

페이퍼 프로토타입 기반의 응용 프로그램 개발 교육방안 연구

최진용*, 손원성*

경인교육대학교 컴퓨터교육과*

요약

교육 패러다임의 전환과 함께 학교교육에서 응용 프로그램 개발교육의 중요성이 부각되고 있으나 여러 현실적인 문제로 인하여 학교 현장에서의 응용 프로그램 개발교육이 제대로 이루어지지 못하고 있는 실정이다. 본 연구에서는 초등학생들이 어려워하는 프로그래밍 코딩 단계를 대신할 수 있는 방법으로 페이퍼-프로토타이핑 기법을 적용하여 아날로그 방식으로 구현할 수 있도록 하였고, 개발자 중심의 개발이 아니라 사용자가 개발에 적극 참여하는 방식의 사용자 중심디자인 모델을 기반으로 하였다. 교육의 효과를 검증하기 위하여 학습자의 응용 프로그램 개발 전후의 개념 모형을 심성모형 이론을 적용하여 측정·분석하였다. 본 연구에서 제안한 프레임워크는 학교 현장에서 응용 프로그램 개발교육에 대한 새로운 접근 방법이 될 것으로 기대한다.

키워드 : HCI, 디자인 모델, 페이퍼 프로토타이핑, 응용 프로그램 개발

A Study on Education Methods to Develop Application Programs Based on Paper Prototyping

Jin-Yong Choi*, Won-Sung Sohn*

Gyeongin National University of Education, Dept. of Computer Education*

ABSTRACT

As the knowledge and information society has emerged, paradigm of school education has moved to improvement of creativity and problem-solving skills of a learner. However, up to the present, ICT education in schools has shown distorted features, which are focusing on the use of application programs, out of its original purpose. This study also allows digital technology to be displayed in analog ways by applying the paper prototyping technique, which can be substituted for the programming phase that elementary school students feel difficulty to handle with. In addition, It enables practical and useful ideas to be designed, constructed and evaluated as a system, based on User Centered Design (UCD), which encourages users actively to participate in the development, rather than focus on developer. To verify the effects of education, we evaluated and analyzed concept models of a learner before and after the application development activity by applying the mental model theory. The framework suggested in this study can be applied to courseware of programming in elementary schools.

Keywords: HCI, Design Model, Paper Prototyping, Develop Application Programs

본 연구는 한국연구재단 기초연구사업(331-2008-1-D00546)을 통하여 수행되었음.

교신저자 : 손원성, 경인교육대학교 컴퓨터교육과

논문투고 : 2009-07-28

논문심사 : 2009-08-20

심사완료 : 2010-01-26

1. 서론

21세기 지식 정보화 사회에서는 정보가 중요한 자원이 되고, 정보의 처리·가공에 의한 가치의 생산을 중요하게 여긴다. 따라서 이러한 사회에서 교육해야 할 내용은 스스로 문제를 해결하는 능력, 협동심, 비판적 사고력 및 창의적 사고력 등이 될 것이다. 학교교육의 패러다임도 학습자의 창의성과 문제해결력 향상으로 방향 변화가 요구된다[1].

그러나 지금까지 교육현장에서 행해진 정보통신 기술교육은 응용소프트웨어 활용에 지나치게 치우친 비정상적인 모습이었다. 응용소프트웨어 기능 위주의 교육은 학생 입장에서는 정보소양과 응용소프트웨어 활용 능력을 동일한 것으로 생각하게 만들어 정보통신기술을 이용한 정보의 생성, 처리, 분석, 검색 등에 관한 기초적인 정보소양 능력을 기르고, 학습 및 일상생활의 문제 해결에 정보통신기술을 적극 활용한다는 본래 교육의 목표를 달성하기 어렵게 만든다[2].

한편, 2005년 12월에 개정된 “초·중등학교 정보통신기술교육 운영 지침”에 의하면 미래지향적인 정보통신기술에 대한 교육이 이루어져야 한다고 방향을 제시하고 있으며 컴퓨터과학 요소의 강화를 위하여 ‘정보처리의 이해’영역을 추가하였다[3]. ‘정보처리의 이해’영역은 문제해결력을 향상을 목표로 초등학교 저학년에 해당하는 1단계에서부터 고등학교 1학년에 해당하는 5단계까지 알고리즘과 프로그램 개발교육의 내용을 체계적으로 제시하고 있으나 초등학교 단계에서는 많은 어려움으로 인하여 제대로 실시되지 못하고 있는 실정이다.

