위한 소프트웨어 버튼이나 하드웨어 장치 버튼을 선택하는 PressBtnStart() 메소드를 호출한다.

② RFID 불량 태그 테스트를 위한 주요 기능 설정하기

②-1. 불량 태그 감지에 대한 화면 설정을 위해 Image

Panel 객체의 CreateScreen() 메소드를 호출한다.

②-2. DBWrapper 객체의 ReadOption 메소드와 Option 객체의 ReadOptionDB() 메소드를 호출하여 데이터베이스로부터 태그 메모리 타입과 정보를 읽는다.

③ 센서 초기화 하기

③-1. frmMain 객체에서 Sensor 객체의 InitSensor() 메소드를 호출하여 센서 초기화를 시도한다.

③-2. DBWrapper 객체의 SensorOption() 메소드를 호출하여 초기값을 읽어 센서 초기화를 시도한다. ImagePanel 객체의 DisplaySensorStatus() 메소드를 호출하여 화면에 값을 출력한다.

④ 센서에서 RFID 태그 정보를 읽은 뒤, 불량 판단하기

④-1. Sensor 객체의 DetectTag() 메소드로 RFID 태그를 감지하고 ImagePanel 객체의 DisplayDetection 메소드로 RFID 태그 감지 정보를 화면에 출력한다. Sensor 객체의 ReadTag() 메소드로 RFID 태그를 읽어들이고 ImagePanel 객체의 DisplayTagID() 메소드로 화면에 태그 ID 값을 출력한다.

④-2. 읽은 태그 정보를 근거로 Validation 객체의 Valid() 메소드를 통해 불량여부를 결정하고 ImagePanel 객체의 DisplayFault() 메소드를 통해 화면에 결정된 불량여부 정보를 출력한다.

④-3. 불량 RFID 태그를 추출하기 위해 Sensor 객체의 ExtractFaultTAG() 메소드를 호출하고 데이터베이스로 불량 정보를 저장하기 위해 DBWrapper 객체의 SaveFaultTag() 메소드를 호출한다.

3.2. LabVIEW OOP 클래스 생성

UML로부터 클래스 다이어그램 명세서가 만들어지면, LabVIEW OOP 클래스로 1:1 매핑을 한다. 그림 5는 UML로 설계된 불량 태그 검출 시스템을 예제로 사용하고 있다. UML의 클래스 다이어그램과 클래스 다이어그램을 기초 데이터로 하여 만들어진 LabVIEW OOP에서 생성한 클래스를 보여주고 있다. 그림 5에서 생성된 LabVIEW OOP의 클래스는 UML의 클래스 다이어그램과 정확히 1:1 매핑이 된다.

생성된 클래스 다이어그램으로부터 LabVIEW OOP의 클래스인 frmMain, Sensor, RFID_Reader, Machine_Controller, Validation, ImagePanel, Options, DBWrapper를 생성한다.

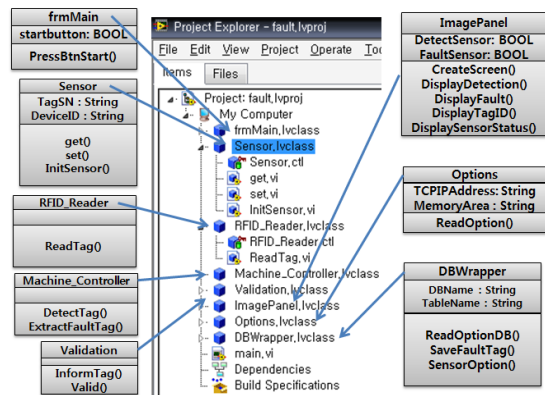


그림 5. 클래스 다이어그램과 LabVIEW OOP 클래스의 관계
Fig. 5 Relation between class diagram and LabVIEW OOP class

Sensor.lvclass는 클래스 이름을 의미하고 Sensor.ct는 클래스 프라이빗 데이터를 의미한다. get.vi, set.vi, InitSens.vi는 클래스에 포함된 메소드를 의미한다.

