

---

**Term Final Project Process Design Report**  
**- with Business Work Flow U-Engine**

June 7, 2007  
Seoul National Univ.

99406-025 Minwoo Ahn  
2000-11969 Kichul Kang  
2000-11998 Jaehong Lee

## — 차례 —

<b>1. Introduction</b>	<b>1</b>
<b>2. U-Engine의 특징</b>	<b>3</b>
2.1. U-Engine의 탄생 배경	3
2.2. U-Engine의 일반적 특성	5
2.3. U-Engine 의 기능구성	6
2.4. U-Engine 의 화면구성	8
<b>3. Process Design</b>	<b>10</b>
3.1. About Policy in Designing Process	10
3.2. Introduction to Macro-business Flow	21
3.3. Buyer Side Business Process	3
3.4. Supplier side business process	5
3.5. Manufacturer side business process	6
3.6. Total Business Flow	17
3.7. Next Step	8
<b>4. Process Execution</b>	<b>19</b>
4.1. U-Engine을 이용한 실제 Process Design	19
4.1.1. 전체 Process Design	19
4.1.2. Sub Process Design	2
4.1.3. Sub - Sub Process Design	22
4.2. Process 실행 화면	3
4.2.1. Buyer Quote 작성 화면	3
4.2.2. Supplier의 decision 화면	4
4.2.3. Process 완료 화면	5
<b>5. 추후 개발 사항</b>	<b>27</b>
5.1. User별 작업 구축	27
5.2. FTP 연동	27
5.3. DB 연동	27
5.4. Web application 연동	27
<b>6. Conclusion</b>	<b>28</b>

## － 그림 차례 －

[그림 1] BPMS의 주요 기능 .....	1
[그림 2] U-Engine 사용자 화면 구성 .....	8
[그림 3] Order 전까지의 Process .....	2
[그림 4] Order 이후의 Process .....	2
[그림 5] Typical Quote .....	4
[그림 6] Total Business Flow .....	7
[그림 7] 전체 Process Design .....	9
[그림 8] Sub Process Design .....	2
[그림 9] Sub - Sub Process Design .....	2
[그림 10] Quote 확인 및 입력창 .....	3
[그림 11] Quote 입력화면(Excel) .....	32
[그림 12] Decision Process .....	4
[그림 13] Decision 입력창 .....	4
[그림 14] Process 완료 화면 .....	5
[그림 15] Process 완료 후의 변수 및 리소스 .....	6

## 1. Introduction

최근 변화의 속도가 가속되고 있다. 속도뿐만이 아니라 변화의 조류 또한 쉽사리 예측할 수 없을 만큼 급변하고 있다. 급변하는 환경 속에 생존을 위해 노력하는 것은 동식물뿐만이 아니다. 변화하는 환경 속에 또 다른 유기체인 기업도 생사를 건 변화에 대한 대응 노력을 하고 있다. 이런 변화 대응 요구에 부응하기 위한 기업용 솔루션이 있다. 관심의 중심에 있는 이 솔루션은 BPMS(Business Process Management System)이다.

BPMS(Business Process Management System)란 BPM의 개념을 실현 가능하게 만들어주는 솔루션이다. BPMS는 프로세스들을 효율적으로 통합함으로써 기업 내 비효율성을 감소시키고 궁극적으로 기업의 이윤을 극대화 시킨다. BPMS는 프로세스에 대한 보다 나은 이해를 제공하고, 프로세스 실행 오류를 감소시킴으로써 프로세스 실행 비용을 최소화하며, 프로세스의 투명한 관리, 측정 및 외부 비즈니스와의 조정을 가능하게 해준다. BPMS의 주요 기능은 아래 [그림 1]과 같다.



[그림 1] BPMS의 주요 기능<sup>1)</sup>

1) <http://tong.nate.com/rockzero/31224984>에서 따옴.

이번 프로젝트에서는 오픈소스 기반의 BPMS인 U-Engine을 이용하여 자동차 구매 프로세스를 직접 디자인해보고, 실제로 시뮬레이션을 통해 그 가능성을 살펴보기로 한다.

## 2. U-Engine의 특징

### 2.1. U-Engine의 탄생 배경<sup>2)</sup>

BPM(Business Process Management)이 우리에게 약속한 가장 큰 의미 중 하나는 업무담당자가 엔지니어의 어떠한 도움 없이도 직접 업무 프로세스를 관리 / 실행할 수 있게 한다는 것이었다. 이는 간단한 Workflow 시나리오 상에서 Built-in된 몇 가지 Activity들의 조합으로 가능하다는 것을 수많은 시연을 통해 보여주고 있다. 하지만, 한 번 쯤 Application들과 사람간의 복잡한 업무 통합을 제공해야하는 좀 더 복잡한 시나리오에 직면해보라. 이러한 한정된 표현력만으로 정확히 원하는 프로세스 모델링을 하기란 불가능한 일이 되고야 만다. 이는 결과적으로 소프트웨어 벤더나 시스템 엔지니어로부터의 많은 도움을 야기하며, 결국 수많은 코딩 작업을 유발하게 된다. 생각을 해보자, 개발자의 도움 없이 어떻게 기술적으로 지속적으로 발전해가고 있고, 복잡하게 엉켜있는 조직 내의 업무 환경을 BPM의 제한된 틀 내에서 완벽하게 설계할 수 있을까? 이것은 처음부터 불가능한 이야기일지도 모른다. 오히려 우리의 BPM 모델링 환경에는 어떠한 개발자에 의한 기술적인 해석과정을 거칠 수 있는 시스템적인 환경이 필요하다. 즉, 현재의 BPMS는 하부기술의 추상화 기능과 표현력의 확장과 같은 기능들을 절대적으로 요구받고 있다.

개발자의 구현중심적인 SW개발 결과물을 업무분석가가 최대한 쉽게 이해하고 조립할 수 있도록 제시된 환경은 여러 가지가 있었다. 그중 UML과 객체지향 프레임워크는 최근까지 가장 각광받아오던 방법론으로 각자 MDA와 컴포넌트 조립도구의 영역까지 발전해 왔었다. 하지만 BPM의 ‘프로세스 중심 통합 방법론’의 대두와 함께 기존의 기술들이 웬지 상호 배타적인 기술들로 받아들여지고 있는 것이 현실이다. 하지만 U-Engine은

2) <http://www.uengine.org/korean/index.html>에서 발췌

이 두 기술들이 상호보완적인 성격을 지녀야 한다는 발상을 통한 BPM의 아키텍처를 제시한다.

만약 객체프레임워크가 BPM도메인에 맞춰져서 설계될 수 있다면, 그리고 Activity의 구현이 컴포넌트로 래핑되어서 그 프레임워크에 플러그인 되는 아키텍처를 갖고 있다면, 한번 프로그래머가 기술적인 내용을 기반으로 하여 업무 분석가가 자주 사용하게 될 Activity의 유형을 컴포넌트화하는 과정을 통해 일종의 '기술적 해석과정'을 거친 후 이기 때문에 그 Activity 컴포넌트를 조합하여 프로세스를 설계하는 업무분석가는 아주 쉽게 프로세스를 설계할 수 있을 것이다. 이것이 U-Engine의 착상이다. U-Engine은 BPM도메인에 맞추어 설계된 컴포넌트 프레임워크이며 조립 도구이다. J2EE가 복잡한 Web Transaction System에서 비즈니스 로직을 분리해내어 개발 할 수 있도록 제공한 범용 컴포넌트 프레임워크라고 한다면, U-Engine 컴포넌트 프레임워크는 개발자로 하여금 복잡한 BPM기반 Activity 개발에 대한 Workflow 이론, EAI, Web Services, GUI개발 같은 개념이 없이 쉽게 Activity Type유형을 만들 수 있게 프레임워크를 제공한다. 이러한 접근방법은 비즈니스 프로세스의 지속적인 개선뿐만 아니라 조직내 프로세스 설계 표현력의 지속적인 개선을 도모하게 한다.

