

워터폴 모델을 적용한 앱 인벤터 프로그래밍 교재개발 연구

설문규* · 손창익**

진주교육대학교* · 감천초등학교**

요약

본 연구는 프로그래밍 교육에서 앱 인벤터 교육용 프로그램의 활용 가능성을 고찰하고 기존 프로그래밍 교육의 한계를 극복하기 위한 초등학생을 위한 프로그래밍 교재를 개발하였다. 스마트기기 프로그램을 통하여 학습 동기를 강화하였고 STEAM적 요소를 가미하고 앱 인벤터 프로그래밍 학습 내용을 학생들의 실생활적인 상황 제시와 소프트웨어 개발과정을 따라가면서 보다 논리적이고 체계적인 학습이 이루어지도록 개발하였다.

프로그래밍 과정에서 소프트웨어 개발이론 워터폴 모델을 도입하여 학생들의 소프트웨어 개발자의 사고과정을 따라가게 함으로써 기존의 교재가 가지는 단순 프로그램 언어와 관련된 지식의 습득에서 벗어나 학생들의 창의성과 문제해결력, 협동 사고력 등 고차원적인 사고력을 신장시키고자 하였다.

키워드 : 교재개발, STEAM, 소프트웨어 개발이론, 워터폴모델, 문제중심학습

A Study on Development of Teaching Materials for App Inventor Programming Using the Waterfall Model

Moon-gu Seol* · Chang-ik Son**

Chinju National University of Education* · Gamcheon Elementary School**

ABSTRACT

The aims of this paper were to review the usable possibility of the educational App Inventor Program in the field of programming education and to develop programming teaching materials that can overcome limitations of the established programming instruction. The study showed that the learners' motivations were strengthened through smart device programs. Containing the elements of STEAM, the teaching materials were developed for the logical and systematic learning that deals with elementary students' real-life situations, and that helps children follow the procedures of software development.

By introducing the Waterfall Model to the process of programming, students are able to follow the software developers' thinking process. In addition, beyond the simplistic programming language and simply acquiring related knowledge, the App Inventor programming was designed to enhance students' higher-order thinking skills such as creativity, problem solving ability, collaborative thinking, and so forth.

Key words : teaching material development, STEAM, software development methodology, Waterfall Model, problem-based learning

이 논문은 2012학년도 진주교육대학교 학술연구과제 연구비 지원을 받아 작성된 것임.

교신저자 : 설문규(진주교육대학교)

논문투고 : 2013-09-14

논문심사 : 2013-09-17

심사완료 : 2013-12-13

1. 서론

21세기 초반 현재 지식정보사회에서는 창의성을 갖춘 고급 인력의 양성이 강조되고 있으며 창의성, 비판적 사고력, 문제해결력 다양한 리터러시, 협력, 커뮤니케이션 능력 등은 이 시대를 이끌어가는 대표적인 개념이자 패러다임이 되었다.[19]

따라서 세계화, 테크놀로지의 발달, 웹2.0시대를 지나 3.0시대의 도래 등 다양한 시대적 변화에 따라 학습자는 전통적인 교육내용·방법에서 벗어나 창의성 신장, 창의적 인재 교육 등 다양한 능력이 요구되었고 이는 하나의 시대적인 화두로서 다양한 방법을 모색하게 되었다.

그 중 특히 논리적사고력과 문제해결력을 향상시키는데 프로그래밍 교육이 효과적이라는 연구 결과는 주목할 만하다.

프로그래밍 교육은 문제를 해결하기 위해 체계적인 접근에 의한 규칙을 찾아내고, 논리적 흐름을 고려한 프로그래밍과 오류를 수정하는 일련의 과정에서 논리적 사고력을 갖게 되고 오류수정 및 알고리즘의 절차를 터득하고 적용해 봄으로써 창의적인 사고과정을 경험하고 이를 통한 문제해결력을 향상시킬 수 있다.[4]

하지만 이러한 긍정적인 측면에도 불구하고 기존의 프로그래밍 교육은 기본 문법과 구조를 이해하고 프로그램 사용법을 익히는데 너무 많은 노력을 요구하여 학습자들로 하여금 프로그래밍 학습을 기피하거나 포기하도록 만들었다.

또한 프로그래밍은 본질적으로 고차원적 사고 및 문제해결전략을 수반하기 때문에 대부분의 일반 학습자들은 프로그래밍 학습을 추상적이고 난해한 것으로 인식하고 있는데, 특히 정보처리과정에 익숙하지 않은 초등학교 학습자에게 프로그래밍의 개념, 원리, 기능습득과 함께 논리적 문제해결능력을 요구하는 것은 학습자에게 심리적 부담을 줄 수 있다.

따라서 소프트웨어개발이론 워터폴을 바탕으로 하는 체계적인 학습 경험을 구체적인 단계와 반복적인 경험을 해나가 프로그래밍에 대한 어려움과 딱딱함을 줄이고 단순히 알고리즘의 암기나 프로그래밍 언어를 익히는데 한정하지 않고 프로그래밍 언어에 쉽게 접

근·이용할 수 있도록 기존의 프로그래밍 언어보다는 그래픽 프로그래밍 환경과 쉬운 사용자 인터페이스, 블록의 조합을 통한 코딩 등 보다 배우기 쉽고, 직관적인 교육용 프로그래밍 언어(EPL)인 ‘앱 인벤터’를 통해 이러한 문제점에 해결하고자 하였다.

