

요구사항 데이터와 GUI 디자인 패턴 모델간의 Mapping 프로세스에 관한 연구

황아영*, 이동현**, 인호**

*고려대학교 컴퓨터정보통신대학원 소프트웨어공학과

**고려대학교 정보통신대학 컴퓨터학과

e-mail : ah6484@korea.ac.kr, tellmhenry@korea.ac.kr, hoh_in@korea.ac.kr

A Study on Mapping Process between Requirement Data and GUI Design Patterns Model

Ah-young Hwang*, Dong-hyun Lee**, Hoh Peter In**

*Graduate School of Computer and Information Technology, Korea University

**Dept. of Computer Science and Engineering, Korea University

요 약

웹 어플리케이션을 고객의 요구사항에 맞게 구현하기 위해서는 발주자와 수주자 간의 명확한 의사소통이 필요하다. 그러나, 잦은 요구사항의 변경과 원활하지 못한 의사소통으로 인해 요구사항의 추출 및 검증이 힘들다. 본 논문에서는 요구사항 데이터와 GUI 디자인 패턴 모델간의 Mapping 프로세스를 통해 신속하고 명확하게 고객의 요구사항을 정의하며, 프로젝트의 생산성을 높일 수 있는 방법을 제시한다. [3]

1. 서론

현대 S/W 개발 프로젝트를 성공적으로 이끌기 위해서는 많은 노력이 필요하다. 많은 업체들이 과다한 경쟁으로 어렵게 프로젝트를 수주하지만, 프로젝트 진행 시 복잡해진 비즈니스, 발주자의 높은 기대치 및 한정된 예산, 불가능한 일정 등으로 많은 개발자들은 힘들다. 특히나 고객의 모호한 요구사항 등으로 계속되는 요구사항 변경은 프로젝트를 성공이라는 결과에서 멀어지게 한다. [3, 4]

이런 이유로 프로젝트 초기 단계에 고객의 요구사항을 명확하게 파악하기 위해 수 많은 업무 설계자는 무수히 노력을 한다. 요구사항 수집의 중요성이 점점 부각되면서, 많은 연구물들이 쏟아졌다.

하지만, 그럼에도 불구하고 빠르고 정확한 요구사항 수집을 할 수 있는 쉽고 정확한 방법은 찾기 힘들다. 변경이 잦은 천방지축 요구사항들을 제어하는 방법들은 하나 같이 방법이 복잡하고 사용 범위가 한정적이라, 프로젝트 내에서 쉽게 적용하기가 불가능했다. 또한 비즈니스 중심의 요구사항 데이터에 대한 체계적으로 관리에 대한 연구 부족은 요구사항 데이터 같은 중요한 재사용 자산들의 활용을 어렵게 하였다. 따라서, 짧은 납기 안에 세밀한 설계로 고객의 요구사항에 부합되는 어플리케이션을 개발 하기 위해 쉽고 정확하게 요구사항 데이터를 수집하는 체계적인 과정이 필요하다.

본 논문에서는 체계적인 요구사항 데이터를 수집하기 위해 웹 서비스의 표준인 UDDI 데이터 모델을 적용하여 요구사항 UDDI 데이터 모델을 만들고 광범위

한 요구사항 데이터를 축적 가능하게 할 것이다. 이를 통해 현재의 프로젝트의 조건과 부합되는 축적된 요구사항 데이터를 찾아 요구사항 리스트를 사진에 예측해 보고 정확하고 신속한 고객의 요구사항 데이터 도출 및 검증이 가능하게 한다. 그리고 이러한 요구사항 데이터를 정형화된 GUI 디자인 패턴 모델과 접목시켜, 업무 설계자가 좀 더 쉽고 빠르게 S/W 개발을 할 수 있도록 한다. 이는 빠른 프로토타입 구현이 가능하도록 하여, 개발 소요시간 단축 및 개발 시스템 품질 향상을 가져오고 개발한 시스템의 유지보수를 용이하게 해준다. [6]

본 논문의 구성은 다음과 같다. 제 1 장에서는 본 연구를 수행하게 된 배경과 목적을 살펴보고 본 연구의 구성을 제시한다. 2 장에서는 논문의 관련 연구를 살펴본다. 3 장에서는 요구사항 데이터와 GUI 디자인 패턴 모델의 mapping 프로세스를 제시한다. 4 장에서는 3 장에서 제시한 프로세스의 적용 및 기존 업무 설계 방법과의 비교 분석, 평가를 제시한다. 5 장에서는 결론 및 향후 연구과제를 제시한다.