즉, U-Engine은 객체프레임워크의 장점을 차용하여 BPMS아키텍처에 기술적 추상화와 표현의 확장성을 더한 제품이다.

## 2.2. U-Engine의 일반적 특성

1. Embedded : 컴포넌트 프레임워크 아키텍처를 기반으로, U-Engine은 어떤 환경이나 기존 제품에 연결, 설정, 적용 작업이 쉽다.
2. Powerful : U-Engine은 Page-flow로 부터, Decision (Rule) flow, Workflow, EAI flow, to B2Bi flow에 이르는 대부분의 유형의 프로세스 패턴을 지원한다.
3. Fully-loaded : Liferay Portal, Mondrian OLAP Server, Apache Axis Web Services Container와 같은 최고의 오픈 소스 제품들을 통합하여 대부분의 사용 BPMS가 제공하는 기능을 커버한다.
4. Open source : 무료로 U-Engine을 사용, 판매, 임베드 할수 있다.



## 2.3. U-Engine 의 기능구성

U-Engine은 사용자의 권한에 따라 크게 일반사용자 영역과 업무관리자 영역의 포탈환경을 제공한다.

첫 번째는 일반사용자 영역으로서 다음과 같은 기능을 제공한다.

- ① 새로운 업무를 시작할 수 있습니다
- ② 할당된 업무를 처리할 수 있는 목록을 제공합니다.
- ③ 참여하고 있는 업무의 진행상태를 조회할 수 있습니다.
- ③ 사용자의 권한에 따라 다양한 통계 자료를 제공합니다

즉, 업무진행자에게 업무를 처리하고 파악할 수 있는 통합된 환경을 제공한다.

두 번째는 업무관리 영역으로서 다음과 같은 기능을 제공한다.

- ① Process Instance 관리
- ② Process Definition 관리 및 Process Design
- ③ Form Application 관리

Process Instance 관리는 모든 Process Instance에 대한 검색 및 정보를 제공하며, Process Definition 관리는 Process Definition를 생성, 수정, 삭제하며 검색하는 기능을 제공한다..

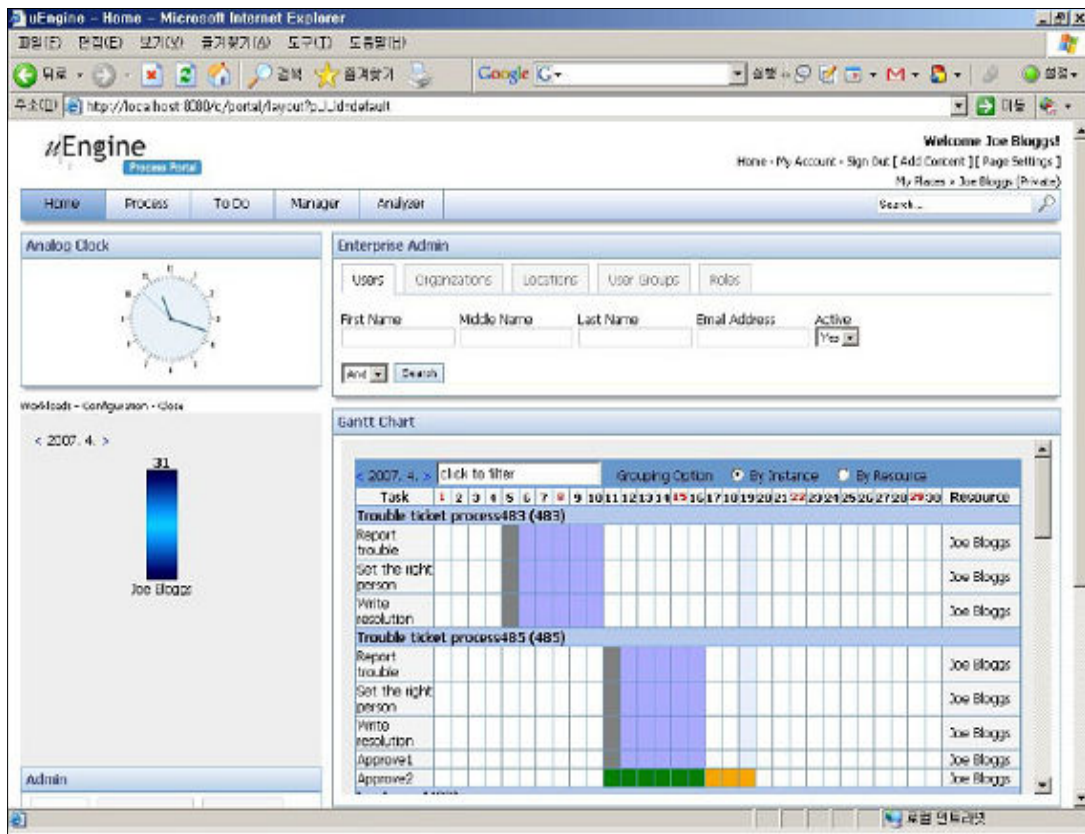
Process Design은 별도의 Design Tool을 통해 특별한 프로그램 코딩 없이 손쉽게 Process를 모델링하고 변경 할 수 있도록 Drag and Drop 방식의 Activity 타입을 제공하고 있으며, 또한 Process Design과 동시에 역할 정의 및 변수를 정의할 수 있다.

Form Application은 Process 상에서 사용하는 Form을 관리한다. Form Application에서 정의된 모든 Form은 Process상에서 양식이나 Application

으로 활용될 수 있으며 폼은 다양한 Form Field 및 Pre-built Component 방식으로 손쉽게 생성 할 수 있도록 기능을 제공한다.

## 2.4. U-Engine 의 화면구성

U-Engine화면의 구성은 크게 2개의 프레임으로 구분되어 있다.



[그림 2] U-Engine 사용자 화면 구성

[그림 2]에서 보이듯이 메뉴 영역인 상단부분과 사용 영역으로 구성된다. 메뉴 영역은 Home, Initiate, My Work, ProcessMgr, Welcome, Shopping 등의 Portlet Page들을 링크하고 있으며, 각 메뉴를 선택함에 따라 사용영역에 그 Portlet Page가 보이게 된다.

Portlet Page	설명
Home	Gantt Chart, Workload 등 각종 정보를 한눈에 볼 수 있다.
Process	새로운 업무프로세스를 시작할 수 있으며, 프로세스 흐름을 한눈에 볼 수 있도록 되어 있다.
To-Do	현재 자신에게 할당된 업무에 대해 List 형식으로 볼 수 있다.
Manager	프로세스를 관리할 수 있고, 현재 진행중인 프로세스를 볼 수 있다.