프로그래밍은 형식적이고 추상적 사고 활동을 위해 보다 구체적인 환경을 제공하는데 즉, 컴퓨터에게 무엇을 할지를 명령, 확인 및 교정하는 활동을 통하여 학습자는 자신의 사고에 대해 형식적으로 사고할 수 있게 된다. 따라서 소프트웨어 개발 상황에서 필요한 요구사항을 직접 분석하고 체계화함으로써 학습이 유발된다.[2]

그러므로 정보교육에서 사용되는 프로그래밍 언어는 산업현장의 프로그래머 양성이 목적인 범용 언어보다는 문제해결력을 기르는데 초점이 맞춰진 교육용 프로그래밍 언어 사용이 효과적이다.[4,15]

특히 스마트기기의 보급으로 학생들이 언제, 어디서든 스마트폰을 활용하고 있으며 학생들의 생활에서 가장 중요한 영역을 차지하고 있으므로 스마트폰은 기반으로 하는 앱 프로그래밍은 단순 컴퓨터용 프로그래밍에 비하여 초등학교 학생들의 흥미도와 집중력을 높일 수 있을 것으로 기대된다.

따라서 본 연구는 프로그래밍에 대한 학습동기를 증진시키고 문제해결력 신장을 위하여 STEAM과 문제중심학습의 특성을 반영하고 소프트웨어 개발이론에 근거한 프로그래밍 교재를 개발하여 보다 논리적이고 객관적인 사고력을 신장하고 창의적인 학생으로 자라날 수 있도록 도움을 주고자 한다.

2. 이론적 배경

2.1 프로그래밍 학습

프로그래밍 교육은 학습자의 고등인지 능력을 향상시키고 컴퓨터의 원리 이해와 소프트웨어의 활용 및 개발에 대한 기초 지식을 쌓을 수 있기 때문에 현재 운용 중인 정보통신기술교육 운영지침에 포함되어 있다. 하지만 초등학교 학생에게 적합한 교육용 프로그래밍 언어의 제공, 프로그래밍 학습 요소 구조화 및 교수·학습 전략 등의 미비로 인하여 최근에 초등학교

프로그래밍 교육에 대한 연구가 활발하게 진행되고 있다.

프로그래밍 과정은 학습자 스스로 창의적 사고력, 확산적 사고력, 논리적 사고력 등 다양한 사고력 및 문제해결력을 향상시킬 수 있게 한다. 그리고 컴퓨터의 내부 동작을 이해하는데 많은 도움을 주며, 궁극적으로 컴퓨터를 보다 깊이 이해하고 활용할 수 있도록 해 줄 수 있다. 이런 과정을 통해 각종 응용프로그램을 더욱 잘 알고 활용할 수 있는 기초를 마련해 준다. 또한 컴퓨터 프로그래밍에 처리 능력, 유추적 추론 기능, 조건적 추론 기능, 절차적 사고, 일시적 추론 기능뿐 아니라 일반적인 계산은 물론 다양한 측면에서의 수학적 능력이 향상될 수 있다. 또한 이러한 처리 능력을 통해 집중적 기억 능력과 정보처리 능력이 요구되며, 유추적 추론 기능은 프로그래밍과 관련된 지식이나 기능을 다른 분야나 상황에 적용시킬 수 있는 능력으로 창의성과 관련된 부분이 많다.[5]

교육용 프로그래밍 언어(Educational Programming Language : EPL)는 프로그래밍 교육에서 학습자들이 가지는 다양한 인지적 부담을 줄여주고, 동기를 부여할 수 있도록 교육적인 목적을 가지고 개발된 프로그래밍 언어를 말한다.[14,17]

교육용 프로그래밍 언어는 그래픽이 활용되는 범위와 작성 방법에 따라 Logo, Karel the Robot, KPL, Dolittle등 과 같이 텍스트 기반으로 구성된 언어인 TPL(Typing-based Programming Language)과 이와는 다르게 AgentSheets, Alice, Labviews, Scratch 등과 같이 프로그래밍의 코딩 과정 자체가 그래픽을 지원하는 VPL(Visual Programming Language)로 구분할 수 있다.[16]

본 연구에서는 가장 최근에 개발된 교육용 프로그래밍 언어중 하나로 논리블록을 조합하여 간단한 안드로이드 어플을 제작할 수 있는 앱 인벤터(App Inventor)를 사용하고자 한다.

2.2 앱 인벤터(App Inventor)

앱 인벤터는 구글이 프로그래밍 지식이 없는 이들도 원하는 어플을 쉽게 만들 수 있도록 WEB브라우

저 상에서 퍼즐모양의 논리블록을 조합하여 간단한 안드로이드용 어플을 제작할 수 있는 무료 웹서비스이다.

그러나 2010년 안드로이드용 앱 인벤터를 공개했던 구글실험실(Google Labs)이 문을 닫으면서 이 프로젝트는 MIT로 넘어가 MIT 모바일 학습센터(MIT Center for Mobile Learning)에서 MIT 앱 인벤터로 공개되었다.

앱 인벤터는 웹페이지와 자바 인터페이스를 이용하여 안드로이드 애플리케이션을 디자인하고 프로그래밍을 하므로 별도의 고성능 하드웨어가 필요하지 않으며 운영체제의 영향을 받지 않고 아무 웹브라우저에서도 사용이 가능하여 교육용으로 탁월한 환경 적응성을 보인다.

앱 인벤터는 디자인 뷰(Design View)와 블록 에디터(Blocks Editor)라는 두개의 작업영역을 가지는 데 디자인 뷰에서는 제작하고자 하는 애플리케이션의 사용자 인터페이스(UI)부분을 설계하고 특정한 기능을 가진 컴포넌트를 추가한다.

블록 에디터는 일반적인 프로그램에서 코딩을 하는 부분으로 디자인 뷰에서 추가한 컴포넌트들을 연결하여 각각의 기능을 연결하고 알고리즘을 적용하여 실제적인 결과를 만들어 내는 부분이다.

앱 인벤터는 MIT에서 이전에 공개한 스크래치(Scratch)의 기존 구조와 문법을 그대로 가져왔기 때문에 기존의 스크래치를 배운 학습자는 쉽게 앱 인벤터로 옮겨올 수 있는 장점을 가지고 있다.