초등학교에서 응용 프로그램 개발교육이 제대로 이루어지지 못하는 가장 큰 이유는 프로그램 코딩의 어려움 때문이다. 프로그래밍이 대부분의 초등학교생들에겐 쉽지 않은 내용이기 때문에 프로그램 개발교육이 초등학교에서 이루어지기는 어렵다. 이를 위하여 EPL, 인플러그드 컴퓨팅 등 다양한 시도가 이루어지고 있으나 아직 초등학교 전학년에서 가능한 프로그램 개발 교육방안은 제시되지 못하고 있다.

이를 위하여 본 연구에서는 프로그램 코딩의 과

정을 거치지 않고 스스로 조작하여 응용 프로그램 개발 교육을 할 수 있는 방법을 모색하였다. 이를 위하여 제안 연구는 사용자가 개발의 과정에 적극적으로 참여하여 사용성을 검토하고 의견을 반영하는 사용자중심디자인(User Centered Design)모델 기반 페이퍼 프로토타이핑 기법[7]을 적용한 응용 프로그램 개발 교육을 위한 프레임워크를 제안한다.

또한 제안한 프레임워크를 실제 초등학교 현장에 적용하여 제안한 방법의 활용 가능성을 모색하고자 하며 학생들의 발달단계에 맞추어 실생활에서 쉽게 접할 수 있는 다기능 리모콘, 개인 홈페이지, 웹 게시판 시스템 등을 페이퍼 타이핑 기법으로 구현한다.

본 연구의 제안 방안을 통하여 학습자들은 프로그램 및 S/W 개발과 관련된 전과정을 쉽게 참여할 수 있었으며 이와 관련된 다양한 심성변화를 확인할 수 있었다.

그 결과 본 연구에서 제안하는 프레임워크를 적용하여 초등학교 과정에서 응용 프로그램 개발교육이 이루어진다면 그 동안 초등 컴퓨터교육에서 쉽게 접근하지 못하던 분야에 대한 새로운 방식의 접근 방법이 될 수 있을 것으로 기대한다.

2. 이론적 배경

2.1 페이퍼 프로토타이핑

프로토타이핑이란 개발자와 사용자들 간의 의사소통 효과를 증진시키기 위하여 취하는 시스템 디자인 기법으로 HCI 분야에서 흔히 사용되는 개발모델이다[7],[8]. 대부분 개발자와 사용자간의 심도있는 상호작용 전 과정을 분석하여 완성된 시스템을 만들기 이전에 시험적으로 그 시스템의 전체 또는 일부분을 시제품(Prototype)으로 만들게 되며, 그 결과에 따라 제품의 타당성 혹은 가능성을 확인하게 된다.

한편 페이퍼 프로토타이핑은 프로토타이핑의 일종으로 일반적인 프로토타이핑과는 상이한 특징을 포함한다. 일반적 프로토타이핑 과정에서 사용되는 코딩 형태의 S/W 혹은 시뮬레이션 등은 높은 수준

의 피델리티(fidelity) 기반 프로토타입이며 고수준의 시간과 난이도를 포함할 수 있다. 이와 비교하여 페이퍼 프로토타입은 주로 사용성을 검증하고, 재설계하는데 이용되는 중히 혹은 목업(mockup)형태의 낮은 피델리티 수준의 프로토타입핑 기법을 의미한다. 따라서 짧은 시간내에 적은 로드를 투여하여 개발대상에 대한 사용성 분석 및 문제의 분석 및 개선점을 파악해낼 수 있다[9],[10],[11],[12].

2.2 심성모형

사용자가 어떤 시스템을 처음 접했을 때 그 시스템에 대해서 불완전하고, 때로는 잘못된 이미지를 가질 수 있다. 이러한 이미지를 심성모형이라고 하며 ‘어떤 시스템에 대해 사용자가 가지고 있는 개념적인 이해’라고 정의한다[5],[6],[8].