[표 1] U-Engine 포틀릿 페이지 구성

### 3. Process Design

#### 3.1. About Policy in Designing Process

본 MIS 프로젝트에는 가상의 사용자와 공급자, 제조업자가 존재한다. 이러한 비즈니스 주체들의 Business Workflow를 제시하기 위해서는 몇 가지 전제조건이 필요하다. 이러한 전제조건이 없으면, 너무나 변수가 다양해지기 때문에, 프로세스 자체가 의미 없이 복잡해질 수 있기 때문이다. 따라서 이러한 전제조건들을 가정된 가상 시장 상황에 대한 정책(Policy)으로서 정리하고자 한다.

- 시장에는 하나의 Buyer와 3개의 Supplier와 5개의 Manufacturer가 있다.
- 하나의 Buyer가 어떤 Supplier와도 계약하지 못할 경우, 프로세스는 Buyer가 계약을 못한 것으로 종료된다.
- 하나의 Buyer는 하나의 Supplier와만 계약이 성사된다.
- 하나의 Supplier는 복수의 Manufacturer와 계약할 수 있다.
- Buyer, Supplier, Manufacturer는 각각 제안을 독립적으로 거부할 수 있다. 단, 모든 Supplier와 모든 Manufacturer가 제안을 거부했을 경우, Buyer는 선택의 여지가 없이 계약을 못하게 되므로, 프로세스는 종료된다.
- Supplier는 Buyer로부터 받은 quote를 독단적으로 수정요청 할 수 없다. 반드시 Manufacturer와 상의 하에 수정요청을 한다.
- Supplier가 Manufacturer에게 요구하는 BOM은 부품단위까지 내려가지 않고, 차량의 옵션이나 도장과 같은 4~5개의 선택사항 정도로 정한다. (본 리포트 Supplement: 견적서 참고)
- 각 비즈니스 주체의 커뮤니케이션에는 반드시 Confirmation message

의 과정이 따른다. 예를 들어, Buyer가 Supplier에게 Quote를 보낼 때, Supplier는 정해진 기일 안에 반드시 Confirmation Message를 보내야 한다. 정해진 기일 안에 Confirmation Message를 못 받을 경우, Buyer는 다시 요청을 하게 된다. 시스템의 복잡도를 줄이기 위해 이 요청과정은 무한루프이며, 정해진 기일은 7일(5 영업일)로 가정한다.

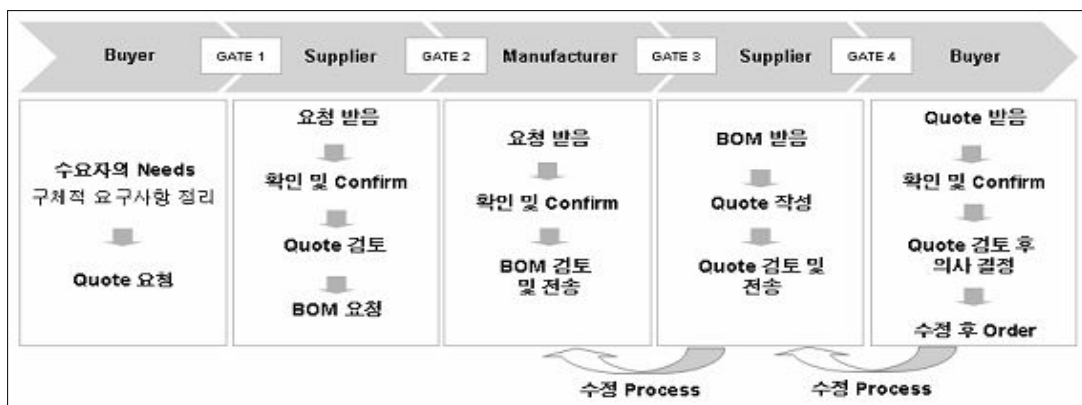
- Order가 이루어지면, 불량이나 하자는 Supplier가 조사하고 조치한다. Buyer는 주문된 제품에 대해 클레임을 제기할 필요가 없다.

### 3.2. Introduction to Macro-business Flow

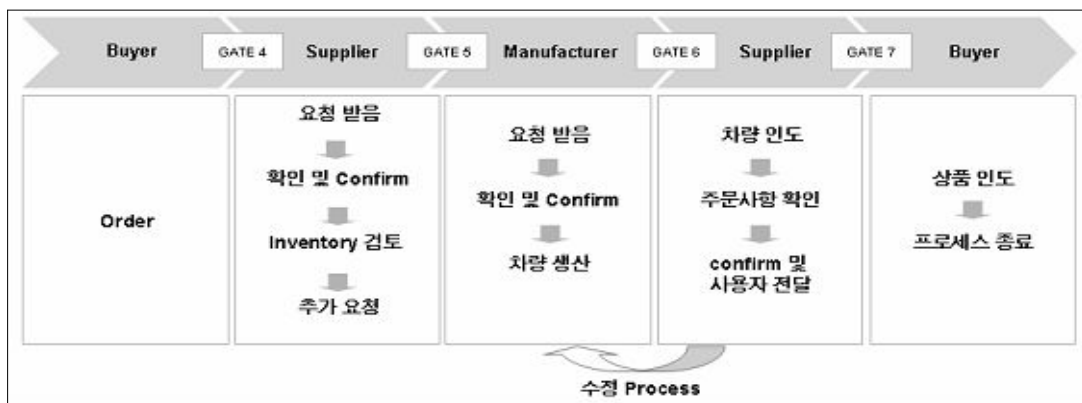
전체적인 Business flow는 다음과 같이 나뉘어서 볼 수 있다.

- ① Buyer가 order를 내리기까지 의 과정
- ② Buyer가 order를 내린 이후의 과정

각 Business flow를 중요한 의사결정을 중심으로 거시적인 프로세스 구분을 하면 다음과 같다. 각 Gate는 중요한 의사결정이 이루어지는 부분이다.



[그림 3] Order 전까지의 Process



[그림 4] Order 이후의 Process

### 3.3. Buyer Side Business Process

Buyer의 입장에서 일어나는 Business Process는 순차적으로 다음과 같이 발생한다.

단, 하나의 Buyer는 3개의 Supplier와 접촉할 수 있으므로, quote에 대해서 모든 안을 거부할 수도, 모두 수용할 수도 있다. 아래의 business flow는 하나의 Buyer가 하나의 Supplier와 대응할 때를 기준으로 Design된 것이다.

- 
1. Buyer의 Needs 발생.
  2. Quote request를 Supplier에게 전달.
  3. Supplier로부터 Confirmation message를 받음
  4. Supplier로부터 확정된 Quote를 받음
  5. Confirmation message를 Supplier에게 보냄
  6. Quote와 관련된 의사결정을 함.
    - A. Modify 요청 → Step 2로 이동
    - B. Quote Accept → Step 8로 이동
    - C. Reject → Step 7로 이동
  7. 해당 Supplier와는 거래 종료
  8. Place an order
  9. Supplier로부터 Confirmation message를 받음
  10. Supplier로부터 차량 생산 일정과 인도시기를 통보 받는다.
  11. 차량을 인도 받는다.
  12. 프로세스 완료
-



프로세스에서 확인할 수 있듯이, Buyer는 철저하게 Supplier와만 의사소통을 한다. Manufacturer와 직접 contact할 경우, business flow가 복잡해질 뿐만 아니라, Supplier의 역할이 불명확해진다. 그리고 Policy에서도 언급했듯이, order가 발생한 이후에는 제품의 생산과 인도는 Supplier의 책임이 된다. 따라서, Buyer의 입장에서는 order를 내리기까지의 business flow가 더 중요하고 더 복잡하다.