2.3 STEAM과 문제중심학습(PBL)

융합인재교육(STEAM)이란 과학기술에 대한 학생들의 흥미와 이해를 높이고 과학기술 기반의 융합적 사고(STEAM Literacy)와 문제해결력을 배양하는 교육을 말한다.

기존의 교육이 각 교과목 별로 정립된 지식과 개념, 이론을 교과서 중심으로 전달하는 것에 주력하였다면, 융합인재교육은 학생들과의 관련성(relevance)을 강조하여 어디에 쓰이는지 왜 배우는지 이해하고 실생활에 활용되도록 한다. 체험하고 스스로 설계하고 탐구·실험하는 과정을 강조하고, 실생활의 문제

해결력을 배양한다.

융합인재교육이 추구하는 교육의 이론적 근거를 정립하기 위하여 교육과학기술부와 한국과학창의재단은 ‘융합인재교육 실행방향 정립을 위한 기초연구’를 추진하여 그 결과로 융합인재교육 학습 준거(틀)을 도출하였다. 학습 준거(틀)은 융합인재교육의 현장 적용을 위한 가이드라인이자 프로그램의 판단 기준으로 활용될 수 있다.

융합인재교육 준거(틀)은 상황제시, 창의적 설계, 감성적 체험으로 이 구조는 문제중심학습(PBL)모형을 적용하기 쉽게 하여 많은 현장수업에서 문제중심 학습이 적용되고 있다.[1,17]

문제중심학습 모형(PBL)은 현실 속에서 복잡하게 얽혀있는 비 구조화된(ill-structured)문제를 학습자가 스스로 의미 있는 해결방법을 찾아내게 함으로써, 교과 지식과 기술뿐만 아니라 문제해결전략과 자기주도 학습을 동시에 익히도록 하는 교수-학습 형태로 본 연구의 중심이 되는 개념을 제공한다. [13]

이에 문제해결학습의 기본적인 문제해결의 과정과 문제의 제시와 관련하여 문제를 만드는데 필요한 문제의 구성요소에 많은 시사점을 얻고 이를 실제적인 교재 제작에 적용하였다.

문제중심학습에서 문제해결의 과정은 Barrows와 Myers(1993), IMSA 모형, Delisle, Fogarty 등 여러 학자들에 의해 여러 모형으로 제시되고 있지만 그 내용은 크게 다르지 않다.[8,13]

문제해결학습에서 제시되는 기본적인 PBL문제의 구성 요소는 제목, 문제 상황에 대한 소개, 문제에 대한 간략한 소개, 개방형 질문 등으로 구성되며 좋은 PBL 문제를 작성하는 것은 세 가지 특징이 요구된다.[18]

가. 우리가 학생들이 배우기를 바라는 내용과 개념 (content and concepts)

나. 문제에 집중할 수 있게 하는 이야기(story)

다. 열려 있고(open ended) 도전할 수 있는 문제 구조(structure)

이러한 문제중심학습에 바탕을 두고 STEAM 교육에 기반을 둔 자료를 개발하기 위해서는 기존의 기초 과학의 틀에서 벗어나 다른 학문분야와 융합적인 관계를 가지며 새로운 과학·기술·공학 지식을 바탕으

로 구체성과 정확성을 가진 학습자 지식수준에 맞춘 자료를 개발해야 한다.

이러한 교재 개발 특히 STEAM 교육 콘텐츠를 개발하기 위한 필수 요소와 특징에 대하여 최정훈(2011)은 다음과 같이 이야기하고 있다.[17]

가. 과학·기술·공학 중심으로 인문학과 융합적인 사고로 미래를 체계적으로 예측할 수 있는 활동을 만든다는 자세가 필요하다.

나. 기초 과학 원리에서 첨단 과학·기술·공학까지 스토리텔링으로 자연스러운 흐름으로 재미있고 이해가 쉬우면서 창의적인 생각을 할 수 있는 표현 기술이 필요하다.

다. 기초과학원리를 응용한 첨단 과학·기술·공학의 다양성을 학년별, 학생수준별 높이에 제시해 주어야 한다. 이것을 통해 과학·기술·공학에 대한 발산적 사고와 창의성을 유도하는 기틀이 되어야 한다.

라. STEAM 개념에 기반을 둔 다양한 창의적인 기법, 학습도구, 체험활동개발이 중요하다.

마. STEAM에 대한 학제 간 협동학습을 적용하여 과학·기술·공학에 대한 탐구 능력과 윤리 및 사회성, 협동성, 리더십, 서로를 배려하고 소통하는 능력을 키움으로써 창의성뿐 아닌 인성교육을 행하여 여러 각 분야에 사회 일원으로 육성하도록 해야 한다.

바. STEAM 교육은 국제적인 감각과 인류의 문화, 역사, 정치, 경제, 환경 등 인류가 가진 문제를 높은 윤리 의식으로 과학·기술·공학지식을 바탕으로 해결할 수 있는 인재 양성을 할 수 있게 해야 한다.

사. STEAM 교육은 통합적이고 전체를 볼 수 있는 능력을 배양하는 교육이 되어야 한다.

아. 가장 최신의 급변하는 융합기술변화에 빠르게 대처하는 적시교육(Just in time learning)이 되는 STEAM 교육이어야 한다.

자. 학생들이 오늘날의 과학·기술·공학 지식을 다루고 활용하는 주체로서 알아야 할 기본 소양을 강조하는 STEAM 교육이 되어야 한다.