사용자의 시스템에 대한 심성모형 즉, 사용하고자 하는 시스템에 대한 이미지 형성은 사용자가 그 시스템을 배우고, 익숙해지는데 중요한 역할을 한다. 사용자가 배우는 과정 없이 소프트웨어를 접하게 되면 사용자 나름대로의 심성모형을 형성하고 시스템을 접하게 된다. 초기에는 그 시스템에 대한 불완전하고 때로는 잘못된 심성모형을 가지게 되지만, 시스템에 익숙해짐에 따라 사용자의 정신 모형은 점차 정확하고 완전한 모습을 띄게 된다[4].

3. 응용 프로그램 개발교육을 위한 프레임워크 개발

3.1 개발의 원칙

본 연구에서는 응용 프로그램 개발 교육을 위한 프레임워크의 개발 원칙을 다음과 같이 설정하였다.

첫째, 각 단계마다 반복적인 사용자 평가와 피드백을 갖는다.

둘째, 초등학생들에게 쉽게 적용할 수 있도록 프로그램 코딩을 대신 할 수 있는 언플러그드 컴퓨팅의 요소를 접목한다.

셋째, 프로토타입핑 기법을 사용하여 사용자와의 의사소통을 높인다.

넷째, 지도교사의 역할을 주어 초등학생이 설계과정에서 범할 수 있는 오류를 미연에 방지한다.

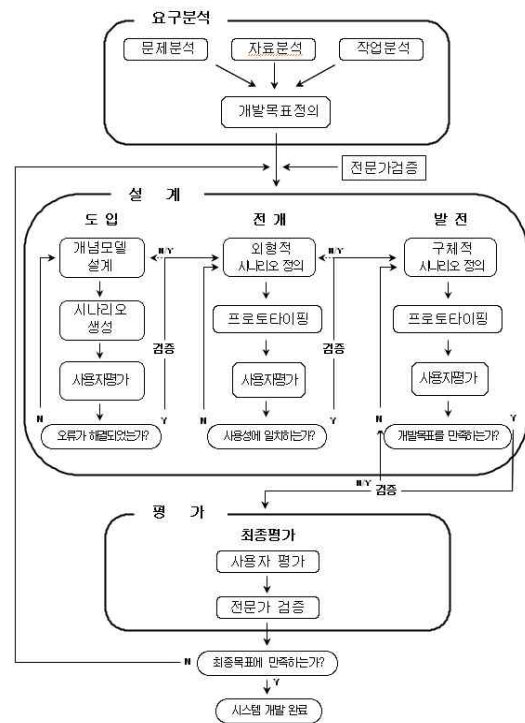
다섯째, 자유로운 재구성이 가능한 프레임워크를 개발한다.

3.2 프레임워크 제안

제시한 개발원칙에 따라 제안하고자 하는 프레임워크의 형태는 (그림 1)과 같다.

3.2.1 요구분석

요구 분석을 하는 것은 시스템 설계의 최초 출발이자 가장 중요한 이슈를 발굴하는 프로세스다. 개발하려는 응용 프로그램의 목표를 확립하는 과정이며, 개발에 들어가기 전에 문제에 대해 연구하는 것을 말한다[6].



(그림 1) 응용프로그램개발교육을 위한 프레임워크

가. 문제분석

우리가 접하는 많은 시스템은 사용자의 편의를 위해 개발되었지만 사용자의 사용성 보다는 개발자의 입장에서 개발된 경우가 많다. 따라서 사용자들은 시스템을 사용하면서 불편함을 느끼게 되고 이것이 바로 응용 프로그램 개발의 출발이 된다.

나. 자료분석

학생들이 개발하려는 소프트웨어는 기존의 타 소프트웨어의 단점을 개선하고 자신이 원하는 기능을 구현할 수 있는 것이어야 한다. 기존 시스템의 문제점을 찾아봄으로써 새로 설계하려는 시스템의 개발 방향과 목표를 설정할 수 있다.

다. 작업분석

이 단계에서는 가상의 시스템 사용 시나리오를 작성해본다. 주변 환경 속에서 일어날 수 있는 사건의 흐름에 맞춰 시나리오를 작성해봄으로써 발생할 수 있는 모든 경우의 수를 탐색하고 그에 적절한 기능이나 오류를 미연에 방지해 목표 설정에 방향을 잡는다.