2. 관련연구

2.1 웹 어플리케이션 UI 디자인 패턴

2006 년 웹 어플리케이션 개발을 위한 UI 디자인 패턴 모델에 관한 논문이 발표되었다. 이 논문에서는 개발자 측면에서의 효용성을 높고 표준을 제시해 줄 UI 디자인 패턴 모델을 제시했다. 이전까지의 UI 디자인 패턴은 그 영역과 대상이 매우 광범위 하며, 재사

용이 가능한 일정한 형식으로 정립되어 있지 않아 개발자들에게 혼동을 주었다[1].

여기서 제시한 10 가지 UI 디자인 패턴 모델은 업무 중심적인 웹 어플리케이션 상에서 보여지는 유사한 화면들을 패턴화시켜 개발자들도 UI 디자이너 없이 쉽게 화면을 제작할 수 있도록 하였다. 또한 사용자의 Activity(행동)에 의한 흐름을 패턴화 하였다.

하지만, 개발 생산성에 중심을 둔 UI 디자인 패턴 모델은 업무 기능의 분석 없이 화면 선택을 먼저 하는 잘못된 S/W 개발 방법을 조성하게 된다. 이는 고객과 개발자 사이의 의사 불일치 현상을 불러온다.

이러한 한계점을 극복하기 위해 앞으로 본 논문에서는 S/W 개발 설계 단계에서 업무 설계자가 고객의 요구사항을 바탕으로 하여 GUI 디자인 패턴 모델을 빠르게 적용할 수 있는 방안을 제시할 것이다.

2.2 UDDI 표준을 통한 데이터 수집기법

UDDI(Universal Description, Discovery and Integration)는 웹 서비스에 대한 정보를 등록하고 검색하기 위한 개념으로, 웹 서비스 시, 사용되는 서비스를 저장하고 재사용하려는 목적으로 만들어졌다.

UDDI 에서는 서비스를 효과적으로 관리하기 위해 4 가지 모델(Business Entity, Business Service, Binding Template, Technical Model)로 데이터를 관리한다. 다음은 4 가지 모델의 내용이다[2].

- ① **Business Entity:** 서비스를 공표하는 웹 서비스 제공자에 대한 정보.
- ② **Business Service:** 특정 웹 서비스에 대한 업무적, 기술적인 정보.
- ③ **Technical Model:** 웹 서비스에 대한 메타 데이터.
- ④ **Binding Template:** Business Service 와 Technical Model 을 연결하는 중간적 역할로 서비스 접점과 구조 명세에 대한 기술적 정보.

하지만 이전까지의 UDDI 저장소 설계는 UDDI 표준이 이런 명료한 데이터 구조를 제시함에도 불구하고, WSDL 제작에 너무 치우친 나머지, UDDI 표준의 적용을 세밀하게 하여, 데이터 수집 시 입력할 웹 서비스의 정보의 항목이 너무 많아 개발자들의 이해도가 떨어지며 사용도가 떨어지는 단점을 가지고 있다.[7]

본 논문에서는 이러한 UDDI 데이터 모델을 적용하여 간단하고 명료한 요구사항 데이터 모델을 자동적으로 요구사항을 적용하는 과정을 정의한다. 이는 요구사항 UDDI 저장소 구축 및 광범위(Enterprise)한 환경의 웹 어플리케이션 개발 시에 고객의 요구사항을 미리 예측하여 빠르게 프로토타입을 생성 가능한 프로세스를 제시한다.