2.4 워터폴(Waterfall) 모델

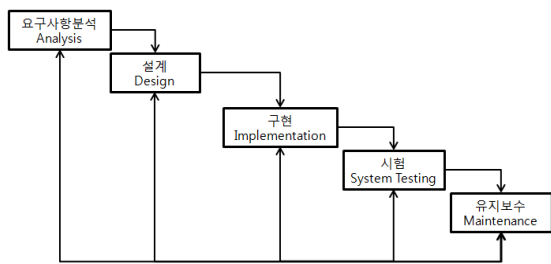
워터폴 모델은 순차적인 소프트웨어 개발 프로세스로 개발의 흐름이 마치 워터폴처럼 지속적으로 아

래로 향하는 것처럼 개념 정립에서 구현까지 하향식 접근 방법을 사용하여 높은 추상화 단계에서 낮은 추상화 단계로 옮겨가는 모델이다.[10,15]

위터폴 모델은 소프트웨어 개발에 구조화된 접근 방법을 제공하고, 쉽고 이해할 수 있고 설명이 가능한 각각의 구분된 단계를 순차적으로 진행한다.

만들기 전에 생각하고, 모든 것을 문서화하고, 계획을 준수하며, 가능한 모든 것을 조직화된 상태를 유지하는 논리성과 구조는 프로그램 학습에 있어 중요한 시사점을 준다.

위터폴 모델의 흐름은 소프트웨어 요구사항 분석 단계에서 시작하여, 설계, 구현, 시험, 통합단계 등을 거쳐, 소프트웨어 유지보수 단계까지 이른다.



(그림 1) 위터폴 모델 단계[14]

2.5 프로그래밍 교육에 대한 선행연구

강혜진(2004)의 연구에 의하면 프로그래밍 기초 교육은 주어진 문제를 해결하는 과정과 절차에 중점을 두고 학생 스스로 조작하도록 함으로써, 거듭되는 동화와 조절작용을 통해 논리적사고력이 향상되는데 효과가 있다는 결론을 얻었다. 따라서 초등학교에서 컴퓨터 교육이 효과적으로 이루어지기 위한 하나의 방법으로 프로그래밍 교육이 가장 적합한 시기에 아동의 지능 수준에 적합한 내용과 방법으로 컴퓨터 교육이 학교별로 정착되어야 한다고 제안 했다.[2]

백영근과 우인상(1994)은 Logo를 활용한 연구에서 학습자가 중심이 되어 능동적이고 적극적으로 문제를 해결하고 교사는 문제해결과정에 따라 그들의 사고과정을 안내해주는 안내된 발견식 수업이 문제를 해결할 때 문제해결력에 효과적임을 발견하였다. 이러한

결과는 다른 연구결과에서도 확인되었다.[9,13]

즉 교사가 시범을 보이고 학생들이 수동적인 위치에서 교사에 의해 제기된 문제를 지시에 따라 해결하는 지식적 수업보다 학습자 스스로 그들의 학습활동을 통제하고 능동적으로 문제를 해결할 수 있는 환경을 조성해 주는 것이 필요하다는 것을 시사한다.

프로그래밍 교육과 관련된 대다수의 연구는 프로그래밍 교육의 목적이 프로그래밍 언어와 관련한 지식습득하는데 두는 것이 아니라 학습자의 창의성과 문제해결력, 협동 사고력 등 학습자의 고차원적인 사고력을 신장시키는데 두고 있음을 시사하고 있다.[4]

본 연구에서 사용된 앱 인벤터에 대한 기존의 연구가 미미한 관계로 같은 문법적 구조와 형태를 가지고 있는 스크래치의 선행연구를 종합하여 보면 프로그래밍 교육에서의 활용적 가치를 다음과 제시할 수 있다.[8]

가. 프로그래밍 교육에서 부가될 수 있는 학습의 적 인지 부하를 감소시킬 수 있다.

기존의 프로그래밍 교육의 가장 큰 문제점인 도구 사용법 자체의 어려움으로 인해 도구의 사용법 학습에 더 많은 시간과 노력을 기하는 문제점을 극복하고 프로그래밍을 통한 추상적 개념의 습득 및 논리적 사고력의 향상에 집중하고 특히 블록들의 조합을 통한 구문 오류를 사전에 예방함으로써 문법적인 오류디버깅에 대한 노력을 최소화할 수 있다.

나. 즉각적이고 구체적인 피드백을 제공함으로써 학습자의 흥미와 내적 동기를 유발할 수 있다.

이은경과 이영준(2008)은 중학생 대상의 프로그래밍 교육에서 학습에 대한 내적 동기를 설명하는 몰입 수준과 프로그래밍 수행 능력 측면에서 유의한 영향을 주었음을 확인하였다.

다. 협력적 문제해결의 기회를 제공하고 상호작용을 촉진할 수 있다.

3. 교재 내용 선정과 구성

3.1 교재 개발방향

본 연구는 초등학교 5-6학년을 대상으로 창의적 체험시간이나 방과 후 활동, 동아리 활동에 활용할

수 있는 앱 인벤터 언어를 활용한 스마트폰용 애플리케이션 프로그래밍 교재를 개발·연구하였다.

3.1.1. 소프트웨어 개발 이론에 근거한 단계 설정

단순한 프로그램 명령어 학습에서 실생활에서의 필요성을 분석하여 소프트웨어 개발 이론의 단계에 따라 작은 단위의 프로그램을 개발하는 과정에서 요구사항 분석을 통한 논리성과 설계·구현단계에서의 창의성을 높일 수 있다.

3.1.2. 실천 중심의 교재

기존의 명령어 및 논리 구조 등 이론 중심에서 문제상황 해결을 위한 소프트웨어 개발 과정을 통한 문제 해결 방법을 찾아가는 실천중심이다.