라. 개발목표 정의

문제분석, 자료 분석, 작업분석의 단계를 거치면서 기존 소프트웨어의 문제점과 새로 설계하려는 소프트웨어에 사용자들이 바라는 요구사항이 밝혀졌다. 이를 정리하여 요구사항 명세서를 작성한다.

3.2.2 설계

충분한 요구분석 단계를 거쳐 개발목표를 정의하고 전문가의 검증을 받은 뒤 실제 설계에 들어가는 단계이다. 이 단계에서는 개발자와 사용자간의 활발한 상호의사소통을 통해 피드백을 주어 사용자 중심의 설계가 되도록 하는 것이 중요하다.

가. 도입

도입단계에서는 개념모델을 설계하고 1차적인 시나리오를 작성한다. 시나리오가 작성되었다면

시나리오를 매체로 사용자를 설계과정에 끌어들이 상호작용할 수 있도록 사용자의 평가와 검증을 받는다. 사용자의 평가를 통과한 개념모델은 다음 단계로 넘어가는 과정에서 전문가의 검증을 거치며 검증의 과정에서 오류가 발생할 경우 다시 개념모델 설계로 환원한다.

나. 전개

전개단계에서는 사용자의 이해를 돕기 위하여 대략적인 구체물 제작을 요구한다. 시스템의 화면 구현을 위해 카드를 사용한 프로토타이핑을 제안한다. 도입단계에서의 시나리오보다는 좀 더 구체적이고 시각적인 매체를 사용하여 사용성을 평가받게 된다. 평가결과 사용성에 일치하지 못하면 다시 전개 초반으로 피드백 되어 외형적 시나리오 작성부터 다시하게 된다.

다. 발전

발전단계에서는 실제 시스템과 동일한 구성과 논리전개를 구현할 수 있는 페이퍼 프로토타이핑 기법을 사용한다. 개발자는 실제 프로그래밍을 통해 소프트웨어를 제작하는 것과 동일한 입장에서 꼼꼼한 프로토타이핑을 작성하여 사용성을 검사받는다.역시 사용자 테스트 후에 전문가 테스트를 거쳐 다음 단계로 넘어간다.

3.2.3 평가

개발된 응용 프로그램의 최종 평가 단계이다. 요구분석 단계에서 정한 개발목표에 얼마나 부합되는 응용 프로그램이 개발되었는지 사용자와 전문가에 의해 최종평가를 받게 된다. 최종평가에서 오류가 발생하면 오류가 발생한 단계로 피드백하여 다시 수정의 과정을 거친다.

3.3 학습주제별 지도 계획

본 연구에서 제안한 사용자 중심 디자인기반의 시스템 개발 교육을 위한 프레임워크를 활용한 학습주제별 주요 학습 내용은 아래 표와 같다.

<표 1> 학습주제별 주요 설계 내용

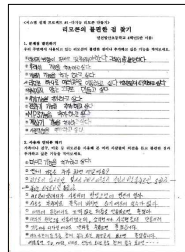
학습주제	주요 설계 내용
1 · 다기능 리모콘 시스템 설계하기	<ul style="list-style-type: none"> · 시스템 개발의 컨셉 · 그래픽 인터페이스 구현 · 하위 메뉴 구현방법 · 최소버튼으로 최대기능 구현 · 버튼의 효율적 배치
2 · 개인 홈페이지 개발하기	<ul style="list-style-type: none"> · 시스템 개발의 주제 설정 · 화면 인터페이스 구성 · 필요기능의 정의 · 인터페이스 기능 측면 보다 구성 체계 정리
3 · 인터넷 서비스 시스템 설계하기 (사진앨범, 동영상 재생기, 뮤직박스 등)	<ul style="list-style-type: none"> · 시스템 개발의 컨셉 · 화면 인터페이스 구성 · 간결, 깔끔한 디자인 구성 · 필요기능의 정의 · 기존 인터넷의 한계 극복

사용자 중심 디자인 기반의 응용 프로그램 개발 교육 방법 연구를 위하여 인천광역시 서구에 소재하고 있는 ○○초등학교 4학년 1개 학급을 선정하여 학생 27명을 대상으로 3개월간 제안한 프레임워크를 활용하여 응용 프로그램 개발 교육을 실시하였다.