Buyer, Supplier, Manufacturer 간에 전달되는 Quote의 flatform은 다음 [그림 5]와 같다.

	A	B	C	D	E	F	G	H	I
1	Quotation For S1								
2									
3									
4	From : M1				To : S1				
5									
6	차종	Package사양	선택사항	색상	차량대수	가격(Price Per Unit)	전체 가격(Total Price)		
7	SM518MT	PLUS II	불소도장	흰색	2	12,140,000	₩24,280,000		
8	SM518AT	CUSTOM I	불소도장	금모래색	0	13,380,000	₩0		
9	SM520MT	AV I	ABS+에어백	회청자색	0	13,700,000	₩0		
10	SM520AT	DELUXE I	조수석에어백	암녹색	0	14,940,000	₩0		
11	SM520SEMT	PRIME I	앞사이드에어백	검정색	0	14,750,000	₩0		
12	SM520SEAT	CUSTOM II	조수석에어백	순노랑색	3	15,990,000	₩47,970,000		
13	SM520V	PLUS II	앞사이드에어백	백진주색	0	19,710,000	₩0		
14	SM525V	CUSTOM IV	ABS	백진주/금모래색	4	24,200,000	₩96,800,000		
15	고급형택시MT	DELUXE II	(택시)AL휠	검정색/회색	0	10,300,000	₩0		
16	고급형택시AT	PRIME II	(택시)AL휠	순노랑색	0	11,650,000	₩0		
17	모범형택시MT	VALUE I	(택시)ABS+AL휠	백진주색	5	11,950,000	₩59,750,000		
18	모범형택시AT	VALUE II	(택시)ABS+AL휠	백진주/금모래색	0	13,300,000	₩0		
19	장애인AT	SAFETY I	(장애)AL휠	검정색/회색	0	13,910,000	₩0		
20									
21							Sub Total :	₩228,800,000	
22							Shipping :	₩2,000,000	
23							Tax :	₩18,304,000	
24							Total :	₩249,104,000	
25									
26									
27									
28									
29									

[그림 5] Typical Quote

### 3.4. Supplier side business process

Supplier의 입장에서 발생하는 process는 다음의 순서를 따른다. Buyer와 Manufacturer의 중간 역할을 하기 때문에, 비교적 많은 workflow를 소화해야 한다.

- 
1. Buyer로부터 구매에 대한 요청을 받음.
  2. 확인 후 Buyer에게 confirmation message을 보냄.
  3. Manufacturer에게 BOM을 보냄.
  4. Manufacturer로부터 quote를 받음.
  5. Manufacturer가 보낸 quote에 대한 의사결정.
    - A. Quote modify → Step 3로 이동
    - B. Quote accept → Step 6으로이동
    - C. Quote reject → 해당manufacturer와의 거래 종료 및 결과 통보.
  6. Manufacturer에게 confirmation message을 보냄.
  7. Buyer의 요청서에 대한 의사결정.
    - A. 요청서 modify → buyer side business process Step 4로 이동
    - B. 요청서 accept → Step 8로 이동
    - C. 요청서 reject → Buyer와의 거래 종료 및 결과 통보.
  8. Buyer에게confirmation message을 보냄.
  9. Buyer로부터 order를 받음.
  10. Buyer에게 confirmation message을 보냄.
  11. Inventory 확인.
    - A. 충분할 시 → Step 14로 이동
    - B. 부족할 시 → Step 12로 이동
  12. Manufacturer에게 order.
  13. Manufacturer로부터 차량 인수에 따른 의사결정
    - A. 결함 및 하자 발견 시 → Step 2로 이동
    - B. 그 외의 경우 → Step 14로 이동
  14. Buyer에게 차량 인계.
  15. 프로세스 종료
-

### 3.5. Manufacturer side business process

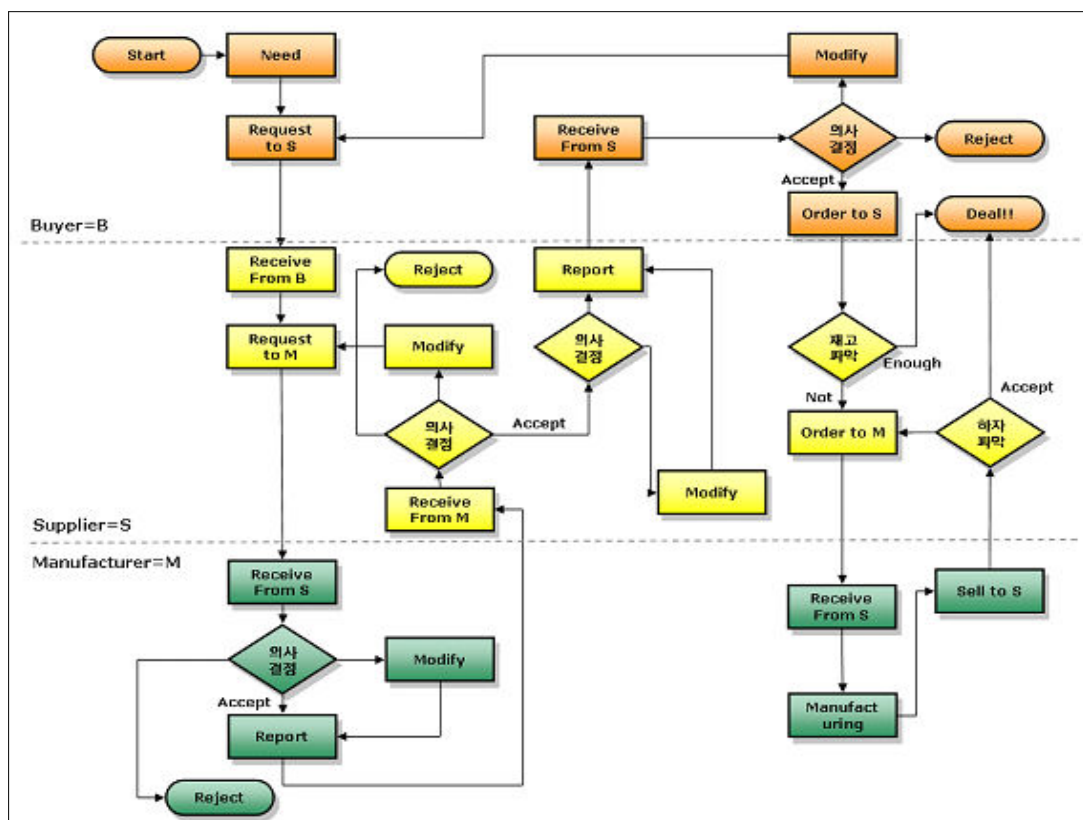
Manufacturer의 입장에서 발생하는 process는 다음의 순서를 따른다. 이 주체는 생산이 가장 중요한 부분을 차지하므로, Communication 과정이 복잡하지 않고 단순하다.

- 
1. Supplier로부터 quote를 받음.
  2. Supplier의 quote에 대한 의사결정
    - A. Quote modify → Supplier side business process step 4로 이동
    - B. Quote accept → Step 3로 이동
    - C. Quote reject → Supplier와의 거래 종료 및 결과 통보.
  3. Supplier에게 confirmation message을 보냄.
  4. Supplier로부터 order를 받음.
  5. Supplier에게 confirmation message을 보냄(due day).
  6. 차량 생산.
  7. Supplier에게 차량 인계.
  8. 프로세스 종료.
-

### 3.6. Total Business Flow

지금까지 Business 주체별로 Design된 business process를 Flowchart로 표현하면 다음과 같다.