3.1.3. 상황중심의 내용 전개

기존의 많은 수의 프로그래밍 학습을 위한 교재는 단순한 프로그램 명령어를 익히기 위한 관련성 없는 정보의 연속적인 나열로 이루어져있고 학생들과 실제적인 관련성이 없는 단편적이고 분절적인 구조화되지 않는 정보의 나열로 학습의 목표인 창의성 신장과 관련이 없는 단순한 프로그램 소양교육에 그치므로 프로그램 학습을 통하여 창의성, 논리성을 신장시키기 위해서 학습내용에 학생들의 실생활과 관련된 상황 즉, 이야기(내러티브)로 구성된다.[8,14,18]

3.1.4. 학생이 학습의 주인공

단순히 예제를 따라가는 학습이 아니라 문제중심 학습 및 소프트웨어 개발이론을 적용하여 실제적으로 학생이 소프트웨어의 요구사항을 분석하여 필요한 요소를 찾아 설계·구현하는 학습이 되도록 하였다.[8]

3.2 전체 교재 내용 구성 체제

<표 1> 교재 내용 구성체제

구성체제	학습주제	내용
앱 인벤터의 기본 구조와 사용법	앱 인벤터 소개	앱 인벤터의 특징과 기본 사용법을 익히는 과정
단위 구조 프로그래밍	첫번째 앱개발	안내된 상황을 해결하기 위한 각 기능에 대한 구체적인 사용법을 익히고 다양한 예제를 통해 자기 스스로 생각하고 프로젝트를 만들기 위한 기반을 다진다.
앱인벤트 프로젝트 실습	앱 프로그래밍	프로젝트 실습으로 창의적 사고와 실천력을 키우며, 지속적인 학습을 통해 창의적 습관과 실천하는 방법을 배우게 된다.
프로젝트 통합	앱창작	기존에 배운 내용을 통합하여 스스로 기획/제작/완성의 모든 단계를 만들어 나가며, 문제해결 및 학습하는 자신만의 방법을 만들어 가는 과정

3.3 앱 인벤터 프로그래밍 학습요소 추출

앱 인벤터 언어에서 초등 학습자의 수준 및 특성, 생활 관련성 등을 고려하여 컴포넌트 중심으로 학습요소를 추출하였다.

<표 2> 앱 인벤터 학습 요소

학습요소	내용
기본 컴포넌트 / 정렬	Button, Label, TextBox, HorizontalArrangement, TableArrangement, VerticalArrangement
그래픽	Canvas, Image
미디어	Camcorder, Camera, Sound
소셜	PhoneCall, Texting, Twitter, SpeechRecognizer, TextToSpeech, Web
센서/위치	AccelerometerSensor, LocationSensor, OrientationSensor
데이터	TinyDB

3.4 교재 학습 내용 선정

학습 내용 선정에 있어 기존의 프로그래밍 언어 요소를 중심으로 선정함에서 탈피하여 문제 중심, 상황 중심으로 교육내용을 선정하였다.

<표 3> 학습주제 및 학습 내용

차시	단원	활동주제	주요학습내용	주요 학습 블록
1	앱 인벤터 소개	스마트폰 알아보기	스마트폰의 정의와 기능, 역사에 대하여 알아보기	
2		스마트폰 프로그래밍	스마트폰의 활용 및 기능 알아보기 (전화, 문자보내기, 음악, 사진및동영상찍기, 앱활용등) 스마트폰 활용 윤리교육	
3		앱 인벤터 소개	앱 인벤터의 소개 및 앱 프로그램에 대해 알아보기 -앱 인벤터 개발환경, 디자인뷰, 블록에디터의 역할과 기능 살펴보기	
4	2. 첫번째 앱개발	First App 개발	'hello world!' 스마트폰 화면에 표시하기 앱 인벤터로 프로그램하기 무작정 따라하기	Label
5		기본적인 프로그래밍 용어 익히기	이벤트, 메서드, 속성, 변수, 프로시저에 대해 알아보기	
6		인사 앱	흔들거나 버튼을 누르면 인사하는 앱만들기	Button, Label, Sound Accelerometer Sensor
7		앱을 스마트폰에 담자	만든 앱을 스마트폰에 담는 방법에 대해 알아보기	Download to This Computer Download to Connected Phone
8		수식계산	두수를 입력받아 블록에디터 활용하여 수식계산을 하도록 한다. (사칙연산앱)	Textbox, Label, Button
9	3. 앱 프로그래밍	비상 전화 앱	이미지 버튼을 누르면 학교나 선생님, 비상 전화로 전화가는 앱 만들기	PhoneCall
10		쪽지 상담 앱	버튼을 이용하여 교사에게 쪽지를 보내는 앱 만들기 *그룹 쪽지 보내기	Texting
11		찍사 앱	사진을 찍어 저장하기	Camera

12	음성노트 앱	소리를 문자로 변환하여 노트하기(음성인식)	Speech Recognizer
13	스토리텔링 앱	이야기 책을 소리내어 읽어주는 앱만들기	Text To Speech
14	Sensors 팔레트 활용하기	Sensors 팔레트를 이용하여 다양한 프로그램 개발하기	Accelerometer Sensor, Location Sensor, Orientation Sensor
15	응원 도구 앱	우리반 체육대회를 위한 응원도구 만들기 (박수소리, 색카드 등)	Sound, Button
16	메일 앱	친구들에게 메일을 보내기 mailto: 이메일 전송 클라이언트 프로그램을 실행	Activity Starter
17	관광지도 학습 앱	지도를 활용한 학교 주변 관광지도 만들기	Image, Listpicker, Activity Starter
18	그림판 앱	그림판을 만들어 간단한 그림을 만들어 보기	Canvas, Arrangement
19	GPS 앱	지구상에서 유일한 지점을 찾아 나타내기, 현재 GPS 좌표값을 나타내어 보기	Location Sensor, Activity Starter
20	위키에게 물어봐 앱	위키피디아의 여러가지 키워드를 이용한 대담프로그램 제작하기	Web
21	우리반 주소록 앱	간단한 데이터베이스를 활용한 우리반 주소록 만들기	TinyDB
22	4. 앱 창작	내가 만들고 싶은 앱 구상하여 제작하기 목적 및 활용방법 서술하기 구현하고 싶은 기능 디자인 뷰, 블록에디터 설계(화면설계 및 내부 플로우 차트) 필요 컨포넌트 찾기	배운 모든 블록사용
23		내가 만들고 싶은 앱 구상하여 제작하기 프로그래밍 및 테스트	배운 모든 블록사용
24		내가 만들고 싶은 앱 구상하여 제작하기 실행 및 테스트, 프로그램 수정 앱 배포 및 친구들에게 자신이 만든 앱 소개하기	배운 모든 블록사용

3.5 프로그래밍 교재구성 체제

프로그래밍에서 가장 중요하게 요구되는 능력은 생활환경에서 부딪히는 요구사항을 문제화하거나 클라이언트가 요구하는 소프트웨어 제안서를 분석하고

이를 구현할 수 있는 능력이다.