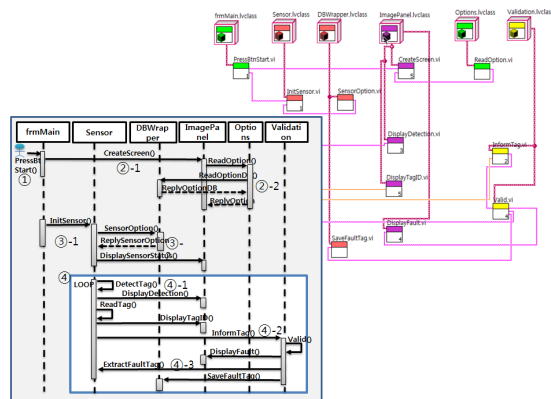


그림 6. 시퀀스 다이어그램과 LabVIEW OOP의 와이어링 관계
Fig. 6 Relation between sequence diagram and LabVIEW OOP wiring

UML로부터 시퀀스 다이어그램 명세서가 만들어지면, LabVIEW OOP 클래스를 이용하여 와이어링(wiring)을 시작한다. 그림 6은 UML로 설계된 불량 태그 검출 시스템을 예제로 사용하고 있다. UML의 시퀀스 다이어그램과 시퀀스 다이어그램을 기초 데이터로 하여 만들어진 LabVIEW OOP에서 생성된 와이어링을 보여주고 있다. 그림 6에서 생성된 LabVIEW OOP의

와이어링은 UML의 시퀀스 다이어그램과 정확히 1:1 매핑이 된다. ①의 PressStart() 함수는 ImagePanel 객체의 CreateScreen()(②-1) 함수를 호출하게 된다. 이를 LabVIEW OOP에서의 와이어링을 살펴보면, PressBtn Start.vi(메소드)와 ImagePanel.lvclass의 CreateScreen.vi(메소드)와 와이어링이 되어 있는 것을 확인할 수 있다.

IV. 프로그래밍 교육효과 설문 조사

4.1. 프로그래밍 적용 교육과정

이 논문에서는 RFID의 FA(Factory Automation) 프로그램 개발에 대하여 교육하고, LabVIEW OOP 방법과 JAVA로 개발하는 것을 실험하였다. 5일의 교육을 통해 학생들에게 예제를 사용하여 교육을 실시하였다. 교육 실시 후, 설문조사를 실시하였다. 그림 7에서 교육 과정을 도식화 하였다.

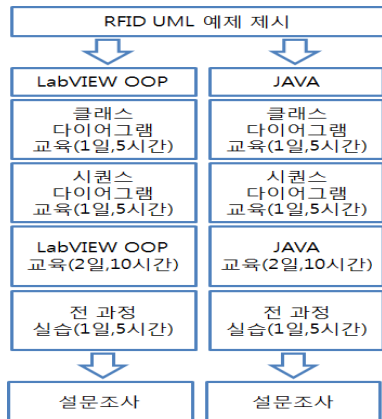


그림 7. 교육 적용 과정
Fig. 7. Course of education

4.2. 교육효과 적용에 대한 설문 결과

설문 조사 대상은 일반 IT 전공학과 3학년, 4학년(C언어, JAVA 가능) 40명을 대상으로 조사를 하였다. UML과 JAVA를 잘 모르는 학생들을 대상으로 미리 사전 교육을 5일 정도 실시하여 클래스 다이어그램, 시퀀스 다이어그램, 유스케이스 다이어그램을 이해하고 기본적인 것을 작성하는 수준이다. 실험 대상의 30% 정도는 JAVA나 C를 개념적인 이해가 부족한 학생으로 판

단된다.

