

# 소프트웨어 개발 프로세스 교육을 위한 웹 기반 로그 시스템의 구현 및 분석

박지연<sup>†</sup> · 유욱성<sup>††</sup>

## 요 약

PSP(Personal Software Process)는 소프트웨어 개발자들이 양질의 코드를 생산할 수 있도록 개발된 정밀하면서도 계량적인 수치를 기반으로 한 소프트웨어공학 기법이다. PSP의 효과적 학습을 위해서는 프로젝트에서 발생하는 많은 양의 데이터를 계산하고 기록하는 작업을 필수적으로 행해야만 한다. 이 작업은 많은 시간이 소요될 뿐 아니라, 오류의 발생이 쉬워 PSP 학습의 동기과 효과를 감소시키는 결과를 초래한다. 이를 개선하기 위해서 본 연구에서는 데이터의 정리, 검증 및 분석을 처리할 수 있도록 웹 기반의 PSP 로그 시스템을 설계하고 구현하였다. 3년간의 실험 결과는 이 시스템이 생산성과 정밀도를 높일 뿐만 아니라, 데이터의 처리보다는 PSP 기법의 프로세스에 대한 학습과 교수에 더 집중할 수 있게 해주었음을 보여주었다.

키워드 : 퍼스널 소프트웨어 프로세서, 웹 기반 로그, 소프트웨어 공학

## Analysis and Implementation of Web Based Log System for The Education of Software Developing Process

Chiyeon Park<sup>†</sup> · Wook-Sung Yoo<sup>††</sup>

### ABSTRACT

The Personal Software Process(PSP) is a rigorous, measurements-based process designed to help software developers produce high-quality codes. Learning PSP requires substantial and tedious record keeping including clerical computations with plenty of data occurred at every project. The required task reduces the educational effect and motive by time consuming characteristic and easy occurrence of errors. To alleviate these problems, a web-based PSP log system having streamlined mechanism of data manipulation is designed and implemented at this study. Three years of experiment shows that the system not only improves productivity and accuracy of the process but also allows students and instructors to focus on learning and teaching PSP, rather than PSP data collection.

Keywords : Personal Software Process, Web Based Log, Software Engineering

### 1. 서 론

PSP(Personal Software Process)기법은 초급수준의 소프트웨어 개발자들이 양질의 코드를 생산할 수 있는 좋은 습관을 가질 수 있도록 교육하기 위해 개발된 정밀하면서도 계량적인 수치를

<sup>†</sup> 정 회 원: 관동대학교 컴퓨터학과 교수 (교신저자)  
<sup>††</sup> 비 회 원: 미국 Gannon University 컴퓨터정보학과 교수

기반으로 하는 기법이다[1]. 많은 교육기관과 산업현장에서 이 기법을 교육하고 있으며[8], Gannon 대학교 컴퓨터정보학과와 학부와 석사과정에서도 소프트웨어공학과 관련된 주요과목중의 하나로 채택하고 있다.

PSP의 기본 목표는 학생들에게 소프트웨어를 개발하는 과정에서 정형화된 프로세서를 채택하게 함으로써 일정 및 품질관리에 근거한 전문적인 접근방식을 기르고, 이를 습관화하게 하는 것이다. 여러 단계로 구성된 일련의 과정을 거친 학생들은 설계나 코딩에서 결함의 조기 발견과 방지 등을 통하여 생산성 및 품질 향상과 개발계획의 일정을 정밀하게 예측할 수 있는 능력을 갖추게 된다[9, 10].

PSP의 수많은 장점에도 불구하고 학생들이 채택하길 꺼려하는 가장 큰 요소는 각 프로세스에서 코드(code)의 크기, 시간, 품질 데이터 등과 관련된 수치를 지속적으로 기록하고 정리하여야 한다는 데에 있다[2, 6, 11]. 일반적으로, PSP를 학습하는 교과목에서 하나의 과제를 완수하기 위해서는 약 500개의 수치 데이터를 추적하면서 기록해야한다. 이는 한 학기 중에 약 10개의 과제를 수행하면서 약 5,000개의 수치 데이터를 계산하고 기록해야 한다는 것을 의미한다. 교수자도 이 프로세스의 효과를 높이기 위하여 주기적으로 모든 학생들의 기록 데이터를 검증해야 한다. 우리의 경험에 의하면 이를 검증하기 위해서 교수자는 한 학기에 각 학생당 약 2~8시간을 소모해야 하는 것으로 관찰되었다.