따라서 학습자에게 프로그래밍 언어를 능숙하게 익히는 것과 함께 문제해결능력을 습득할 필요가 있다.

문제해결학습은 학습자가 관심 있어 하는 문제를 다루어 학습자가 스스로 문제를 해결하는 것을 돕고 학습자가 다음에 또 다른 문제를 만났을 때 당황하지 않고 차근차근 문제를 이해하고 정보를 수집하여 문제를 해결 할 수 있게 한다.[13]

이에 프로그래밍 학습의 기본 구성체제는 소프트웨어 개발이론 중 워터폴 이론의 개발단계를 기본으로 융합인재교육 준거틀과 문제중심학습 과정을 적용하여 개발하였다.

<표 4> 프로그래밍 교재구성 체제

워터폴 이론	융합인재 교육 (STEAM)	문제중심 학습	프로그래밍 교재 구성 체제	
요구사항 분석	상황제시	문제에 직면하기	1	상황제시
설계	창의적 설계	구조설정하기	2	요구사항 분석
구현		문제 확인하기 문제 재확인하기	3	창의적 설계
시험		산출물 또는 수행산출하기	4	프로그래밍
유지보수	감성적 체험	수행과 문제 평가하기	5	테스팅
			6	좀 더 나아가기
			7	응용과제

3.5.1 상황제시

문제중심학습에서 가장 중요시하는 것이 문제의 제시이다. 제시되는 문제의 설계는 전체 학습의 흐름과 학습 효과를 좌우하게 되므로 매우 중요하다.

학생들의 실생활에서 부딪힐 수 있는 상황을 중심으로 이야기(내러티브) 제시한다.

문제해결의 필요성을 구체적으로 느끼고 문제해결을 위한 요구사항분석 및 문제를 해결할 수 있는 실마리를 제공할 수 있어야 한다.[1,18]

- 일상생활에서 내게 일어날 수 있는 상황
- 나에게 닥칠 수 있는 상황
- 학생의 흥미를 자극하는 것
- 이렇게 하려면 어떻게 해야할까? - 창의적 설계가 가능하도록 이야기를 만든다.

3.5.2 요구사항 분석

상황제시에 제시된 문제의 요소를 분석하여 원래의 아이디어를 명확한 문장으로 작성하여 아이디어 목록을 작성한다. 각 엘리먼트(또는 컴포넌트)가 무슨 일을 할 것인지 말로 정의하여 구체적 목표를 나열한다.[10,15]

- 아이디어를 명료하게 만들기: 아이디어를 명확한 문장으로 작성하기
- 아이디어 목록 작성하기: 앱의 구체적인 목적 나열
- 아이디어 목록 정교화 하기

3.5.3 창의적 설계

애플리케이션의 외적인 동작 및 화면 디자인을 설계한다. 디자인 목표를 달성하는데 필요한 로직과 알고리즘, 인터페이스 등을 나열하여 단위목표를 세운다. 각 세부 목표를 달성하기 위한 실제 프로그래밍 용 기본 지침을 나열한다.

- 아이디어 목록을 단위목표(primitives)목록으로 변환하기
- 새로 사용하게 되는 중요한 컴포넌트 및 블록 설명하기

3.5.4 프로그래밍(Programming)

설계된 내용을 바탕으로 실제로 디자인 뷰로 화면 디자인을 블록 에디터로 프로그램 코딩을 한다.

3.5.5 테스팅(Testing)

문제 상황과 결과를 매치시켜 문제를 해결할 수 있는 앱을 만들었는지 테스트하고 피드백한다.

3.5.6 좀 더 나아가기

간단한 수준의 단위목표들부터 여러가지 단위 목표를 추가해 좀 더 복잡한 앱을 만들도록 한다.

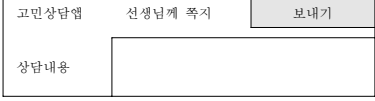
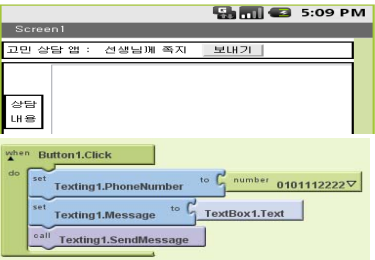

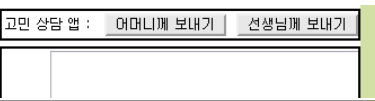
작성된 프로그램을 좀 더 확장시키거나 개선시킬 수 있는 여지가 없는지 문제 상황에서 좀 더 심화, 확대된 상황을 제시한다.

3.5.7 응용과제

학습한 컴포넌트·블록을 활용하거나 알고리즘의 응용할 수 있는 상황을 만들어 본다.