- ① Buyer 영역(주황색): 한 번의 의사결정 과정이 있고, order가 발생한 이후에는 거래가 성사되는 과정만 남는다.
- ② Supplier 영역(노란색): Manufacturer와 Buyer 사이에서 3번의 의사결정을 해야 한다. Business process가 가장 길고 복잡하다.
- ③ Manufacturer 영역(녹색): 한 번의 의사결정 과정이 있고, Business process가 가장 간단하다.



[그림 6] Total Business Flow

### 3.7. Next Step

U-engine을 이용하여 앞에서 정의한 process를 구현한 후 simulation을 통해 process monitoring 및 analysis를 할 것이다. 이를 바탕으로 process를 optimization하여 보다 강력하고 효율적인 process로 개선시킬 것이다. 이를 위한 단계 및 세부사항은 다음과 같다.

- 
1. U-engine을 이용한 process 구현.
    - A. 설계한 Workflow를 U-engine상에서 구성.
    - B. 세부 변수 고려된 DB 구성 및 연동
  2. 구현된 process monitoring 및 측정.
    - A. Simulation을 통한 monitoring.
    - B. 전체 process 상황 조회.
    - C. Process별 상태 조회.
    - D. Process 실행 추적 및 node counting.
  3. Process analysis.
    - A. Process 오류 분석
    - B. Step 2의 자료를 바탕으로 비효율적 process 선별.
  4. Process optimization.
    - A. 개선 여지 판단
    - B. 기존의 문제점을 개선하는 process design.
  5. 기존 Process redesign → Step 1으로 이동
  6. 위의 단계를 반복적으로 실행하여 최적화된 Business process를 구축
-

## 4. Process Execution

### 4.1. U-Engine을 이용한 실제 Process Design

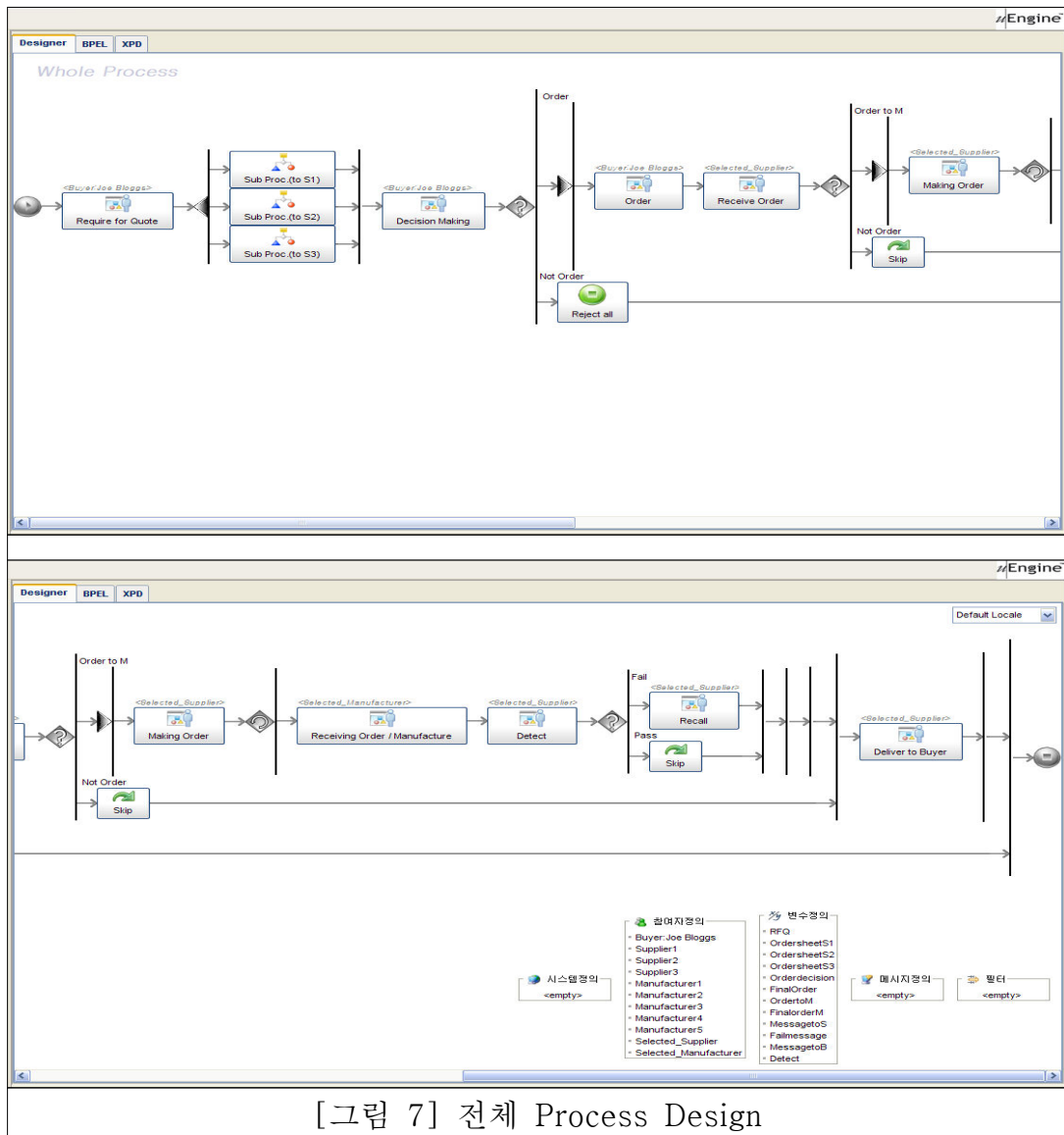
먼저 자동차 구매 Activity에 대한 전체 Process를 크게 하나 설계하였고, 그 전체 Process안에는 3개의 Sub Process가 존재한다. 그리고 각각의 Sub Process안에도 역시 3개씩의 Sub Process가 존재한다.

#### 4.1.1. 전체 Process Design

큰 Process는 다음의 순서와 같이 흘러간다.

- 
1. Buyer가 각각의 Supplier에게 Quote 요구
  2. 각각의 Supplier가 Quote를 요구 받은 후 각각의 Sub Process로 들어감
  3. 각 Supplier의 Sub Process 종료 후 Buyer가 구매여부 결정
    - A. Buyer가 Sub Process에서 나온 주문서들을 검토 후 주문하지 않기로 결정하면 Process 종료
    - B. Buyer가 주문을 결정하면 Step 4로 전개
  4. Buyer가 선택한 Supplier에게 주문
  5. 선택된 Supplier는 Buyer로부터 받은 Order를 바탕으로 재고를 파악한 후 생산의뢰 여부 결정
    - A. 재고량이 충분할 때 Step 8로 전개
    - B. 재고량이 충분하지 않으면 선택한 Manufacturer에게 생산을 의뢰하고, Step 6으로 전개
  6. 선택된 Manufacturer는 생산의뢰 받은 것을 바탕으로 생산을 실행하고, 생산된 자동차를 Supplier에게 인도
  7. Supplier는 인수받은 자동차에 대한 결함 검사
    - A. 인수받은 자동차에 결함이 없으면 Step 8로 전개
    - B. 인수받은 자동차에 결함이 발견되면, Step 5-B로 전개
  8. Supplier는 Buyer에게 자동차 인도 후 Process 종료
-

위에 흐름에 따라 Process를 디자인한 결과는 아래 그림과 같다.