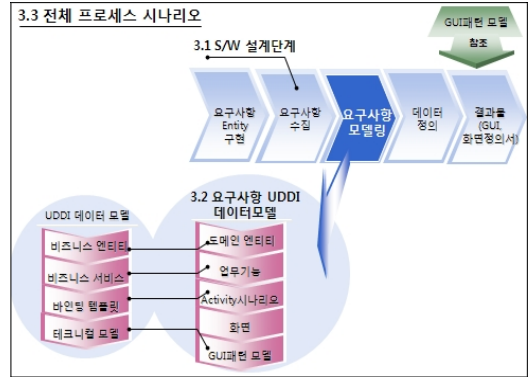
3. 요구사항 데이터와 UI 디자인 패턴 모델 Mapping

프로세스

전체 시나리오는 (그림 1)과 같다. 본 논문의 프로세스를 진행하기 위해서는 우선, 데이터베이스에 3.2 에서 제시할 요구사항 Entity 를 정의해 놓아야 한다. 만약, 웹 서비스 기능을 이용하여 웹 어플리케이션을

구현한다면 Server 측 데이터베이스에만 구현되어 있으면 된다.

이후, 프로젝트의 S/W 설계단계에서 업무 설계자는 업무 기능 담당자에게서 요구사항을 수집하고, 수집한 요구사항 데이터를 모델링 하여, 이를 요구사항 UDDI 저장소에 저장 및 결과물(GUI, 화면정의서)을 얻는다.



(그림 1) 프로세스 단계

3.1 프로세스 범위 (S/W 설계단계)

요구사항 정의는 S/W 개발 라이프 사이클 중 분석, 설계 단계에서 이루어 진다. 하지만 이 두 단계에서 얻어지는 요구사항은 각각 성격이 다르다.

분석단계에서 얻어지는 요구사항은 비기능적 요구사항과 H/W, S/W 장비 및 환경 전반에 걸친 큰 범위 기능 요구사항이 정의된다. 따라서 발주자 중 어플리케이션 총괄 책임자를 대상으로 요구사항 정의가 일어나게 된다.

반면, 분석단계를 거쳐 설계단계로 넘어오게 되면, 좀더 세밀한 요구사항 정의가 일어나게 된다. 업무 설계자는 발주자인 업무 담당 실무자를 통해 업무 기능 단위의 요구사항을 얻게 된다. 즉, 화면 단위에 시나리오 및 기능 명세를 얻게 되어, 화면 정의와 데이터 설계가 이루어 진다.

본 논문에서는 설계 단계에 초점을 두어 업무 기능이 반영된 요구사항 데이터가 화면에 적용되는 프로세스를 정의한다.

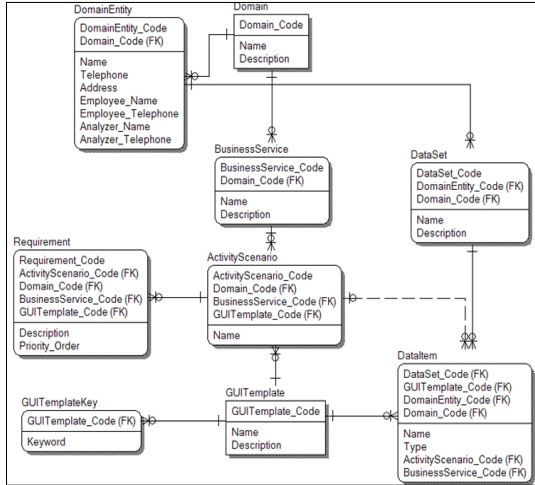
3.2 요구사항 UDDI 데이터 모델

고객의 요구사항을 짧은 시간에 정확히 얻어내기 위해 요구사항 데이터의 체계적인 수집과 사전에 얻어질 요구사항의 예측 과정이 필요하다. 이를 위해 웹 서비스의 표준인 UDDI 데이터 모델을 적용하여, 광범위하게 요구사항 데이터를 축적하며, 업무 설계자는 요구사항 UDDI 저장소를 통해 현재의 프로젝트의 조건과 부합되는 축적된 요구사항 데이터를 찾아내어 요구사항 리스트를 사전에 예측해 본다. <표 1>에서는 UDDI 데이터 모델을 요구사항 데이터 모델과 접목 시킨 것을 나타낸 표이다.

