

정책 평가 시뮬레이터 기능지원 전자상품 거래시장 솔루션

iForum은 사용자들에게 시장 agent interface를 제공하여 전자 상품(digital product)을 생산, 거래할 수 있도록 하며, 사용자가 제어하는 agent들을 기반으로 하는 전자 시장을 구현해서, 관리자가 이 시스템을 전자 시장에서의 정책을 평가할 수 있는 시뮬레이터로 이용할 수 있는 전자 시장 솔루션이다.

	List	Current
Financials	1	2
Tradeable	4	13
Marketplace	22	12
Products	0	117
Managing Capital	1000	1000
GM	1000	1000
Government Expenses	0	1000
Government	1000	1000
Country	1000	110000
System's Capital	1000	1100
Country's Bank	1000	100000
Country's Government	0	1000
Country's	0	10
Country's	0	10

<iForum 관리자 화면>

사람이 조종하는 agent-based 방식의 정책 평가 시뮬레이터는 기존의 agent-based 정책 평가 시뮬레이터보다 정책에 민감하게 반응하며, 시뮬레이터 스스로 발전할 수 있기 때문에 현실을 더 잘 반영할 수 있다.

이 전자 시장 솔루션이 기존의 지식 거래 사이트와 다른 점은, 첫째, 시장 관리자에게 기본적인 시장관리 기능은 물론, 관리자가 제도적으로 시장의 변수들을 변경해서 관리자가 원하는 방향으로 시장을 유도할 수 있는 기능을 제공하기 때문에 정보 경제학을 연구하는 사람들에게 저렴하고 현실적인 정보경제학적 모델을 제공할 수 있고, 둘째, 이 프로그램은 시장에서 부품으로 사용되는 전자 상품들을 조달해서 새로운 전자 상품을 생산하는 과정을 지원하기 때문에 전자 상품의 지적 재산권을 체계적으로 관리할 수 있다는 점이다.

이 전자 시장 솔루션이 기존의 지식 거래 사이트와 다른 점은, 첫째, 시장 관리자에게 기본적인 시장관리 기능은 물론, 관리자가 제도적으로 시장의 변수들을 변경해서 관리자가 원하는 방향으로 시장을 유도할 수 있는 기능을 제공하기 때문에 정보 경제학을 연구하는 사람들에게 저렴하고 현실적인 정보경제학적 모델을 제공할 수 있고, 둘째, 이 프로그램은 시장에서 부품으로 사용되는 전자 상품들을 조달해서 새로운 전자 상품을 생산하는 과정을 지원하기 때문에 전자 상품의 지적 재산권을 체계적으로 관리할 수 있다는 점이다.

또한 이 프로그램은 컴포넌트 기반 개발 방법을 사용해서 설계, 자바 빈 컴포넌트로 구현하였기 때문에 프로그램을 변경할 때 적은 부분만을 수정해서 빠르게 변화를 반영할 수 있다. 따라서 더 복잡한 전자 상품 시장 모델을 구현하거나 새로운 전자상거래 기술을 적용하고자 할 때 유리하다.

1. 작품명 : iForum (전자상품 거래시장 솔루션)

2. 제작자

대표자 : 최경환

개발참여자 : 최경환, 강동훈, 최은경, 이문헌

소속 : KAIST

대표 email : kywch0@hugsvr.kaist.ac.kr

3. S/W 요약설명

iForum은 사용자들에게 시장 agent interface를 제공하여 전자 상품(digital product)을 생산, 거래할 수 있도록 하며, 사용자가 제어하는 agent들을 기반으로 하는 전자 시장을 구현해서, 관리자가 이 시스템을 전자 시장에서의 정책을 평가할 수 있는 시뮬레이터로 이용할 수 있는 전자 시장 솔루션이다.

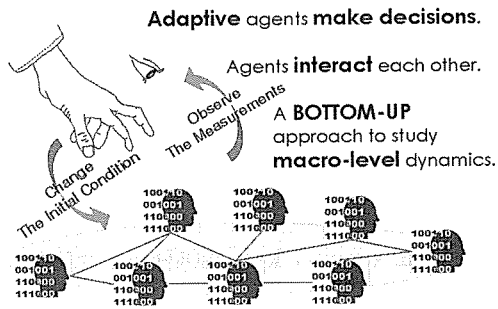
사람이 조종하는 agent-based 방식의 정책 평가 시뮬레이터는 기존의 agent-based 정책 평가 시뮬레이터보다 정책에 민감하게 반응하며, 시뮬레이터 스스로 발전할 수 있기 때문에 현실을 더 잘 반영할 수 있다.