본 연구에서는 이러한 문제점을 해결하고 학습자 및 교수자가 PSP를 적극적이고 효율적으로 활용할 수 있도록 웹 기반의 로그 시스템을 설계하고 구현한 후, 그 효과를 분석하였다.

## 2. 관련 연구

PSP 프로세서는 일련의 소프트웨어 개발 과제를 수행하면서 진행된다. 각 과제는 단계(level)별로 세부적인 프로세스로 세분되며, 각 프로세서는 정해진 항목에 따라 정밀하게 정의된 수치 데이터에 의해 관찰하게 되어 있다[1, 3]. 이러한 수치 데이터들은 소프트웨어 개발과정에서 학생

들이 자신의 기록을 근거로 수집한다. PSP의 학습과정에서는 필연적으로 다량의 수치 데이터 계산과 기록이 발생하게 되었으며, 이러한 문제점을 최소화하기위해서 현재까지 여러 종류의 지원 도구가 연구되어 소개되었다[4, 5, 7].

본 연구에서 구현한 PSP 로그 시스템(PSP log system)을 포함하여 PSP를 지원하는 도구들의 대표적인 특징을 나타내면 <표 1>과 같다. 상업용 도구들은 고가의 가격으로 인하여 대부분의 교육환경에서는 채택하기가 용이하지 않으므로 이 비교에서는 제외하였다.

<표 1> PSP 지원도구들의 특징 비교

특징	PSP 지원 도구			
	PSP 로그 시스템	Process Dashboard [7]	PSP Studio [5]	Support PSP [4]
Type	Web Based	Stand-alone	Stand-alone	Spreadsheet
Data Storage Method	Web accessible database	Local Files	Local Database	Local Files
Platform	Windows	Windows, Linux	Windows	All
User Support	Multiple User	Single User	Single User	Single User
PSP Forms	All	All	All	All
Editable Forms	All	All	All	All
Chart Flexibility	Yes	No	No	N/A
Purpose	Teaching	Real World	Teaching	Either
Interface	Simple Interface	Simple Interface	Complex Interface	No Interface
Metrics Calculation	Automatic	Automatic	Automatic	Manual
PSP Levels	All Levels	All Levels	All Levels	Basic PSP
Instructor Support	Yes	No	No	No
User Help	Included	Included	Manual Only	Manual Only

<표 1>에서 보이는 4가지 지원 도구는 공통적으로 PSP에서 사용되는 모든 양식(form)과 단계(level)의 전체 또는 일부를 지원한다. 기존의 도구와 PSP 로그 시스템의 주요 차이점은 운용 형태, 용도, 사용자 수, 분석 기능, 및 교수자 지원 기능이라는 특징에서 찾아볼 수 있으며, 기존의 시스템은 다음과 같은 단점을 가지고 있다.

첫째, 기존의 자립형 시스템(stand alone)은 교

육목적의 환경에서는 그 효율성이 떨어진다. 예를 들어, 시스템을 사용하는 학생들은 각 과제를 수행하면서 수집한 데이터를 파일등으로 저장하여 교수자에게 별도로 제출 또는 전송해야만 한다.

둘째, 교수자에 대한 지원과 분석기능이 없는 기존의 시스템에서는 교수자가 개개의 파일을 온라인(on-line) 또는 오프라인(off-line)으로 자신의 시스템에 적재한 후, 일일이 각각의 데이터에 대해 검증하고 별개의 툴(tool)을 이용해 분석해야만 한다.

셋째, 기존의 시스템은 한 과제가 완성된 후에야 학생들의 데이터를 검증하게 됨으로써 과제를 진행하는 과정에서 발생하는 오류를 조기에 지적할 수 있는 기회를 잃는 것이다. 일반적으로 PSP 프로세스에 미숙한 학생들에게서 쉽게 발생하는 데이터 오류는 다음 단계의 과제로 진행하기 전에 모두 수정되어야만 한다. 경험에 의하면 이는 시간적인 소모뿐만 아니라 학생들에게 이 기법에 대한 학습의욕마저 저하시키는 경향이 있다[3].

즉, 기존의 지원도구는 학습자 개개인의 독립적인 훈련에 그 목적을 둬으로써 교육기관의 환경에서 필요한 학습자와 교수자의 상호작용에 요구되는 기능은 대체적으로 미약하거나 배제되어 있는 것을 알 수 있다.