<표 1> 요구사항 UDDI 데이터 모델

UDDI 데이터모델	요구사항 UDDI 모델	기능	컬럼 내용
Business Entity	Domain Entity	Domain Client 정보	-도메인 영역(<예>금융업,제조업) -고객 정보
Business Service	Business Service	업무 기능 정보	-서비스 영역(<예>입출금업) -도메인 영역 -서비스 설명
Binding Template	Activity Scenario	Activity 단위 시나리오 정보	-시나리오 번호 -시나리오 내용 -GUI 디자인 패턴 코드 -서비스 영역 -도메인 영역
Technical Model	GUI Template	GUI 디자인 패턴	-GUI 디자인 패턴 코드 -GUI 디자인 패턴 설명

(그림 2)은 위의 표를 바탕으로 요구사항 데이터를 구조화 한 Entity 다이어그램이다. 이를 통해 요구사항 데이터를 일정한 형식에 맞게 요구사항 UDDI 저장소에 구조화 시킨다.



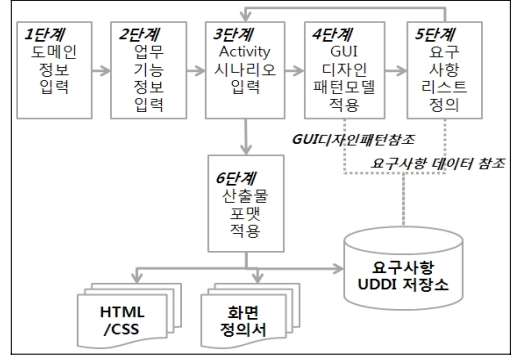
(그림 2) 요구사항 데이터 Entity Diagram

도메인 영역(<예>금융)은 하나 이상의 업무 기능들(<예> 입출금업무, 신규대출업무)을 가지며, 업무 기능은 하나 이상의 Activity 시나리오들(<예>계좌조회를 한다, 입금을 한다)을 가진다. 이러한 Activity 시나리오의 GUI 디자인 패턴 모델을 중 하나의 패턴(<예> >List to Detail)과 mapping 된다.

3.3 전체 프로세스 시나리오

3.2 에서 요구사항 데이터 모델을 구조화시켰다. 이제, 업무 설계자는 고객의 요구사항을 프로세스의 순서에 맞게 정의한다. (그림 3)과 같이 총 6 단계의 프로세스에 의해 요구사항 데이터는 정보 입력과

GUI 디자인 패턴 모델과의 접목의 과정을 거치며 그 결과로 요구사항 UDDI 저장소에 저장 및 산출물(화면정의서, HTML 등)이 생성된다. [6]



(그림 3) 요구사항 GUI Mapping 프로세스

1 단계에서는 도메인의 정보(고객 정보)를 등록하며, 2 단계에서는 1 단계에서 선택한 도메인 영역을 바탕으로 Service 영역 등록 등의 업무 기능 정보를 입력 한다. 3 단계에서는 업무 기능을 화면 단위의 시나리오들을 작성하고 각각은 자세하게 작성 가능하다. 하나의 Activity 별 시나리오로 GUI 디자인 패턴 적용 및 샘플 데이터를 입력하여 화면을 구성하고 상세 요구사항 리스트를 작성한다. 이렇게 3, 4, 5 단계를 반복하면 하나의 도메인과 업무기능 안의 여러 화면 시나리오들이 완성되어, 어플리케이션에 대한 프로토타입이 생성된다.