이 전자 시장 솔루션이 기존의 지식 거래 사이트와 다른 점은, 첫째, 시장 관리자에게 기본적인 시장관리 기능은 물론, 관리자가 제도적으로 시장의 변수들을 변경해서 관리자가 원하는 방향으로 시장을 유도할 수 있는 기능을 제공하기 때문에 정보 경제학을 연구하는 사람들에게 저렴하고 현실적인 정보경제학적 모델을 제공할 수 있고, 둘째, 이 프로그램은 시장에서 부품으로 사용되는 전자 상품들을 조달해서 새로운 전자 상품을 생산하는 과정을 지원하기 때문에 전자 상품의 지적 재산을 체계적으로 관리할 수 있다는 점이다.

또한 이 프로그램은 컴포넌트 기반 개발 방법을 사용해서 설계, 자바 빈 컴포넌트로 구현하였기 때문에 프로그램에 추가하거나 수정할 기능이 있을 때 프로그램의 적은 부분만을 수정해서 빠르게 변화를 반영할 수 있다. 따라서 더 복잡한 전자 상품 시장 모델을 구현하거나 새로운 전자상거래 기술을 적용하고자 할 때, 적은 노력으로 원하는 바를 달성할 수 있다.

3.1 개발 배경 및 목적

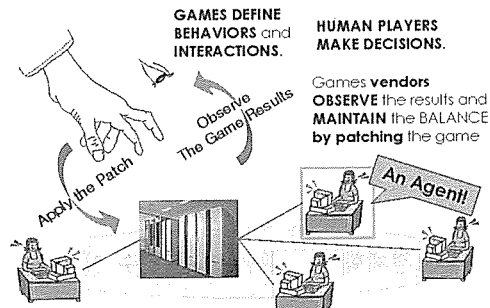
현재 시장을 agent based simulation을 통해서 분석하고, 이러한 agent 기반의 시장 모델을 시장 정책 평가에 응용하려는 시도가 많이 있다. 그러나 현재의 기술 수준으로는 프로그램된 agent들이 사람 수준의 판단 능력에 미치지 못하기 때문에 이러한 방법을



<Agent Based Simulation>

통해 사람들끼리 복잡한 과정을 거쳐 거래가 이루어지는 시장을 분석하거나, 이러한 시장에 적용하는 정책을 평가하려는 시도는 태생적 한계를 가진다. 이러한 한계를 극복할 수 있는 방법은 두가지다. 시장 Agent의 지능을 사람 수준으로 올리거나, 사람으로 하여금 각각의 시장 agent를 조정하도록 해서 시장 agent들이 사람의 판단능력을 빌리도록 하는 방법이 그것이다. 물론 첫 번째 방법은 현재로서는 불가능하다.

얼마 전, 고속 인터넷이 지금처럼 광범위하게 보급되기 전까지는 두번째 방법도 불가능했다. 그러나 지금 게임을 통해서 서비스되고 있는 Battle.Net, 리니지등의 대규모의 가상사회를 볼 때, 두번째 방법은 기술적으로 충분한 실현 가능성을 가진다.