본 연구에서 제안하는 PSP 로그 시스템은 교육 목적의 환경이라는 관점에서 기존 도구들의 여러 단점을 극복할 수 있도록 웹 기반의 시스템을 구현함으로써, 학습자와 교수자에게 분산적 시스템을 제공하고 상호간에 조기 피드백(feedback)이 가능하게 하는데 주 목적을 두었다.

### 3. PSP 로그 시스템의 구성 및 기능

PSP 로그 시스템은 크게 5가지의 요소로 구성되어 있다. 이 중 4개 요소(PSP 온라인 폼, 교수자 지원 도구, PSP 차트, Q/A)는 학습자와 교수자에 대한 PSP의 직접적 기능과 관계된 것이며, 나머지 1개 요소(DB)는 각 요소들을 결합하여 필요한 기능을 제공하기 위한 중앙의 데이터베이스(database)이다.

스(database)이다.

#### 3.1 PSP 온라인 폼

PSP 온라인 폼은 PSP 기법의 학습과정에서 발생하는 필수 데이터를 처리하는 핵심 기능이다. 시스템에 로그인(login)한 모든 학생들은 현재 부여된 과제에서 지정하는 PSP 레벨에 해당되는 PSP 폼에 접근하여 PSP의 각 프로세서를 진행하면서 측정된 수치 데이터를 입력 또는 갱신할 수 있다. 이 기간 중에 교수자는 언제라도 데이터를 검증할 수 있으며, 오류가 발견되면 이에 대해 지적하여 수정을 요구할 수 있다. 과제가 완료되면 프로세서에서 발생한 수치들을 종합하여 분석을 할 수 있다.

이 시스템은 PSP에서 정의하는 여섯 단계의 레벨에 대한 폼과 스크립트(script)를 지원할 수 있도록 설계되었다. 수준별 레벨로 구성된 폼과 스크립트는 PSP에서 목표로 하는 개념과 테크닉을 점진적으로 익힐 수 있도록 개발되었다. <표 2>는 시스템에서 제공하는 PSP의 레벨과 각 레벨이 포함하는 폼 및 스크립트를 나타낸다.

<표 2> PSP의 각 레벨의 폼 및 스크립트

Forms and Scripts	PSP Level						
	0	0.1	1	1.1	2	2.1	3
Project Plan Summary	0	0	0	0	0	0	0
Time Recording Log	0	0	0	0	0	0	0
Defect Recording Log	0	0	0	0	0	0	0
Process Improvement Postscript		0	0	0	0	0	0
Test Report Template			0	0	0	0	0
Size Estimating Template			0	0	0	0	0
Task Planning Template				0	0	0	0
Schedule Planning Template				0	0	0	0
Operational Scenario Template							0
Functional Specification Template							0
State Specification Template							0
Logic Specification Template							0
Issue Tracking Log							0

학생들에게 부여되는 과제는 특정 PSP 레벨과 연계되어 있으며, 과제가 진행될수록 고수준의 레벨과 연계되고 고려해야할 데이터의 수도 증가하게 되어있다. 시스템에는 각 개발과제를 완성해가는 동안 데이터의 정확성을 높이기 위하여

데이터 입력 항목에는 팝업 윈도우(pop-up window)로 이루어진 온라인 도움말 기능을 제공한다.

<그림 1>은 시스템에서 제공하는 양식의 한 예로 계획 요약(plan summary) 폼의 내용이다. 이 폼은 PSP 레벨 2.1에 해당하며 계획 및 실제 소요시간을 포함한 다양한 데이터를 나타낸다.

Plan Summary 2.1			
Summary	Plan	Actual	To Date
LOC/Hour	6.65	10.67	28.6
Planned Time	205		2120
Actual Time		225	2188
CP(Cost-Performance Index)			0.97
% Reused (R/T)*100	409.16	36.36	14.11
% New Reused (TN/RA)*100	0	0	0
Test Defects/KLOC	34.22	0	14.30
Total Defects/KLOC	239.56	126	60.81
Yield %	100	0	3.77
% Appraisal COQ	9.76	13.33	7.77
% Failure COQ	14.63	13.33	14.06
COQ A/F Ratio	0.67	1	0.56
<b>Program Size(LOC):</b>	<b>Plan</b>	<b>Actual</b>	<b>To Date</b>
Base(B)	310		310
Deleted(D)	0		0
Modified(M)	0		0
Added(A)	29		40
Reused(R)	200	200	239
Total New & Changed (N)	29.22	40	1043
Total LOC(T)	539.22	590	1694

<그림 1> PSP Level 2.1의 Plan Summary

### 3.2 교수자 지원도구

교수자 지원도구는 학생관리, 프로젝트관리 및 평가관리를 위한 도구로 구성되어 있다.