Activity 시나리오 별로 GUI 패턴 모델을 적용하는 4 단계와 요구사항을 등록하는 5 단계에서는 요구사항 UDDI 저장소에 저장된 요구사항 데이터의 조회 및 재 사용이 가능하다. 사전에 GUI 디자인 패턴 모델을 대상으로 설계자가 쓰기 쉬운 키워드를 추출해 놓고(<표 2>) DB 에 저장해 놓아야 한다. 요구사항 UDDI 저장소에 저장된 키워드는 3 단계에서 Activity 시나리오를 통해 일치 여부를 파악하여, 사용 가능한 GUI 화면 패턴을 4 단계에 리스트화 시켜 보여준다.

<표 2> GUI 디자인 패턴 모델과 키워드 추출 예시

GUI 패턴모델	영문이름	한글이름	키워드
P1-1	Single Detail [View to Edit]	자세한 단 건 정보(조회에서 수정)	Single Detail, View to Edit, 자세한 단 건, 조회&수정
P1-2	Single Detail [Edit]	자세한 단 건 정보 (수정)	Single Detail, Edit, 자세한 단 건, 수정
P2-1	Multi Detail [View to Edit]	자세한 단 건 정보(조회에서 수정)	Multi Detail, View to Edit, 자세한 단 건, 조회&수정

모든 단계가 끝나면 결과물로 GUI 와 화면정의서가 얻어진다. 이는 S/W 개발 라이브 사이클 구현 단

계로 넘어가기 이전에 발주자와 수주자 간의 명확한 요구사항 정의를 가능하게 한다.

4. 사례연구

본 논문에서는 위의 프로세스를 적용한 도구를 만들어 2006 년 시행한 E 사의 ‘연구비 사용 내역 관리 시스템’을 가지고 본 논문에서 제시한 요구사항 데이터를 GUI 패턴 모델에 접목시키는 프로세스를 적용해 보았다.

- ① 1 단계에서 ‘도메인 정의’에는 공공(연구소)를 선택하고 고객 사인 E 사의 정보를 정의한다. 여기서 정의한 정보는 요구사항 UDDI 저장소에 Domain Entity 테이블에 입력된다.
- ② 2 단계인 ‘업무 기능 정의’에는 만들고자 하는 ‘연구비 사용 내역 관리 시스템’의 정보를 기술한다.
- ③ 3 단계에서는 ‘연구비 사용 내역 관리 시스템’의 3 가지 시나리오인 ‘집행 내역 등록’, ‘집행내역 수정/취소’, ‘집행 내역 조회’의 정보를 Activity 시나리오에 각각 입력하고, 하나의 시나리오는 상세 정의 단계에서 구체화 시킨다.
- ④ ‘Activity 시나리오 정의’ 단계인 4 단계에서는 3 단계에서 입력한 시나리오의 정보를 가지고 요구사항 UDDI 저장소에서 얻어진 GUI 패턴 모델을 중에 원하는 화면 템플릿을 선택 가능하다. ‘집행내역 수정/취소’ Activity 에서는 화면 패턴 ‘3-1’ 을 선택하였으며, 이는 List to Detail 을 정의하는 GUI 디자인 패턴 모델이다.
- ⑤ Activity 에 맞는 화면 템플릿과 샘플 데이터를 입력 한 후, 5 단계에서는 Activity 에 대한 요구사항 들을 정의한다. 요구사항을 정의할 때는 이전의 유사 시스템의 요구사항 리스트를 참조할 수 있다. ‘집행내역 수정/취소’ Activity 에서는 ‘집행 내역은 페이지 형태여야 한다.’ 등과 같은 요구사항 리스트가 접수 되었다.

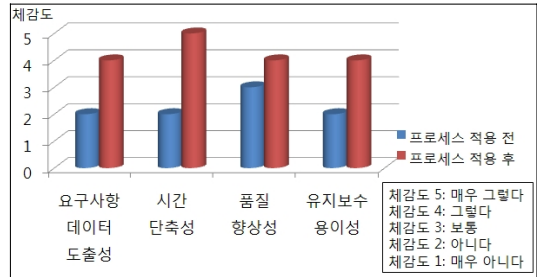
본 논문의 서론에서 밝혔던 연구 목적인 빠른 프로토타입 구현을 통한 S/W 프로젝트의 생산성 향상이 제대로 이루어졌는가를 검증하기 위해 위의 사례의 실제 업무 설계자와 도구를 사용하여 설계/개발을 한 개발자에게 개발 시스템 생산성 측면에 대해 설문 을 실시하였다. 설문 항목은 다음과 같으며, 이를 체감도 점수로 평가하였다(그림 4)[1].