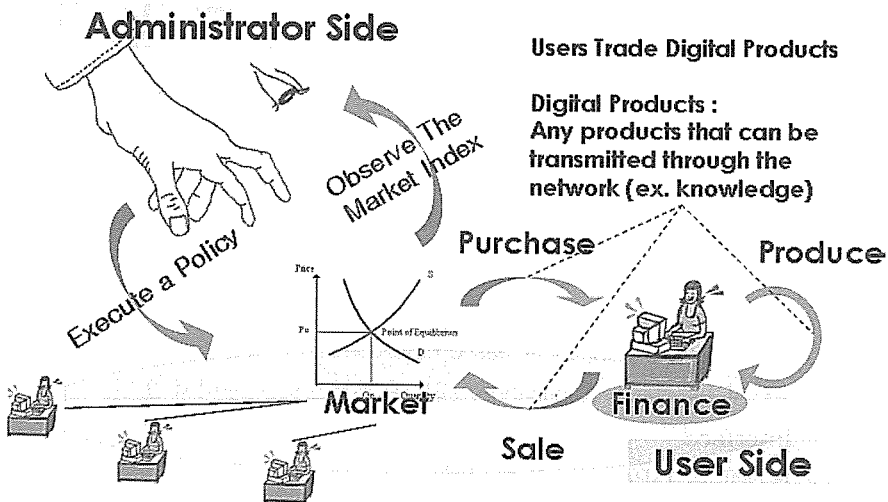


<게임으로 연결된 가상사회>

즉 가상사회 사용자들에게 각각 하나의 시장 agent interface를 제공하고 이를 조종하도록 한다면, 사람의 판단능력을 가진 시장 agent들로 구성된 시장을 구현할 수 있다.

이 프로그램은 사용자들에게 전자 상품(digital product)을 생산, 거래할 수 있는 시장 agent interface를 제공하여 이 agent들을 기반으로 하는 전자 시장을 구현해서, 관리자가 이 시스템을 전자 시장에서의 관리자 정책을 평가할 수 있는 시뮬레이션 시스템으로 이용할 수 있도록 하는 것을 목적으로 한다.

3.2 시스템 개요



이러한 목적을 달성하기 위해서 이 프로그램은 다음과 같은 기능을 제공하도록 설계되었다.

사용자 (User Side)

- Produce : 사용자가 전자 상품을 기획, 제작할 수 있다. 이 시스템에서는 사용자가 시장에서 자재를 조달해서 새로운 상품을 생산하는 과정을 지원한다.
- Purchase & Sale : 사용자가 상품 시장에서 상품을 사고 팔 수 있다. 이 시스템에서는 정가판매, 역경매방식의 상품 거래를 지원한다.
- Finance : 사용자가 자신의 재정상황을 볼 수 있고, 돈을 빌릴 수 있다.

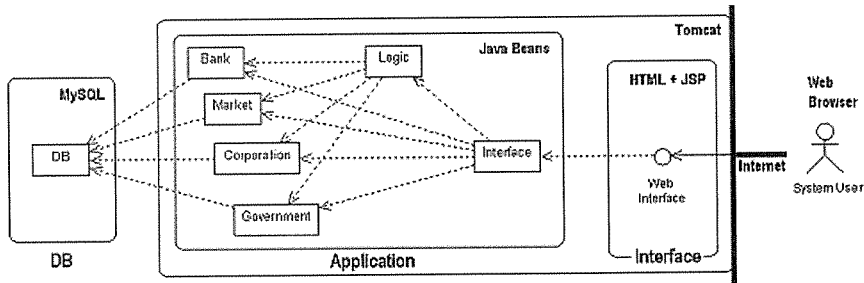
관리자 (Administrator Side)

- Execute a Policy : 관리자가 사용자를 관리하고, 시장의 질서를 유지하며,

시장을 관리하기 위해 재정정책과 금융정책을 집행한다.

- Observe The Market Index : 관리자가 시스템의 상황을 볼 수 있다. 이 기능은 관리자가 시스템에 적절한 정책을 선택하는데, 적용한 정책의 결과를 평가하는데 필요한 데이터를 관리자에게 제공한다.

3.3 시스템 구성



이 프로그램은 MySQL과 Java Beans, JSP와 HTML을 사용하여 구현되었다.

사용자들은 웹 브라우저를 사용해서 인터넷을 통해 이 프로그램을 사용할 수 있다. 프로그램은 관리자의 서버에만 설치되므로, 사용자는 프로그램을 받아 설치하지 않고 즉시 사용할 수 있다.

이 프로그램은 component based development 방법론을 참조하여 만들었으며, 크게 Interface, Application, DB 세 부분으로 나누어져 있다.

Interface

- HTML과 JSP를 사용하여 구현
- 프로그램의 실행부를 interface와 완전 분리, 따라서 사용자 interface부분을 쉽게 변경할 수 있다.