4.2 단계별 교수·학습활동의 실제

가. 요구분석 단계

요구분석 단계에서는 문제분석, 자료분석, 작업분석, 개발목표 정의 활동이 이루어진다.



(그림 2) 문제분석 및 자료분석

<표 2> 단계별 교수-학습활동계획

단계	하위 단계	주제활동	학습계획
요구분석		문제분석	· 기존 시스템의 문제점 발견 · 설계 아이템 선정하기
		자료분석	· 다양한 기존 시스템 분석 · 실제 사용자 인터뷰 및 정리
		작업분석	· 상황 설정 및 시나리오 작성
		개발목표정의	· 요구사항 명세서 작성
		전문가 검증	· 검증사항 반영
설계	도입	개념모델설계	· 개념모델의 정의 및 예시
		시나리오 생성	· 개발목표의 1차 스케치
		사용자 평가	· 사용자대상 시나리오 검증
		검증	· 검증사항 반영
	전개	외형적 시나리오	· 대략적인 시나리오 정의
		프로토타이핑	· 카드기반 프로토타이핑 구현
		사용자 평가	· 사용자 시나리오 검증
		검증	· 검증사항 반영
	발전	구체적 시나리오	· 구체적인 시나리오 구현
		프로토타이핑	· 페이퍼 프로토타이핑 구현
		사용자 평가	· 사용자 시나리오 검증
		검증	· 검증사항 반영
평가	사용자 평가	· 사용자에 사용성 평가 의뢰	
	전문가 평가	· 전문가에 사용성 평가 의뢰	
	설계 완료	· 개발자 자기 평가 및 보완	

나. 설계 단계

설계 단계에서는 도입, 전개, 발전의 세 단계를 거치며 개괄적이고 추상적인 모델에서 구체적인 모델 개발로 옮겨 간다.



(그림 3) Mock-up 및 Paper-Prototyping 제작

다. 평가 단계

최종평가에서는 사용자 평가와 전문가 검증의 단계를 두어 최초 정의된 개발목표와의 적합도를 검증하고 사용성을 테스트한다.

4. 적용 결과

4.1 적용대상



(그림 4) 최종 사용자 평가 및 전문가 평가

5. 결과 분석

본 연구에서 제안한 프레임워크를 활용한 교육방안을 적용하여 다기능 리모콘 개발하기, 개인 홈페이지 만들기, 웹 게시판 시스템 설계하기의 3가지 주제 프로젝트를 수행하기 전과 후 학생들의 개념 모형의 변화를 알아보고자 심성모형 이론을 적용한 후 결과를 분석하였다.

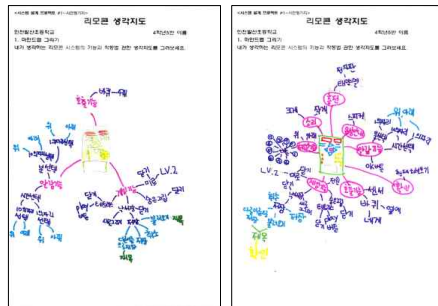
심성모형의 결과를 측정하기 방안은 매우 다양하다. 본 연구에서는 학생들이 제안 프레임워크를 수행한 전후의 심성모형을 측정하기 위하여 사용자에게 시스템을 일정기간 사용하게 한 후 해당 시스템의 구조나 기능을 사용자가 이해하는 대로 표현하도록 유도하도록 한다. 이 방법의 장점은 사용자가 큰 어려움 없이 관찰자의 요구에 따를 수 있고, 또 사용자의 심성모형에 대한 자세한 서술적인 정보를 얻을 수 있다는 것이다. 하지만 이 방법의 단점은 막상 사용자가 작성한 그림을 보고 사용자가 해당 시스템의 기능이나 구조를 얼마나 제대로 이해했는지를 파악하기 힘들다는 것이다. 이러한 문제를 해결하기 위해서 제안 연구에서는 마인드 맵을 이용하여 심성모형을 분석하는 정성적인 방안과 별도의 설문을 분석하여 결과를 분석하는 정성적 방안을 복합적으로 수행하였다.