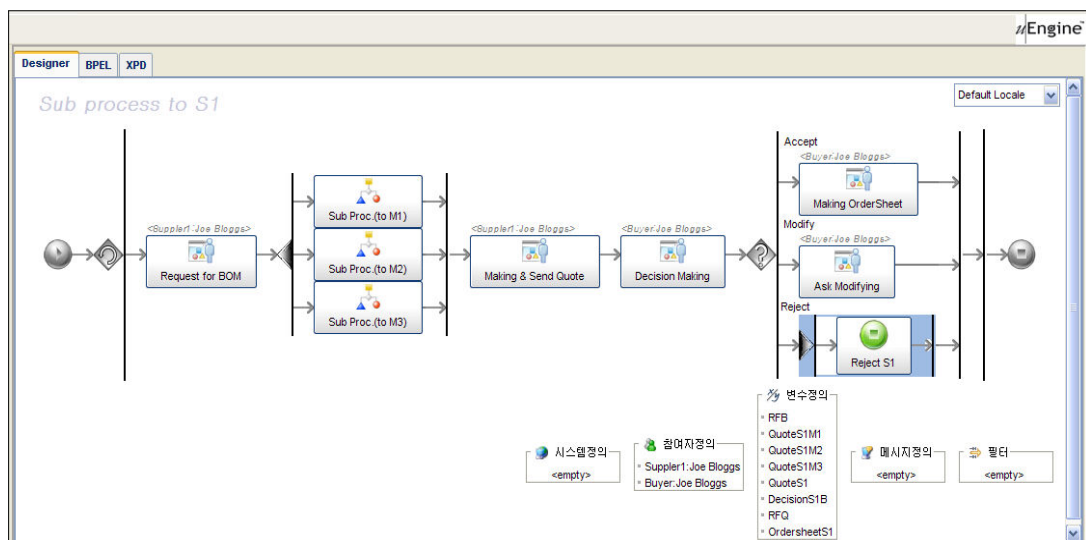
[그림 7] 전체 Process Design

#### 4.1.2. Sub Process Design

전체 Process 흐름 중 Step 2에 나오는 Sub Process의 흐름은 다음과 같다.

1. 각각의 Supplier는 요청받은 Quote를 바탕으로 자신들과 계약된 Manufacturer에게 BOM 요구
2. 각 Supplier와 계약된 Manufacturer는 BOM을 요구 받고 각각의 Sub Process로 전개
3. Supplier는 각각의 Sub Process에서 빠져나온 BOM을 바탕으로 Buyer에게 Quote 발송
4. Buyer는 Supplier로부터 받은 Quote를 바탕으로 의사결정
  - A. 받은 Quote를 Accept 하기로 결정하면 주문서 작성
  - B. 받은 Quote에 대한 Modify를 결정하면 Step 1으로 전개
  - C. 받은 Quote를 Reject 하기로 결정한다면 Sub Process 종료 후 Main Process로 빠져나감

위의 흐름에 따라 Process를 디자인한 결과는 아래 그림과 같다.



[그림 8] Sub Process Design

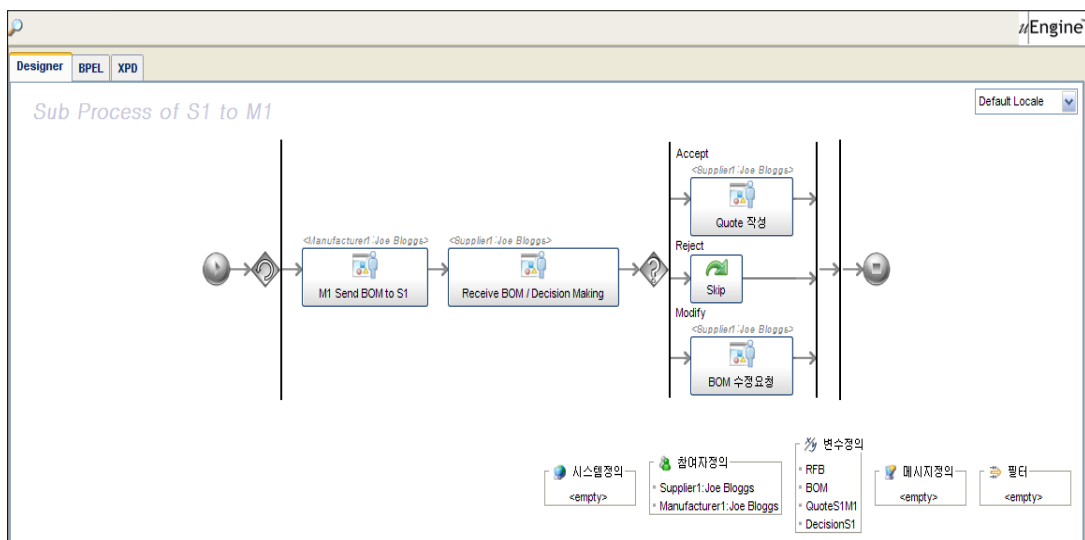


### 4.1.3. Sub - Sub Process Design

Sub Process Design의 흐름 중 Step 2에 나오는 Sub - Sub Process의 흐름은 다음과 같다.

1. 각각의 Manufacturer는 요구받은 BOM 작성 후 Supplier에게  
전송
2. Supplier는 BOM을 받은 후 의사결정
  - A. Accept 하기로 결정하면 Buyer에게 보낼 Quote 작성 후  
Process 빠져나감
  - B. Reject 하기로 결정하면, skip 후 Process 빠져나감
  - C. Modify를 요구하기로 결정하면 BOM 수정요청 후 Step 1  
으로 전개

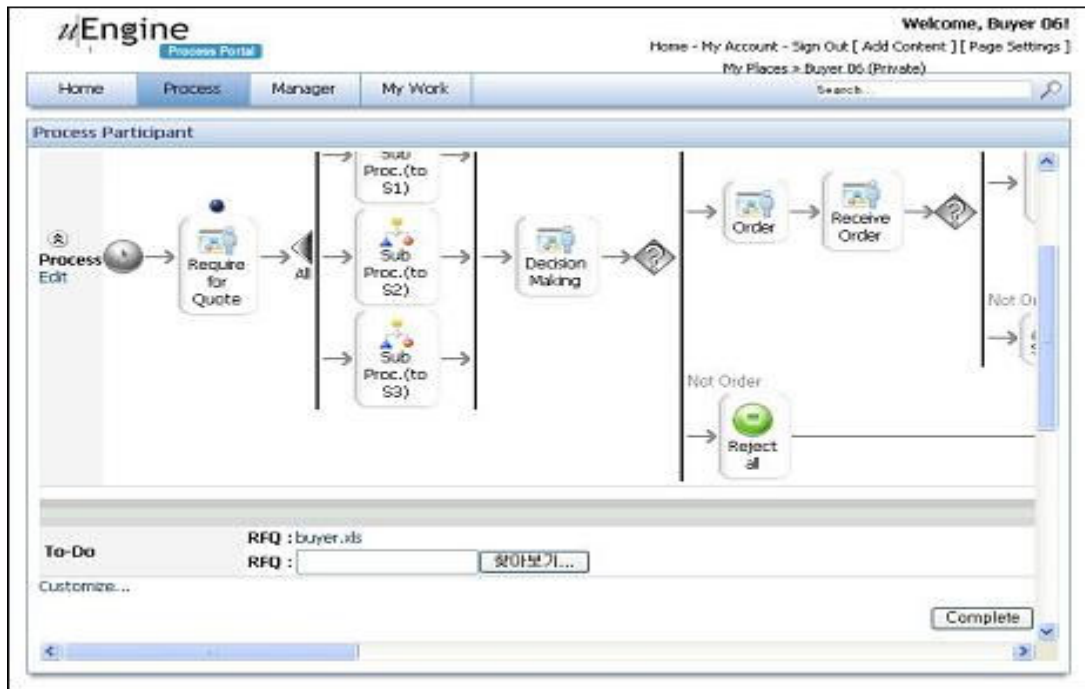
위에 흐름에 따라 Process를 디자인한 결과는 아래 그림과 같다.