Application

- Java Beans를 사용하여 컴포넌트 아키텍처를 구현
- Interface, logic, entity의 세 계층을 가진다.
- Interface 계층은 application이 외부에 제공하는 기능들을 모두 모아놓아서 내부의 구조를 외부에 숨길 수 있도록 하고, 프로그램의 내부를 수정할 때 내부의 변화가 외부로 전파되는 것을 최소화 한다.
- Entity 계층은 application과 DB 부분을 이어준다. Entity 계층의 객체들은 서로 독립적이며, 다른 객체에 의존하지 않고 수행할 수 있는 기능들을 제공한다. Entity 계층은 Bank, Market, Corporation, Government 객체 모임

(Package)으로 이루어져 있다.

- Logic 계층은 다수의 entity 계층의 객체들을 조작해서 비즈니스 로직을 수행하는 객체들을 제공한다. Logic 계층에는 각각 금융거래 프로세스, 사용자사이 프로세스, 관리자정책 프로세스, 사용자관리 프로세스 묶음이 있다. 이렇게 Entity 계층과 Logic 계층이 분리된 아키텍처의 장점은 비즈니스 로직은 자주 바뀌나 이들이 조작하는 entity들은 자주 바뀌지 않기 때문에 비즈니스 로직을 변경하려고 할 때, application의 변화를 Logic 계층에 국한 시켜서 변화의 범위를 최소화 할 수 있는 것이다.

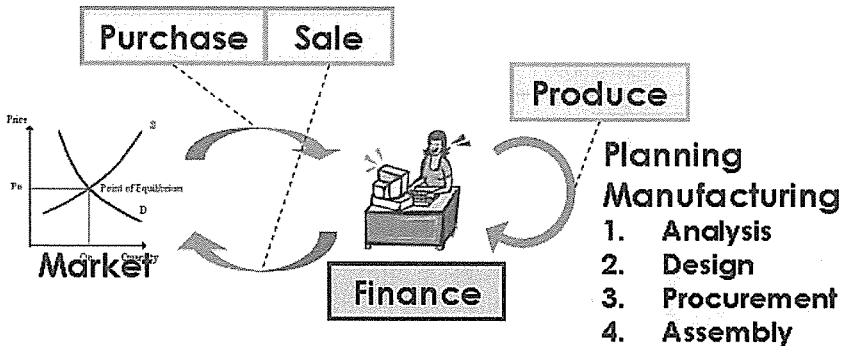
DB

- MySQL을 사용하여 구현

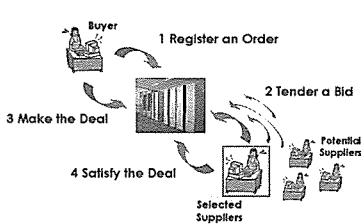
3.4 프로그램 기능

사용자에게 제공되는 기능

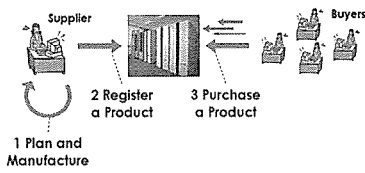
Make-to-Order Product Production / Distribution
Make-to-Stock Product Production / Distribution
Product Evaluation / Dispute Settlement



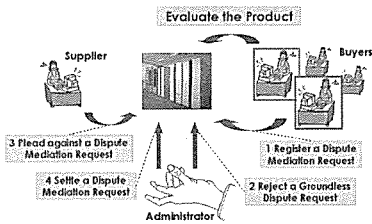
Purchase & Sale



- Make-to-Order Product Production / Distribution : 사용자는 상품을 역경매 방식으로 주문, 판매하는 방식으로 상품을 거래할 수 있다. 사용자는 맞춤 상품을 구매할 수 있다.

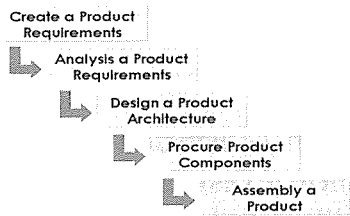


- Make-to-Stock Product Production / Distribution : 사용자는 상품을 정가판매 방식으로 주문, 판매하는 방식으로 상품을 거래할 수 있다. 사용자는 상품을 등록할 때 적절한 상품 분류 코드를 부여해야 한다.



- Product Evaluation : 사용자는 상품을 구매한 후 평가할 수 있다.
 - Dispute Settlement : 사용자는 구매한 상품에 불만을 제기하거나, 판매한 상품에 대한 불만에 응답할 수 있다.