5.1 심성모형을 이용한 논리적 사고방식의 측정

심성모형을 이용하여 구조모형을 측정하는 것은 다분히 주관적인 것이기 때문에 최소한의 객관성을 확보하기 위하여 마인드맵 기법을 활용하였다. 마인드맵 기법을 적용하여 분석한 내용은 다음과 같다.

가. 사고의 폭이 넓어졌다.

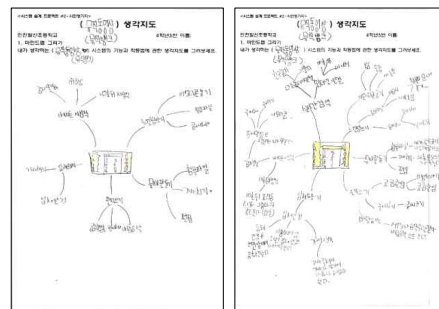
소프트웨어 개발교육을 받은 학생들의 실험 전·후를 비교해 보면 다수의 아동에게서 중심 이미지에서 처음 뻗어 나오는 주가지의 개수 증가를 볼 수 있었다. 주가지의 개념은 시스템이 가지고 있는 기능을 의미하는 것으로 개발자의 심성모형에 있어서는 폭넓고 다양한 사고가 가능해졌음을 의미한다.



(그림 3) 사전·사후 평가의 예

나. 사고의 깊이가 깊어졌다.

실험후 학생들의 마인드 맵에서 주가지에서 뻗어 나오는 부가지의 차수 증가를 발견할 수 있었다. 부가지의 단계가 깊어졌다는 것은 기능의 구현을 위해 이루어지는 중간과정을 개발자의 개념 모형 속에 형성하였다는 것을 의미하는 것으로 사고의 깊이가 깊어졌다고 할 수 있다.

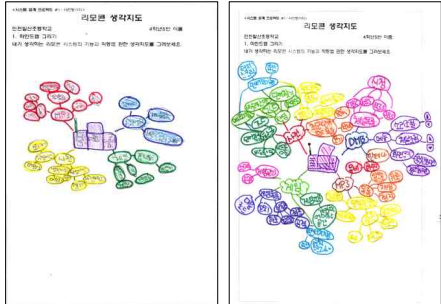


(그림 4) 사전·사후 평가의 예

다. 구조화 된 사고를 하게 되었다.

개념모형을 형상화하는 마인드맵을 보면 상위

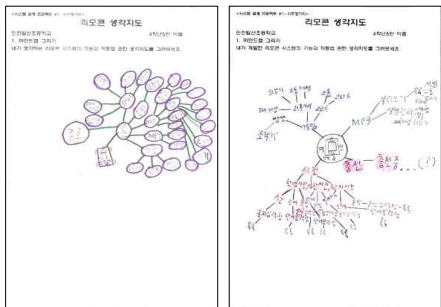
단계에서 하위 단계로 내려가면서 생각의 폭이 피라미드형으로 체계화 됨을 발견 할 수 있었다. 이는 개발자의 사고가 구조화 되었음을 의미한다.



(그림 5) 사전·사후 평가의 예

라. 균형적인 사고를 하게 되었다.

머릿속의 개념모형을 표현한 마인드 맵을 보면 교육 전에는 학습지에 치우치게 그리는 등 공간 활용이 균형적이지 못하였다. 그러나 개발활동 후에는 개념모형이 이미 형성된 내용을 표현하는 것이기 때문에 주어진 공간을 균형적으로 활용하여 표현할 수 있었다.



(그림 6) 사전·사후 평가의 예

5.2 심성모형을 이용한 가치태도 측면의 분석

가치태도 측면의 분석은 본 연구에서 제안한 프레임워크의 활용에 및 교육에 대한 학습자들의 반응을 설문을 통하여 분석하였다.

설문문항은 <표 3>의 내용과 같이 기능적 가치, 유희적 가치, 사회적 가치, 개인적 가치의 4 부류[5]

로 이루어져 있으며 각 부류마다 3 가지 가치항목을 질의하였으며 Likert 척도에 의하여 5단계의 답변을 얻어 한 영역 당 15점 만점으로 점수를 주도록 하였다. 이를 SPSS 12.0으로 대응표본 t검증을 통해 결과를 분석하였으며 유의수준은 p<.05로 설정하였다.