#### 3.2.1 학생 관리 도구

교수자는 학생들의 계정을 생성 또는 삭제하거나 아이디(ID)와 패스워드(password)를 설정할 수 있다. 또한 교수자는 필요에 따라 이 과목의 진행과 관리를 위해 공동 교수자나 보조자들에게 권한을 부여할 수도 있다.

#### 3.2.2 프로젝트 관리 도구

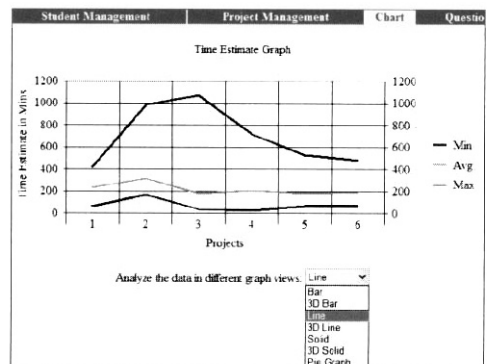
교수자는 각 프로젝트 과제에 대해 적절한 PSP 레벨을 지정할 수 있다. 프로젝트에 레벨이 지정되면 이에 대응하는 폼과 스크립트는 자동으로 생성되어 제공된다.

#### 3.2.3 평가 관리 도구

평가관리 도구는 교수자가 학생들의 프로젝트를 여러 관점에서 관찰하고 평가할 수 있도록 지원한다. 교수자는 각 학생들에 의해 입력된 PSP 데이터를 프로젝트가 완료되기 전에도 검증할 수 있으며, 최종적으로 완료된 이후에는 평가를 할 수 있다. 이 과정에서 교수자는 해당 프로젝트가 진행되는 도중에도 현 상태에 대한 문제점등을 온라인상에서 지적할 수 있으므로 조기에 학생들에게 적절한 피드백을 보낼 수 있다. 프로젝트가 완료되면 요약 기능과 차트 기능을 이용하여 강좌 전체 학생들의 PSP 데이터에 근거한 성취도를 분석할 수 있다.

### 3.3 차트 도구

각 프로젝트의 개발과정에서 발생한 데이터의 분석은 학생들에게 소프트웨어 개발자로서의 발전과정을 이해하고 동기부여가 될 수 있는 중요한 요소이다[3]. PSP 로그 시스템은 그래프를 이용하여 학생들의 성과를 다양한 관점에서 가시화시킬 수 있도록 지원한다. 차트 도구는 계획시간, 소요시간, 결함로그, 결함시간 비율, 결함크기 비율, 라인당 결함수, 프로젝트 대비 컴파일 및 테스트 비율, 시간당 결함수 등과 같은 측정치에 대해 선그래프, 막대그래프 및 3차원 그래프를 제공한다.



<그림 2> 지원하는 분석 차트의 예

프로젝트를 거치면서 학생들은 분석차트를 통해 각 항목에 대해 자신의 개발 능력이 발전해가는 과정을 검토할 수 있고, 전체 학생들의 종합적인 데이터와 비교하여 문제점을 파악하고 자신이 어느 수준의 성과를 이루는지를 확인할 수 있다.

#### 4. 실험결과 및 분석

시스템을 설계하고 구현한 후, 이 교과목에서 3년 동안에 수집된 데이터와 설문을 통한 학생들의 반응을 분석하였다. 시스템의 효율성을 측정하기 위해서 학생들을 두 그룹으로 분리하여 실험하였다. 한 그룹의 학생들은 개발 과정에서 PSP 로그 시스템을 사용하게 하였고, 다른 그룹의 학생들은 제시된 양식에 따라 기존의 방식대로 데이터를 수집하게 하였다. 실험의 객관성을 높이고 학생 개인의 특성에 따른 영향을 배제하기 위해서 총 10개의 프로젝트 중 5개의 프로젝트를 마친 후에는 두 그룹의 역할을 서로 바꾸었다. 이러한 비교 결과는 앞에서 소개한 다른 도구를 사용한 시스템[4, 5, 7]에서는 아직 연구되어 소개된바가 없다.

3년 동안 3학기에 걸쳐 학기당 10개의 프로젝트를 진행하면서 학생들이 소요한 시간 중 각 프로젝트에서 데이터 정리 및 기록 단계(postmortem phase)에서 걸린 시간을 요약하면 <표 3>과 같다.