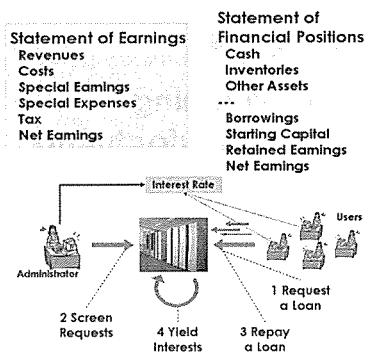
Produce



- Planning : 사용자는 새로운 상품의 기획안을 정리할 수 있다.

- Manufacturing : 사용자는 Analysis(분석), Design(설계), Procurement(구매), Assembly(조립)의 네 단계를 거쳐서 상품을 만들 수 있다.

Finance



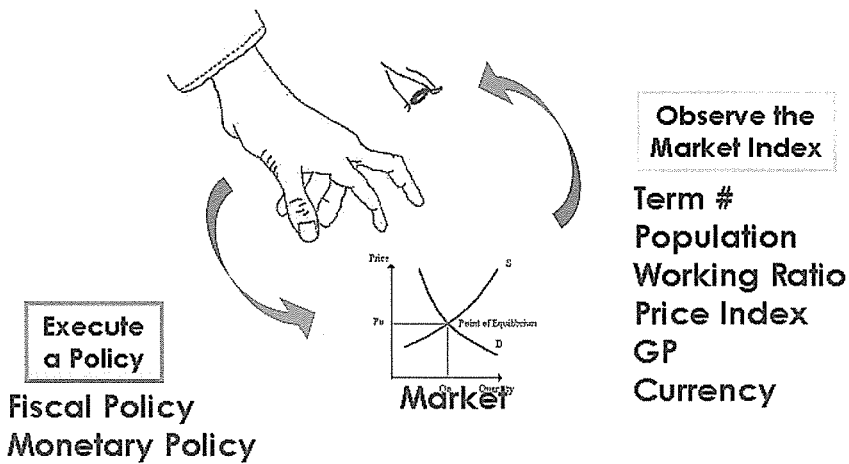
사용자는 자신의 재정상태를 실시간으로 제공되는 Statement of Earnings와 Statement of Financial Positions을 통해서 확인할 수 있다.

또한 사용자는 돈이 필요할 경우 대출을 신청할 수 있다.

관리자에게 제공되는 기능

Observe the Market Index

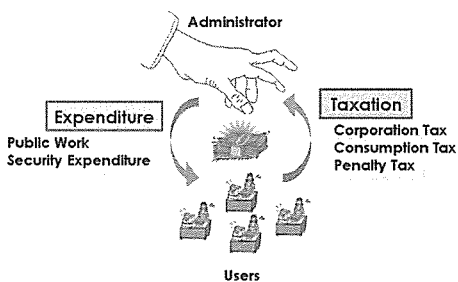
- Term # : 시스템의 시간 지표
- Population : 시스템에 등록된 사용자 총 수



<관리자에게 제공되는 기능>

- Working Ratio : 한 Term 동안 시스템에 새 상품을 등록한 사용자 비율
- Price Index : 한 Term 동안 거래된 모든 상품의 평균 가격
- GP(Gross Product) : 한 Term동안의 최종상품 거래액수 총액, 공공상품 거래액수 총액, 질서유지비용 총액의 합
- Currency : 사용자가 가진 화폐 총액, 정부가 가진 화폐 총액, 은행이 사용할 수 있는 화폐 총액의 합

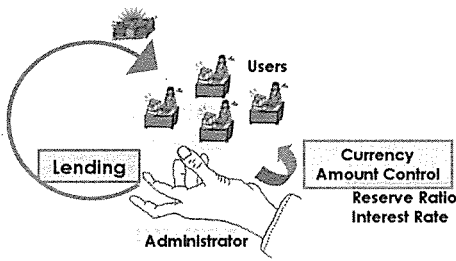
Fiscal Policy



- Taxation : 정부(관리자)는 매 term마다 징수하는 Corporation Tax, 매 거래마다 징수하는 Consumption Tax, 규칙 위반시 징수하는 Penalty Tax, 이 세가지의 징수 방식을 통해서 정부 재원을 충당한다.

- Expenditure : 정부는 공공상품을 발주하는 Public Work Expenditure, 질서 유지를 위해 사용되는 Security Expenditure등의 방식으로 예산을 집행한다.