<표 3> 가치 태도 측정 문항

분류	문 항
기능적 측면	·초등학교에서도 소프트웨어 개발교육은 필요하다. ·소프트웨어 개발교육을 통해 많은 것을 배웠다. ·소프트웨어 개발은 실제 필요한 시스템을 개발한다.
유희적 측면	·소프트웨어 개발 교육은 흥미롭다. ·소프트웨어 개발 교육을 자주 해보고 싶다 ·소프트웨어 개발 교육을 할 때 기분이 좋다.
사회적 측면	·함께 소프트웨어 개발을 하면 다른 사람들과 더 친근해 질 것이다. ·소프트웨어 개발을 하면 다른 사람들에게 도움이 될 것이다. ·소프트웨어를 개발하면서 다른 사람들과 공감대를 느낀다.
개인적 측면	·나만의 시스템을 통해 나를 남과 차별화 시킬 수 있다. ·윈도우나 핸드폰처럼 실제 소프트웨어 개발을 해보고 싶다. ·소셜 계발을 위하여 소프트웨어 개발을 계속 하고 싶다

가. 기능적 가치 분석

기능적 가치 영역에 대한 분석결과 15점 만점에 평균값이 0.89점 올랐으며 이는 95% 신뢰구간 내에 위치하여 유의미한 차이가 있는 것으로 나타났다. 또한 자신들이 실제 생활에 필요한 것을 개발하였다고 응답해 응용 프로그램 개발교육과 실생활과의 높은 연관관계를 나타내고 있다.

<표 4> 기능적 가치 사전-사후검사

	평균	표준편차	t
사전	9.33	1.96	-5.451
사후	10.22	1.93	

p<.05

나. 유희적 가치 분석

유희적 가치 영역에 대한 분석결과 평균값이 1.07점 올랐으며 이는 95% 신뢰구간 내에 위치하여 유의미한 차이가 있는 것으로 나타났다. 개발 교육 초기에는 낯설어 하는 학생들도 있었으나 스스로 프로그램을 개발한다는데 흥미를 느끼게 되었다.

<표 5> 유희적 가치 사전-사후검사

	평균	표준편차	t
사전	8.93	2.40	-4.508
사후	10.00	1.94	

p<.05

다. 사회적 가치 분석 문항

사회적 가치 영역에 대한 분석결과 평균값이 1.00점 올랐으며 이는 95% 신뢰구간 내에 위치하여 유의미한 차이가 있는 것으로 나타났다. 응용 프로그램 개발교육을 통해 교육 전 보다 훨씬 높은 사회적 가치 문항에 대한 답변을 얻을 수 있었다. 이는 공동개발활동을 통해 협동심을 개발할 수 있으며 프로그램 개발을 통해 다른 사람에게 이익을 줄 수 있을 것이라는 생각을 가지게 되었다. 이는 함께 살아가는 사회에서 정보통신기술의 중요성을 느끼는 계기가 되었다.

<표 6> 사회적 가치 사전-사후검사

	평균	표준편차	t
사전	8.44	1.88	-4.088
사후	9.44	2.03	

p<.05

라. 개인적 가치 분석 문항

이번 응용 프로그램개발 교육을 통해 학생들은 자신의 개성과 소질을 발견하고 지속적으로 소프트웨어 개발교육에 관심을 가져 실제로 유용한 시스템을 개발해 보고 싶다는 응답을 한 학생들이 많았다.

사회적 가치 영역에 대한 분석결과 평균값이 0.96점 올랐으며 이는 95% 신뢰구간 내에 위치하여 유의미한 차이가 있는 것으로 나타났다. 응용 프로그램 개발교육에 흥미와 소질을 보이는 학생들은 체계적인 교육과정 내에서 지속적인 소프트웨어 개발교육 받기를 희망하였다.

<표 7> 개인적 가치 사전-사후검사

	평균	표준편차	t
사전	8.52	2.41	-5.107
사후	9.48	2.17	

p<.05

5. 결론

본 연구에서는 그 동안 초등학교 정보통신기술교육에서 체계적으로 다루기 어려웠던 응용 프로그램 개발교육을 위한 교육 방안을 모색하였다. 제안한 프레임워크로 실제로 학생들에게 적용하는 실험을 실시하였고 학생들에게 형성된 개념모형 측정을 위하여 심성모형평가를 실시하여 다음과 같은 결론을 얻었다.