<표 5> 교재적용 수업과정 예시

쪽지 상담 앱 만들기	
단계	내용
상황제시	인수는 며칠 전 친한 친구였던 동영이와 다툼 후 쉽게 화해하지 못하고 있다. 누군가에게 조언을 얻고 싶지만 이 고민을 누구에게 이야기 할지 모르겠다. 이럴 경우 손쉽게 선생님께 고민을 상담할 수 있는 쪽지프로그램이 있다면 좋을텐데.....
요구사항 분석	<ul style="list-style-type: none"> 아이디어를 명확한 문장으로 작성 보내고자 하는 내용을 텍스트 상자에 적어 보내기 버튼을 누르면 내용이 문자 메시지로 전송된다. 아이디어 목록 작성 고민상담 앱 제목 라벨 여러 줄의 내용을 입력할 텍스트 박스 문자를 보내기 위한 버튼 상담내용 라벨과 텍스트를 한 라인에 연결하며 텍스트 박스를 꽉차게 한다.(Fill) 버튼을 누르면 텍스트 박스에 써여진 내용이 특정 번호(선생님 번호)로 문자메세지가 보내진다.
창의적 설계	<기능 설계> <ul style="list-style-type: none"> 새 컴포넌트 <ul style="list-style-type: none"> TextBox : 여러 줄의 생각을 입력 상자에 입력한다. Button : 보내기 버튼을 누르면 특정 동작이 이루어진다. Texting : 지정 번호(선생님)에 문자 메세지 보내기 HorizontalArrangement :정렬하기 새 블록 <ul style="list-style-type: none"> Click/PhoneNumber/Message/SendMessage TextBox에 상담내용을 쓴 후 보내기 버튼을 누르면 사전에 지정된 번호(선생님)로 문자가 보내짐

	<화면구성 설계> 
프로그래밍	화면 및 블록에디터 프로그래밍 
테스팅	실제 휴대폰이나 에뮬레이터로 실행한 후 미비점을 보충하고 피드백한다. 
좀 더 나아가기	버튼을 이용하여 보낼 수 있는 사람을 여러 명으로 지정하기 
응용과제	간단한 메시지 앱 구현하기 (수신인 입력 또는 목록 선택, 메시지 내용 입력, 이미지 발신버튼)

4. 적용

개발한 교재는 창원감천초등학교 초등학교 5, 6학년 학생 20명의 학생을 대상으로 1개월 간 창의적 체험활동 시간을 이용하여 10차시에 걸쳐 수업을 실시하였으며 수업 및 교재 적용 전 사전 설문과 적용 후 학생들의 성취도 및 만족도 설문을 실시하였다.

사전 설문결과 이번 연구 적용 이전에는 프로그래밍 교육을 받아 본적이 있는 학생이 전무한 것으로 나타나 이번이 학습이 재미있고 학습에 도움을 줄 것이라고 생각하는 학생이 85%로 매우 높은 것으로 나타나 새로운 것에 대한 두려움 보다는 평소 친근한 기기활용에 대한 기대가 높았다. 특히 스마트폰의 기본적인 기능이나 원리에 어느 정도 이상으로 잘 안다고 생각하는 학생이 35%정도로 스마트 폰에 대한 이해와 프로그래밍과 알고리즘의 이해가 부족하여 첫

학습부터 많은 시간을 소모하게 하였고 교재에 많은 수정이 요구되었다.

교재 적용 결과 설문에서 자신이 스마트폰에 대한 이해가 높아졌다는 의견이 90%로 대부분의 학생이 기기의 다양한 센서와 기능을 알게되었고 학습에 대한 만족도가 80%로 4명을 제외한 대다수의 학생들이 긍정적으로 생각하였다.

프로그래밍을 통한 생활 속에 활용할 수 있는 앱을 제작한다는데 45%의 학생들이, 스마트폰에 대한 이해가 높아진데 대해 35%의 만족감을 나타냈으며 앱이 어떻게 움직이는지 알고리즘을 이해하게 되었다고 답한 학생도 75%가 되었으며 이를 바탕으로 나만의 상황에 맞는 앱을 만들 수 있을 것으로 생각하는 학생도 60%로 절반이상의 학생이 긍정적인 대답을 하였다. 반면에 35% 학생들은 따라가기 식이 아닌 계속적으로 분석하고 생각을 많이 하도록 요구하는 교재의 구성에 부정적인 대답을 하므로 인하여 학생들이 이런 구성에 대한 적응이 부족함을 알 수 있었다.

5. 결론

지식정보화 사회로의 변화는 기존 전통적인 지식의 이해나 암기위주의 교육에서 창의적 사고, 문제해결력, 협동 능력을 길러줄 수 있는 교육으로 전환되어 가고 있으며 이러한 요구로 말미암아 2007년 이후 개정교육과정의 정보교육목표가 고등사고력과 문제해결력 고양에 있음에도 불구하고 컴퓨터 교육이 응용 소프트웨어 사용법 위주의 컴퓨터 활용 교육이 계속적으로 이루어져 왔다.

이에 컴퓨터 프로그래밍 교육은 자기 주도적 문제 해결을 통한 문제해결력과 고등사고력을 향상시킬 수 있다는 점에서 재조명 받고 있다.

하지만 기존의 프로그래밍 학습은 단순한 프로그램 명령어를 익히기 위한 관련성 없는 정보의 연속적인 나열로 이루어져있고 학생들과 실제적인 연관성이 없는 단편적이고 분절적인 구조화되지 않는 정보의 나열로 학습의 목표인 창의성 신장과 관련이 없는 단순한 프로그램 소양교육에 그쳤다.

하지만 본 연구는 프로그래밍 학습을 통하여 창의성, 논리성을 신장시키기 위해 STEAM과 문제중심

학습을 구조를 활용하여 학습내용에 초등학생들의 생활과 관련된 상황 즉, 이야기(내러티브)로 구성하게 하였다.

이는 단순한 스토리가 아니라 치밀하고 구조적으로 구성된 내러티브로 소프트웨어 개발이론 워터폴 모델의 요구사항분석 단계에서 분석되고 설계되어 자기 주도적 학습이 이루어지도록 구성하였다.