- ✓ 업무 설계단계에서 정확하고 신속한 고객의 요구 사항 데이터 도출이 가능했다.
- ✓ 업무 설계 및 개발 소요시간은 짧다고 느껴졌다.
- ✓ 개발된 시스템 품질은 우수하다.
- ✓ 개발한 시스템의 유지보수가 용이할 것이다.

업무 설계 시 많은 변수들이 작용하나 (업무 담당자와의 회의 일정, 설계자의 몰입도 등) 본 논문에서 제시된 프로세스를 적용하면, 요구사항 데이터 표준 과 GUI 디자인 패턴 모델을 사용함으로써 설계/구현 단계에서 정제된 개발 표준을 따를 수 있었다. 이는 자산(Asset)의 재 사용률을 높이는 효과를 가져왔으며, 단기간 내에 HTML/CSS 등의 개발 프로토타입

구현체와 화면 정의서와 같은 산출물을 얻을 수 있어 업무 설계 시 신속성과 품질을 높여 주었다.

하지만, 업무 기능의 분류가 많은 경우 요구사항 데이터 모델의 확장이 요구되었으며, 이 경우 정확한 요구사항 데이터가 등록되지 못하는 단점이 있었다.



(그림 4) 개발 시스템 생산성 체감도 결과

5. 결론 및 향후 과제

본 논문에서는 요구사항 데이터를 관리하여 체계화 하고, GUI 디자인 패턴 모델과 접목하는 S/W 설계 단계의 프로세스를 살펴보았다. 제시된 프로세스를 사용함으로써 다음과 같은 장점을 얻을 수 있었다. 첫째, 업무 설계자가 요구사항 도출 시에 본 논문에서 제시한 프로세스를 통하여 고객이 요구사항을 효과적인 방식으로 선택, 결정할 수 있었다. 둘째, 과거에 정의했던, 요구사항 데이터를 사전에 검토하여, 놓치기 쉬운 요구사항 정의를 빠르게 파악할 수 있었다. 셋째, 개발의 표준을 준수함으로써 프로젝트의 생산성을 높일 수 있었다. 하지만 업무 기능이 여러 단계로 분류될 경우 정확한 요구사항 데이터가 등록 되지 못하는 경우가 발생할 수 있다.

향후, 이러한 점을 보완하여 웹 서비스 환경 하에 ‘요구사항 데이터와 GUI 디자인 패턴간의 Mapping 프로세스’ 를 적용하여 관찰하는 연구 및 발주자가 제시한 초기 데이터를 가지고 Entity 패턴을 만들어 본 연구에 적용하는 S/W 개발 자동화 영역 연구로 연구를 폭을 확장할 것이다.

참고문헌

- [1] 김창검, “웹 어플리케이션 개발을 위한 UI 디자인 패턴 모델 개발과 효용성에 관한 연구”, 홍익대학교 멀티미디어 커뮤니케이션, 2007
- [2] Oasis organization, www.oasis-open.org
- [3] Paul Haung, “Using modeling and simulation for rapid prototyping and system integration”, Minnesota Univ.
- [4] 한정현, 인호 “기민한 컴포넌트 개발 프로세스”, 제 29 회 정보처리학회 추계학술대회, 2008
- [5] 윤용진, “웹 어플리케이션 사용자의 요구분석 프로세스에 관한 연구”, 전남대학교, 2002
- [6] 박철호, “요구사항 정의 프로세스의 개선에 관한 연구”, 아주대학교, 2006
- [7] 김영민, “웹 서비스의 품질정보 자동수집기법”, 숭실대학교. 2007