첫째, 응용 프로그램 개발교육 후 학생들의 논리적 사고력이 신장되었다.

응용 프로그램 개발교육 전·후 학생들의 개념모형 평가에서 학습전보다 훨씬 폭 넓고 체계적이며 구조적인 사고가 가능하다는 것을 알 수 있었다.

둘째, 응용 프로그램 개발교육 학습 후 학생들은 응용 프로그램 개발 교육의 필요성을 느끼고 흥미를 갖게 되었다.

응용 프로그램 개발 교육을 진행하면서 처음엔 학생들이 생소한 학습주제에 낯설어 했으나 곧 자신만의 프로그램을 개발한다는 것을 좋아하게 되었다.

셋째, 응용 프로그램 개발교육을 통해 협동심 고취 및 소질 개발, 진로교육이 가능하였다.

응용 프로그램의 개발의 과정을 통해 협동심을 기르고 타인과 협상하는 능력을 키우게 되었다. 또한 그동안 알지 못했던 자신의 소질과 능력을 발견하는 계기가 되었으며, 더욱 정진하여 장래 진로에도 영향을 끼치는 교육활동이 될 수 있었다.

넷째, 본 연구에서 제시한 응용 프로그램 개발 프레임워크는 초등학교에서의 응용 프로그램 개발교육에 대한 새로운 접근방법이 될 수 있다.

어려운 프로그래밍을 대신하여 페이퍼 프로토타이핑 기법을 활용한 본 프레임워크는 그동안 코딩의 어려움으로 인해 실시하기 어려웠던 초등학교에서의 응용 프로그램 개발 교육의 새로운 방법이 될 수 있을 것이다.

참고문헌

- [1] 조성환 외 (2009), CPS에 기반한 스크래치 EPL이 문제해결력과 프로그래밍 태도에 미치는 효과, 한국정보교육학회 학술발표논문집, 12-1, 77-88.
- [2] 김철동 외 (2007), QBasic을 이용한 초등학교 프로그래밍 교육과정 연구, 한국정보과학회 학술발표논문집, 34-2, 151-155.
- [3] 교육인적자원부 (2005), 초·중등학교 정보통신 기술교육 운영지침.
- [4] 김한샘 외 (2003), 정신모형과 감성 요소를 이용한 소프트웨어 사용성 평가 모델 개발, 정보과학회논문지, 30-1, 117-128.
- [5] 김진우 (2005), Human Computer Interaction, 서울: 안그래픽스.
- [6] 김희철 (2006), 인간과 컴퓨터의 상호작용, 서울: 사이텍미디어.
- [7] Carolyn Snyder (2003), Paper Prototyping, New York: Morgan kaufman.
- [8] Preece, J., H. Rogers, and H. Sharp.(2002), Interaction design: Beyond human-computer interaction. New York: Wiley.
- [9] Klee, M. (1986), Five paper prototyping tips, Available from User Interface Engineering, Snyder.
- [10] Grady, H. M.(2000), Web site design: A case study in usability testing using paper prototypes, IEEE Professional Communication Society Conference, 39-45.
- [11] Muller, M. J., and R. Carr.(1996), Using the CARD and PICTIVE participatory design methods for collaborative analysis, In Field methods casebook for software design, New York: Wiley.
- [12] Uceta, F. A, M. A. Dixon, and M. L(1998), Resnick, Adding interactivity to paper prototypes, In Proceedings of the Human Factors Society, 506-511.

저자소개

최진용



1999년 경인교육대학교 실과교육과 (학사)
2009년 경인교육대학교 컴퓨터교육과 (석사)
2007년~현재 인천발산초등학교 교사
관심분야 : HCI, Digital Textbook, Scratch Programming

e-mail : flark@naver.com

손원성



1998년 동국대학교 컴퓨터공학(학사)
2000년 동국대학교 컴퓨터공학(석사)
2004년 연세대학교 컴퓨터과학(박사)
2004년~2006년 Carnegie Mellon University, Post Doc.
2006년~현재 경인교육대학교 컴퓨터교육과 조교수

관심분야 : HCI, 멀티미디어 문서처리, 컴퓨터교육
e-mail : sohnws@gin.ac.kr