이를 위해 프로그래밍 교육에서의 인지적 부담을 줄이기 쉽고 간단하게 프로그램을 구현할 수 있는 언어로 교육용 프로그래밍 언어인 웹 인벤터를 선택하였으며 이는 기존 프로그래밍 교육인 컴퓨터 소프트웨어 개발에서 스마트폰 애플리케이션의 개발로의 전환을 의미하고 스마트기기의 이해와 활용성을 높이는 데 기여할 것으로 생각된다.

본 연구는 특정한 프로그램의 활용법을 배우기보다는 초등학생들의 창의성과 고등사고력 향상에 목적을 두고 학습 할 수 있도록 구성하였다.

또한 초등학교 학생인 학습자의 수준을 고려하여 실 교육현장에서 활용되었을 때 부담없이 수업이 진행될 수 있도록 하였으며 교사 주도가 아닌 보조적 역할로 학생이 주가 되어 수업을 이끌어 나가도록 구성하였다. 실제 초등학교 5, 6학년학생 20명을 창의적 체험 활동시간을 이용하여 수업을 실시한 결과 학생들의 80%정도가 학습만족도가 높았고 대다수 학생들이 긍정적인 생각을 하고 흥미를 가졌다

본 연구에서 소프트웨어 개발과정을 활용한 학습 방법을 통하여 작은 소프트웨어 개발 프로젝트를 완성하고 더 나아가 작은 단위의 소프트웨어들이 합쳐진 복잡한 소프트웨어를 개발하는 계속적이고 반복·확대되는 소프트웨어 개발과정을 통하여 스스로 아이디어를 분석하고 이에 알맞은 여러 가지 프로그래밍 언어를 선택하여 문제를 해결할 수 있는 애플리케이션 개발 능력을 갖추게 될 것으로 기대한다.

참 고 문 헌

- [1] 강인애외 2명(2007), PBL의 실천적 이해: PBL수업을 위한 길라잡이, 문음사.
- [2] 강혜진(2004), 프로그래밍기초 능력 배양을 통한 아동의 논리적 사고력 신장에 관한 분석, 숙명여

자대학교 교육대학원 석사학위 논문.

[3] 국가정보화전략위원회, 교육과학기술부(2011), **인재대국으로 가는 길, 스마트교육 추진전략**.

[4] 김종진외 4명(2011), 현장 검증을 통한 교육용 프로그래밍 언어 교재 개발, **한국콘텐츠학회논문지 Vol.11 No.1**.

[5] 김종훈외 2명(2005), 프로그램 요소를 이용한 창의성 신장 교재 개발 연구, **컴퓨터교육학회 논문지 Vol.8 No.5**.

[6] 김천식(2012), 스마트러닝의 개념 및 방향, **IT PLUS, 2012 SPRING_vol.60 26-29**.

[7] 배영권(2006), **창의적 문제해결력 신장을 위한 유비쿼터스 환경의 로봇프로그래밍 교육 모형**, 한국교원대학교 대학원 박사학위논문.

[8] 배학진외 2명(2009), 문제 중심 학습을 적용한 스크래치 프로그래밍 교수 학습 모형, **컴퓨터교육학회논문지, 12권 3호**.

[9] 백영균, 우인상(1994), Logo프로그래밍의 수업방법이 문제해결력에 미치는 효과에 관한 연구, **교육공학연구, 9(1), pp.73-90**.

[10] 양윤석(2011), **시스템과 소프트웨어 공학에서 기술과 소프트웨어 개발 프로세스의 통합에 관한 사례 연구**, 금오공과대학교 대학원.

[11] 오상진(2003), **애니메이션 저작도구를 이용한 초등 정보과학영재용 프로그래밍 교육과정의 개발**, 경인교육대학교 교육대학원 석사학위논문.

[12] 유인환, 김태완(2006), MINDSTORMS을 이용한 프로그래밍 학습이 창의력에 미치는 효과, **한국컴퓨터 교육학회 논문지 9권 1호**.

[13] 이숙미, **문제중심학습 기반 프로그래밍 학습 사례 분석**, 숙명여자 대학교 교육대학원 석사학위논문.

[14] 이재인, 성영훈(2011), 초등학생을 위한 스토리텔링 기반 로봇 프로그래밍 교육 시스템, **한국정보교육학회 논문지 Vol.15 No. 2**.

[15] 이중현(2004), 시스템 요구사항 정의 프로세스 및 도구개발, 아주대학교 대학원 박사학위논문.

[16] 정숙경(2010), **자원기반학습시스템을 활용한 프로그래밍 수업이 중학생의 창의적 문제해결 성향에 미치는 영향**, 한국교원대학교 교육대학원.

[17] 최정훈(2011), 융합을 기반으로 하는 STEAM 교육이란?, **한국과학창의재단-월간과학창의 2월호**.

[18] Delisle, R.(1997), *How to use problem-based learning in the classroom*, Alexndria, VA: Association for Supervision and Curriculum Development.

[19] Metiri Group. (2003). The enGauge 21st Century Skills. Retrieved November 1, 2011, from <http://projects.brevardschools.org/21stCenturySkills/default.aspx>

저 자 소 개

설 문 규

1972 동아대학교 전자공학과 (공학사)
 1988 동아대학교 대학원 전자공학과(공학박사)
 1996 Visiting Researcher
 University of Saskatchewan
 2001 Visiting Professor University of Maryland
 1993- 현재 진주교육대학교 컴퓨터교육과 교수
 관심분야 : 스마트교육, 네트워크 통신, 디지털논리학
 e-mail : mgseol@cue.ac.kr



손 창 익

1998 진주교육대학교(교육학사)
 2001 진주교육대학교 초등컴퓨터교육학과(교육학 석사)
 2013- 현재 경남 창원 감천초등학교 교사
 관심분야 : 스마트 교육, 로봇 교육, STEAM 교육
 e-mail : netsci@empas.com