[그림 9] Sub - Sub Process Design

## 4.2. Process 실행 화면

### 4.2.1. Buyer Quote 작성 화면

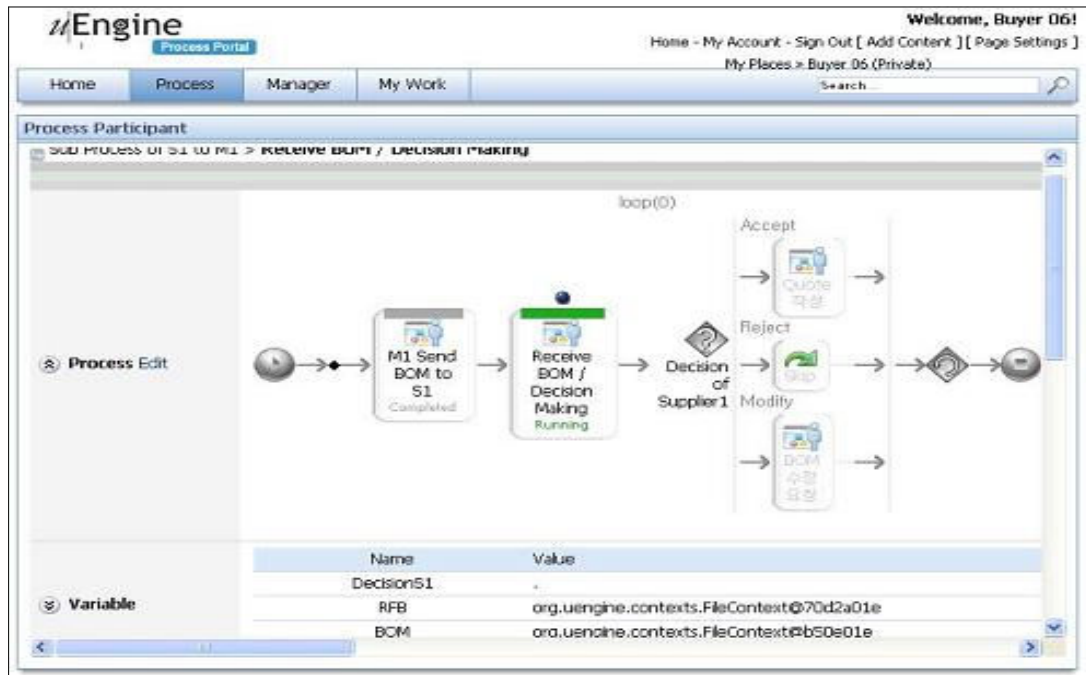


[그림 10] Quote 확인 및 입력창

From : Buyer				To : Supplier 1, 2, 3		
제품	Package/Size	단위	가격	수량	가격(Price Per Unit)	전체 가격(Total Price)
SMV10MT	PLUS II	물소도장	원래	0		₩0
SMV10AT	CUSTOM I	물소도장	금호여백	0		₩0
SMV20MT	AV I	ABS+에어백	화장자재	0		₩0
SMV20AT	DELUXE I	조수석에어백	입욕식	0		₩0
SMV20SEMT	PRIME I	앞사이드에어백	점점식	0		₩0
SMV20SEAT	CUSTOM II	조수석에어백	승노향식	0		₩0
SMV20V	PLUS II	앞사이드에어백	벽진주식	0		₩0
SMV25V	CUSTOM IV	ABS	벽진주/금호여백	0		₩0
구분명세/MT	NOI I VIE II	구분명세/MT	구분명세/MT	0		₩0

[그림 11] Quote 입력화면(Excel)

## 4.2.2. Supplier의 decision 화면



[그림 12] Decision Process

Name	Value
DecisionS1	.
RFB	org.usengine.contexts.FileContext@70d2a01e
BOM	org.usengine.contexts.FileContext@b50e01e
QuoteS1M1	org.usengine.contexts.FileContext@7772601e

Possible Actions: There's no actions currently activated.