표 1. 5점 척도 기준의 평균분석
Table. 1 Average analysis of 5-point degree

평가 항목	UML		LabVIEW OOP		JAVA	
	평균	표준 편차	평균	표준 편차	평균	표준 편차
요구사항 분석						
클래스 생성을 위한 요구 분석 이해도	4.75	0.43	1.00	0.81	0.63	0.86
분석된 클래스의 도식화	4.88	0.33	2.50	0.95	2.00	1.24
분석된 클래스의 호출관계	4.88	0.33	2.50	0.87	2.00	0.84
프로그램 설계						
클래스와 객체의 상관성 및 상호작용에 대한 이해도	4.50	0.67	4.25	0.86	3.00	1.30
객체의 속성 값에 대한 이해 정도	3.75	0.80	3.00	1.47	2.50	1.50
메소드의 호출 및 기능에 대한 이해정도	4.63	0.58	3.75	1.32	3.50	1.38
프로그램 구현(코딩)						
클래스 시각 표현	4.75	0.43	4.75	0.43	1.00	1.30
객체의 실제화의 용이성	2.50	0.77	4.50	0.77	2.50	1.64
도구/환경 사용의 용이성	2.50	0.95	2.50	1.26	2.00	1.38
클래스 구현과 인터페이스에 대한 이해도	3.00	1.00	4.50	0.81	2.50	1.53

이 논문에서 설문 조사한 3가지 교육방법(UML, LabVIEW OOP, JAVA)에서 설문 문항별로 만족도의 집단 평균치에 차이를 자료 분석하여 표 1에서 정리하였다. 설문조사에 포함된 평가항목들은 프로그램 분석 단계, 설계단계, 구현단계별로 구분하여 분류하였다. 각 평가 문항의 분석에서 설문 문항별로 UML, LabVIEW OOP, JAVA의 평균값을 표에서 표시하였다. 자료 분석은 SPSS Statistics 19를 사용하였으며 5점 척도의 채점표 분석에 평균분석을 이용하였다. 채점표의 5점 척도로 '매우 쉽다' 5점, '쉽다' 4점, '보통' 3점, '조금 어렵다' 2점, '매우 어렵다' 1점으로 하였다. 실증적

분석 방법은 일원분산분석법을 활용하여 분석하고 신뢰도 95%의 범주에 포함되면 유의성 있다고 해석하였다. 어떤 교육방법들이 평균차이를 보이는지에 대한 사후검증을 위해 Scheffe를 사용하였다.

4.3. 프로그래밍 교육효과 분석 결과

설문조사 결과인 표1의 평가항목들을 분석 단계, 설계 단계, 구현 단계별로 세 가지 교육방법(UML, LabVIEW OOP, JAVA)의 평균값을 분석하여 다음과 같은 결론을 얻었다.

(1) 프로그램 분석 및 설계 단계

컴퓨터 프로그램을 개발할 때 요구 분석 단계 및 프로그램 설계 단계에서는 시각적으로 표시되는 UML과 같은 객체지향 설계 도구에서 이해도가 높은 것으로 분석되었다.

(2) 프로그램 구현 단계

프로그램 구현 단계에서는 UML 모델링 결과인 클래스 구조도, 시퀀스 다이어그램과 유사한 구조와 1:1 매핑되는 프로그램 구조를 가지는 객체지향 시각 프로그래밍 언어인 LabVIEW OOP를 사용한 프로그래밍이 교육적으로 효율적이라는 결과가 나왔다.

V. 결 론

프로그래밍 언어 교육 시, 논리적인 사고를 텍스트가 아닌 시각적인 방식으로 표현하여 교육적인 효과를 높이는 것은 매우 중요하다. 이 논문에서는 논리적 사고를 표현하기 위해 체계화되고 검증된 객체지향 소프트웨어 설계도구인 UML을 통해 프로그램에 대한 논리적인 사고를 효율적으로 표현하고, LabVIEW OOP와 같은 객체지향 시각 프로그래밍 개발 도구를 이용하여 구현하는 교육을 시행하였다. 실제적인 적용 예에서는 UML과 LabVIEW OOP를 접목하여 좀 더 효율적이고 직관적인 내장형 시스템을 설계 및 구현하기 위한 적용 예로서 RFID 불량 검출 시스템에 적용하였다.

이 논문에서 제안한 교육 방법론을 적용한 결과로서, 프로그램 분석 및 설계 단계에서는 UML과 같은 객체지향 개념을 가지는 도구로 먼저 개념을 도식화하고,

구현 단계에서는 LabVIEW OOP와 같은 시각적 프로그래밍 도구로 도식화된 개념을 프로그래밍하는 교육 방안이 가장 효과가 높았다.