<표 3> 데이터 정리 및 기록 시간 비교

	1차년도	2차년도	3차년도
PSP Log System	18 min.	25 min.	17 min.
Paper Form	34 min.	42 min.	33 min.
Improvement	47.1%	40.5%	48.5%

<표 3>에서 볼 수 있듯이 PSP 로그 시스템은 기존방식에 비해 평균적으로 약 45%의 시간을 절약하여 PSP 프로세서의 생산성을 높일 수 있었다. 이는 학생들이 프로젝트를 완성해 가면서 PSP과정을 익히는 총 소요 시간에서 데이터 정

리 및 기록과정이 차지하는 비중이 상당히 감소한다는 것을 의미한다.

매 학기말에 학생들의 경험을 바탕으로 기존 방식과 PSP 로그 시스템과 비교하여 데이터 처리의 효율성과 학습 내용의 이해도의 관점에서 시스템에 대한 선호도 조사를 실시하였다. 이 조사에서 평균적으로 학생들의 86%가 기존방식에 비해 PSP 로그 시스템을 선호하는 것으로 나타났다.

<표 4> PSP 로그 시스템에 대한 선호도 비교

	1차년도	2차년도	3차년도
PSP Log System	80%	88%	91%
Paper Form	12%	4%	4%
No Preference	8%	8%	5%

<표 4>에서 보듯이 평균 약 7%의 학생들은 PSP 로그 시스템보다 기존의 방식을 선호하였다. 그 이유를 분석한 결과, 이 학생들은 프로젝트를 수행하는 동안 인터넷에 접근하기가 어려운 네트워크 환경에 있었던 것으로 조사되었다. 즉, 먼저 수기 또는 스프레드시트를 이용하여 기록한 이후에 다시 온라인 양식을 이용하여 재입력해야 했다. 실험결과를 보면 해가 갈수록 인터넷에 대한 접근성이 용이해짐에 따라 PSP 로그 시스템에 대한 선호도가 증가함을 알 수 있다. 학생들의 피드백에 따라 시스템의 기능을 지속적으로 업데이트(update)한 부분도 선호도를 높이는데 기여했다고 볼 수 있다.

이 시스템으로 인해 교수자의 시간적 부담도 급격히 감소한 것으로 관찰되었다. 첫째, 학생들의 입력데이터에 대한 검증과 평가에 소모되는 시간은 평균 약 31% 정도가 감소하였다. <표 5>는 10개의 과제를 수행할 경우 학생 1명당 소요된 교수자의 평균시간을 나타낸다.

<표 5> 검증 및 평가 소요시간 비교

	1차년도	2차년도	3차년도
PSP Log System	168 min.	115 min.	108 min.
Paper Form	209 min.	176 min.	182 min.
Improvement	19.6%	34.7%	40.7%

둘째, 학생 데이터의 분석 및 이를 가시화하는 그래프 작성에 걸리는 소요시간을 단축시켜주었다. 이 일은 단순하고 시간 소모적이지만 교육적인 효과가 큰 부분이다. 한 과제에 대해 전체 학생의 수행성과를 요약하기 위해서는 평균 27분 정도의 소요시간이 걸렸으나, 시스템을 이용할 경우 교수자는 시간을 거의 소모할 필요가 없었다. 또한, 이 시스템은 학생 및 교수자가 데이터가 입력된 파일을 전송하거나 다른 매체로 이동시키는 과정에 필요한 수고를 완전히 제거시켜주었다.

학생과 교수자의 소요시간을 감소시키는 정량적 효과뿐만 아니라, PSP 과정의 정성적 학습 효과도 크게 향상시켰다. 프로젝트를 완료하기 전에도 학생들의 데이터를 실시간으로 확인할 수 있으므로 교수자는 각 프로세스에서 학생들이 경험하는 어려운 점을 쉽게 파악할 수 있다. 대부분의 학생들은 PSP에서 필요로 하는 데이터 기록 과정에 대한 이해가 부족하므로, 잘못된 수치를 입력하는 경우들이 많이 발생한다[2, 3]. 신뢰성이 없는 데이터는 단계가 진행하면서 계속 영향을 미치게 되고, 결과적으로 전 과정의 데이터가 의미 없는 오류로 처리되게 되므로 되도록이면 빠른 시기에 이를 정정할 수 있어야 한다. 수기나 스프레드시트(spreadsheet)에 의존한 방식에 비해 이 시스템에서는 빠른 시기에 이를 발견하여 조기 수정이 가능하도록 해주었다.