Instruction

To-Do

BOM : 1181193458578\_buyer-3.xls

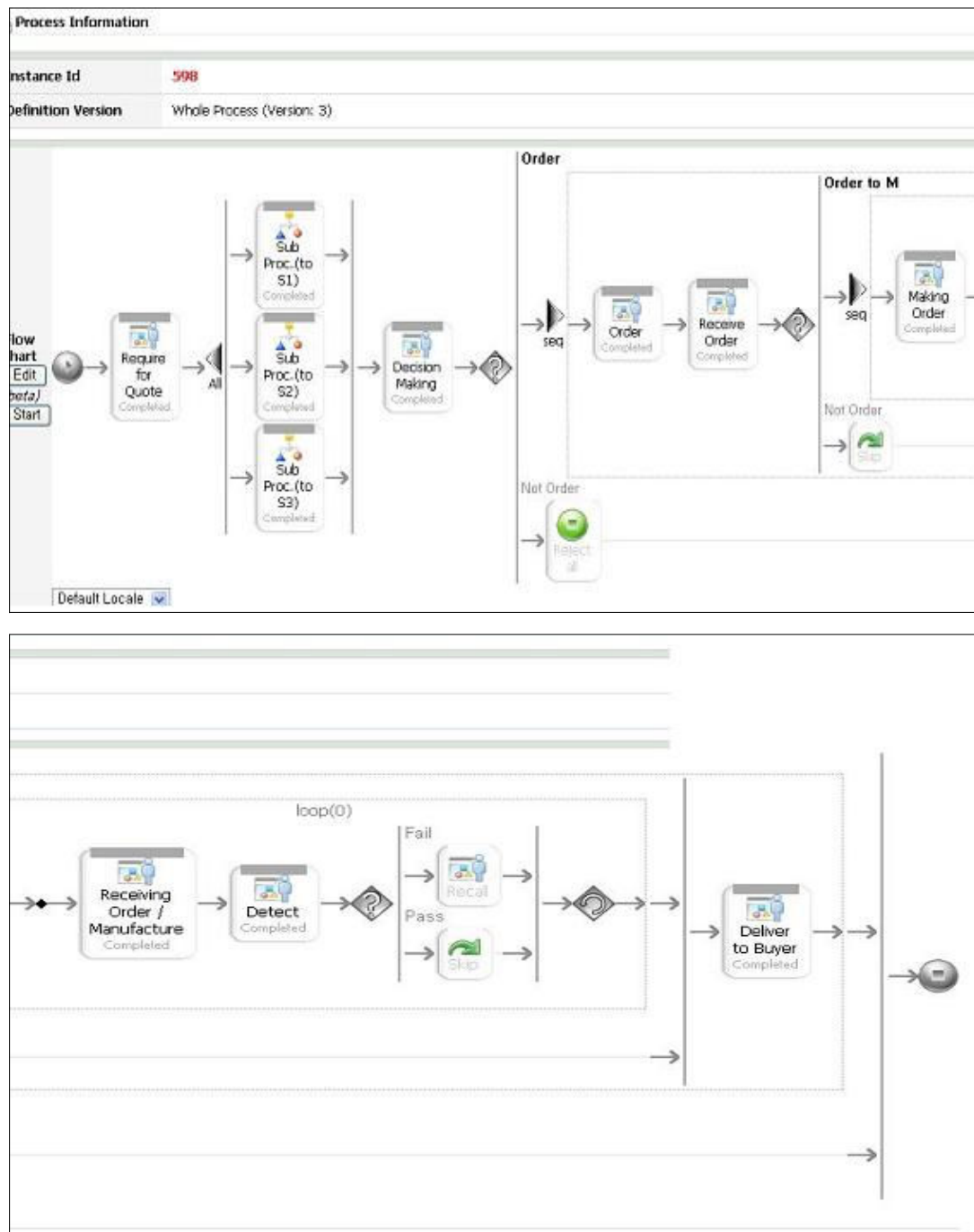
DecisionS1 : ☐ Accept ☐ Reject ☐ Modify

Customize...

Delegate Save Complete

[그림 13] Decision 입력창

### 4.2.3. Process 완료 화면



[그림 14] Process 완료 화면

		Name/Type	Value		
Process Variables		Orderdecision/java.lang.String	Yes	XML	change
		OrdertoM/java.lang.String	Yes	XML	change
		Detect/java.lang.String	Yes	XML	change
		RFQ/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@b26a0e1	XML	change
		OrdersheetS1/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@7282a0e1	XML	change
		OrdersheetS2/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@b6320e1	XML	change
		OrdersheetS3/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@715ba0e1	XML	change
		FinakorderM/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@2a02a0e1	XML	change
		MessageToS/java.lang.String	fdsa	XML	change
		MessageToB/java.lang.String	frewqfsa	XML	change
Resources		Failmessage/java.lang.String	null	XML	change
		FinalOrder/org.uengine.contexts.FileContext	org.uengine.contexts.FileContext@1054e0e1	XML	change
		Name	Binding		
		Buyer:Buyer 06	Buyer 06/liferay.com.1010/M	change	
		Supplier1	Buyer 06/liferay.com.1010/M	change	
		Supplier2	Buyer 06/liferay.com.1010/M	change	
		Supplier3	Buyer 06/liferay.com.1010/M	change	
		Manufacturer1	Buyer 06/liferay.com.1010/M	change	
		Manufacturer2	Buyer 06/liferay.com.1010/M	change	
		Manufacturer3	Buyer 06/liferay.com.1010/M	change	
		Manufacturer4	Buyer 06/liferay.com.1010/M	change	
		Manufacturer5	Buyer 06/liferay.com.1010/M	change	
		Selected_Supplier	Buyer 06/liferay.com.1010/M	change	
		Selected_Manufacturer	Buyer 06/liferay.com.1010/M	change	

[그림 15] Process 완료 후의 변수 및 리소스

## 5. 추후 개발 사항

### 5.1. User별 작업 구축

현재 프로세스는 project 발표에서 참여자의 model에 대한 이해도를 높이고, process 시현 시간 단축을 위해 한 명의 참가자가 모든 작업을 시행하게 구축하였다. 차후 실제 process에 참여하는 user 등록 및 작업 분담 설정을 하여 보다 현실적으로 구축하여야 할 것이다.

### 5.2. FTP 연동

현재 완성된 Program에서는 Local Excel file을 load하여 확인 및 수정 과정을 하게 된다. FTP 서버 확보 및 계정 생성을 하게 되면 Excel file을 FTP 서버에 올려 거리에 상관없는 process 작업이 가능하게 된다. 현재 노트북에서 FTP 서버를 실행해본 결과, U-Engine의 서버 작업만으로도 불안정한 상태를 보여 FTP 연동은 차후로 미루었다.

### 5.3. DB 연동

개발 초기에는 DB 연동을 통한 데이터 작업을 추구하였으나 개발 인력의 정보 부족으로 process의 변수를 저장하는 단계까지만 구현 가능하다는 것을 발견하였다. 추후 U-Engine과 DB의 연동으로 즉각적인 Analysis가 가능한 model을 세워야 할 것이다.

### 5.4. Web application 연동

Project 기한의 제약으로 인해 web application 구축은 제외하였다. Web application의 연동을 구현한다면 보다 나은 Interface 제공이 가능할 것이다.

## 6. Conclusion

전 장에서 보인 것과 같이 자동차 구매 프로세스가 끝이 났다.

실제로 BPMS를 사용하여 프로세스를 디자인해서 성공적으로 끝이 났다. 하지만 BPMS를 사용해서 성공했다는 것에만 의의를 두기에는 뭔가 아쉬움이 남는 프로젝트였다.

U-Engine을 사용하면서 아쉬웠던 점 몇 가지를 적어보았다.

첫 번째로 Debugging을 하는 것이 힘들었다는 점을 들고 싶다. 단순히 잘 몰라서 힘이 들었다는 것이 아니라, 프로세스 디자인을 마치고 디플로이를 해서 되지 않으면, 어디가 어떤 문제가 있어서 안되는 것인지 알 수 있는 방법이 전혀 없었다. 무작정 잘못되었다고 예상되는 지점에 가서 수정을 해보고 안되면 다른 곳을 수정하고 이런 식으로 될 때까지 해보는 수밖에 없었다.

두 번째로 User-Guide의 부실함을 지적하고 싶다. 프로세스를 디자인하는 과정에서 DB연동을 위해 수차례 User-Guide를 읽어보았지만 그 방법을 찾지 못해 사용을 포기해야만 했다. Guide에는 단순히 연동을 하는 방법만 나타나 있을 뿐, 자세한 사용방법이 나와있지 않았다.

마지막으로 브라우저 호환의 문제를 들고 싶다. 엑셀파일 연동을 하여 프로세스 디자인을 마치고 디플로이까지 다 된 상태에서 시뮬레이션을 하는 과정에서 엑셀파일이 깨져서 나오는 문제가 발생했다. 이런 저런 수정을 해봐도 좀처럼 문제가 해결되지 않았는데, 브라우저를 IE에서 FireFox로 바꾸고 나니 문

제가 해결되었다. 둘 다 오픈소스 기반의 프로그램이니까 혹시 하는 마음에 써 보았는데 다행히도 실행이 되었다. 만약 그 생각을 하지 못했더라면 프로젝트를 무사히 마치지 못했을 것이란 생각을 하면 아직도 식은 땀이 흐른다.

정말 많은 시행착오를 겪으면서 힘들게 한 작업이었지만, 수업시간에 배운 것처럼 전문적으로 코딩을 하는 사람이 아니더라도 충분히 사용이 가능한 프로그램을 이용해서 결과물을 산출했다는 것에 뿌듯함을 느낀다. (물론 고급기술을 사용하는 데는 전문적인 지식이 필요하긴 하다). 결론의 도입부에 적었던 것처럼 실제로 BPMS를 사용해보았다는 것이 이번 프로젝트를 통해 얻은 가장 큰 경험이었다. 그리고 U-Engine 또한 조금 더 문제점을 보완한다면 우수한 프로그램으로 거듭날 수 있을 것이라 생각한다.