향후 연구에서는, UML을 사용하여 설계된 클래스 다이어그램과 시퀀스 다이어그램이 LabVIEW OOP의 클래스와 블록 다이어그램으로 구현되는 좀 더 구체적인 교육 방안에 대한 연구가 필요하다. 이와 관련하여 UML의 클래스 다이어그램과 시퀀스 다이어그램에서 LabVIEW OOP의 클래스와 다이어그램으로 자동화 프로그래밍되는 주제에 대한 연구, 조사가 필요하다.

감사의 글

이 논문은 2014년도 동의대학교 연구비를 지원받았음(과제번호 : 2014AA438)

REFERENCES

- [1] Y-U Lho, M-P Jung, H-G Cho, and D-G Jung, "A Development and Analysis of the Assessment Metrics for Text/Visual Programming Education Using LabVIEW," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 16, no. 11, pp.2544-2549, 2012.
- [2] D-G Jung, M-P Jung, H-G Cho, and Y-U Lho, "A Development of the Evaluation Metrics and Analysis of the Object-Oriented Visual Programming Education Using Alice Programming," *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 18, No. 3, pp.742~748, 2014.
- [3] T. Levendovszky and L. Lengyel, "A UML class diagram-based pattern language for model transformation systems," *World Scientific and Engineering Academy and Society (WSEAS)*, 2005.
- [4] "UML(Unified Modeling Language)", [Internet]. Available: <http://www.uml.org/>
- [5] OMG, *UML 2.0 Infrastructure-Final Adopted Specification*, OMG document ptc / 03-09-15, 2003.
- [6] M. Fowler M and K. Scott K, *UML Distilled* (3rd Ed.), Addison Wesley, 2004.

- [7] National Instrument, *LabVIEW*, [Internet]. Available: <http://www.ni.com/labview/>
- [8] *Tutorial - LabVIEW Object-Oriented Programming: The Decisions Behind the Design*, [Internet]. Available: <http://zone.ni.com/devzone/cda/tut/p/id/3574>, 2009.
- [9] W.-K. Jehng, "Using LabView to Integrate RFID system and Database for Supply Chain Efficiency Improvement," *International Journal of Intelligent Control and Systems*, vol. 13, no. 3, pp.189-195, 2008.
- [10] D. Leffingwell and D. Widrig, *Managing Software Requirements: A Use Case Approach*, Pearson Education, 2003.
- [11] D. Bell, *UML's Sequence Diagram*, IBM, 2004.
- [12] K Taniguchi, *Extracting sequence diagram from execution trace of Java program*, Principles of software Evolution, 2005.



정덕길(Deok-Gil Jung)

1983년 부산대학교 이학사
1986년 서울대학교 이학석사
1994년 서울대학교 이학박사
1986년~현재 동의대학교 컴퓨터과학과 교수
※ 관심분야 : 프로그래밍언어, 컴퓨터교육, 임베디드 프로그래밍, RFID 응용 소프트웨어



정민포(Min-Po Jung)

1996년 울산대학교 공학석사
2011년 동의대학교 이학박사
1999년~현재 영산대학교 사이버경찰학과 교수
※ 관심분야 : 실시간 소프트웨어 설계, RFID 응용 소프트웨어, 보안 소프트웨어, 인공지능



조혁규(Hyuk-Gyu Cho)

1988년 부산대학교 이학사
1990년 부산대학교 이학석사
2009년 부산대학교 이학박사
2003~현재 영산대학교 사이버경찰학과 교수
※ 관심분야 : 한국어정보처리, 정보검색, 컴퓨터교육, 디지털포렌식



노영욱(Young-Ugh Lho)

1985년 부산대학교 이학사
1989년 부산대학교 이학석사
1995년 부산대학교 이학박사
1989년~1996년 한국전자통신연구원(ETRI) 연구원
1996년~현재 신라대학교 컴퓨터교육과 교수
※ 관심분야 : 내장형시스템, 멀티미디어시스템, 컴퓨터교육, 비주얼 객체지향 언어, 클라우드 컴퓨팅