부가적으로, PSP 기법에 확신을 가지지 못하거나 학습에 태만한 학생들의 고의적 오류 데이터의 입력도 조기에 발견하여 지적함으로써 이들의 참여도를 높이는 효과도 있었다. 비록 정량적으로 측정할 수는 없었으나 시스템에 내장한 온라인 도움말기능도 입력 데이터의 오류율을 감소시켰다고 믿어진다.

## 5. 결론

본 연구에서는 PSP 기법의 훈련에 필요한 필수적인 입력 및 검증 과정을 위해 교육기관의 환경이라는 목적에 적합하도록 효율적이면서 상호작용이 가능한 시스템을 설계하고 인트라넷 환경

에서 이를 구현하였다. 개발한 PSP 로그 시스템을 지난 3년 동안 정규 교과목에서 PSP 기법을 교육하면서 적용하고 그 결과를 분석하였다. 시스템에서 제공되는 학습자 개개인의 성취도 분석 기능 및 상호작용에 의한 오류의 조기 검증 기능 등은 시스템에 대한 학생들의 높은 선호도와 함께 PSP 기법의 학습 목적에 대한 확신 및 학습 동기의 부여라는 관점에서 효과적으로 기여했음을 관찰할 수 있었다. 또한, 이 시스템은 PSP 기법의 학습에 소요되는 학습자와 교수자의 시간과 수고를 감소시켜주었을 뿐만 아니라, 장기적으로는 PSP 로그 데이터의 정확성을 향상시키고 이 데이터들의 축적을 가능하게 함으로써 이를 반영한 강의 내용의 개선에 유용하게 활용할 수 있었다.

본 연구의 향후 계획으로는 온라인 입력데이터의 파일저장 및 호환기능, 향상된 온라인 도움말기능, 오류 가능 데이터의 자동 발견과 경고 기능 등의 추가이며, 이러한 기능들은 학습자들의 오류를 조기에 감소시켜 데이터의 신뢰성을 더욱 확보할 수 있을 것으로 본다.

## 참고 문헌

- [1] Humphrey, Watts S. (1995). A Discipline for Software Engineering, Addison Wesley.
- [2] Disney, A & Johnson, P. (1998). Investigating Data Quality in the PSP, Sixth International Symposium on the Foundations of Software Engineering(SIGSOFT'98), Orlando, Fl.
- [3] Lisack, S. (2000). The Personal Software Process in the Classroom: Student Reactions(An Experience Report), 13th Conference on Software Engineering Education & Training, Austin, Texas.
- [4] Postema, M., Dick, M., Miller, J. & Cuce, S. (2000). Tool Support for Teaching the Personal Software Process, Computer Science Education, 10(2).
- [5] Henry, Joel. (1997). PSP Studio User's Manual, ETSU.
- [6] Van Buren, Jim. (1999). A Review of

Personal Software Process(PSP) and Team Software Process(TSP), STSC.

- [7] Tuma, D. (1998). Process Dashboard, United Air Force.
- [8] Ramsey, M. (1996). Experiences teaching the Personal Software Process in academia and industry, Proceedings of the SEPG Conference.
- [9] Hayes, W. and Over. J. (1997). The Personal Software Process: An empirical study of the impact of PSP on individual engineers, Technical Report CMU/SEI-97-TR-001, Software Engineering Institute, PA.
- [10] Williams, L. (2000). The Collaborative Software Process, PhD thesis, University of Utah.
- [11] Borstler, J., Carrington, D., Hislop, G., Lisack, S., Olson, K. and Williams, L. (2002). Teaching PSP: Challenges and lessons learned, IEEE Software, 19(5).

## 박 지 연



1985 경희대학교 기계공학과 (학사)

1987 플로리다공대 전산학과 (석사)

1996 플로리다공대 전산학과(박사)

1997~현재 관동대학교 컴퓨터학과 교수

관심분야: 지능형 시스템, 웹 기반 교육

E-Mail: cypark@kwandong.ac.kr

## 유 옥 성



1982 서울대학교 치과대학 (학사)

1987 플로리다공대 전산학과 (석사)

1995 플로리다공대 전산학과(박사)

1996~현재 미국 Gannon University  
컴퓨터정보학과 교수

관심분야: 소프트웨어공학, 웹 기반 교육

E-Mail: yoo001@gannon.edu