Monetary Policy



- Currency Amount Control : 정부 (관리자)는 은행의 지급준비율 (Reserve Ratio)와 이자율(Interest Rate)를 조정해서 유통되는 통화량을 조절할 수 있다.
- Lending : 은행(관리자)은 정부 또는 사용자들에게 대출해줄 수 있다.

<Monetary Policy>

3.5 주요 특징

정책 평가 시뮬레이터

이 프로그램은 관리자에게 현실과 유사한 가상사회로 이루어진 정보 경제 모델을 제공하고, 관리자가 이 모델에 여러 다양한 정책을 수행할 수 있도록 하기 때문에, 정보 경제 사회의 정책 입안자들이나, 정책 연구자들에게 저렴한 사회적 비용으로 정교하고 현실적인 정책 평가를 할 수 있는 시스템을 제공한다.

대중적인 지식거래 유도

이 프로그램은 일반 사용자들에게 가상화폐를 무료로 제공하고 이 화폐로만 상품을 거래하도록 하였기 때문에, 사용자들이 부담없이 서로의 전자 상품을 거래할 수 있는 분위기를 제공해서 활발한 지식 거래와 새로운 지식의 생산을 유도할 수 있다.

체계적인 지적 재산권 관리

기존의 지식거래 사이트에서는 이미 등록된 상품을 이용해서 새로운 상품을 만들 경우 상품사이의 의존성을 관리하지 않기 때문에 지적재산권 분쟁이 생길 수 있다. 그러나 이 프로그램은 한 사용자가 다른 사용자의 전자 상품들을 이용해서 새로운 상품을 만드는 과정을 지원하기 때문에, 생산 과정에서 생기는 전자 상품 설계를 통해 상품 의존성을 체계적으로 관리할 수 있으며, 이를 이용해서 사용자의 지적 재산권을 보호할 수 있다.

컴포넌트 기반 개발

이 프로그램은 컴포넌트 기반 개발 방법을 사용해서 설계, 구현하였기 때문에 프로그램에 추가하거나 수정할 기능이 있을 때 프로그램을 적게 수정해

서 빠르게 변화를 반영할 수 있다.

3.6 응용 분야

전자 상품의 가격 정책 평가

인터넷의 발달로 점점 많은 전자 상품이 거래되고 있으나, 이러한 전자 상품에 어떻게 가격을 매길 것인가에 대한 연구는 아직 미흡하다. 가격 정책은 기업의 사활을 좌우하기 때문에 새로운 가격 정책은 현실에서 검증 받기가 쉽지 않다. 그러나 이 프로그램이 제공하는 시장 안에서 새로운 가격 정책을 시험해볼 경우 적은 위험과 비용으로 거래 데이터를 얻을 수 있으며, 이를 분석해서 가격 정책이 효과적이었는지를 검증할 수 있게 된다.

정보 경제 사회에서 발생할 수 있는 새로운 비즈니스 모델 평가

이 프로그램은 현실과 매우 유사한 가상사회로 이루어진 정보 경제 모델을 제공한다. 따라서 프로그램의 사용자는 새로운 비즈니스 모델을 사업화 하기 전에 먼저 적은 비용으로 이 프로그램이 제공하는 모델에 비즈니스 모델을 적용, 시험적으로 운영해보고 평가할 수 있다.

새로 개발된 전자상거래 기술의 시범 적용

이 프로그램이 제공하는 전자 상품 시장은 새로운 전자상거래 기술들을 빠르게 시험해 볼 수 있는 기회를 제공할 수 있다.

3.7 발전 방향

지식 관리 시스템

이 프로그램은 기본적인 지식의 생산, 공유, 평가, 분류 기능과 지식 활동 평가 기능을 제공한다. 이 기능들을 목적에 맞게 보강하고, 지식의 갱신과 검증 기능을 추가하면, 이 프로그램은 지식 관리 시스템으로 활용될 수 있다.

현실적인 정보 경제 시뮬레이터

사용자에게 더 많은 기업 행동을 제공하고, 관리자에게 더 정교한 정책수단을 제공해서 이 프로그램을 좀 더 현실적인 정보 경제 시뮬레이터로 변화시킬 수 있다.

사회 시뮬레이터

관리자에게 집중된 입법(시장 규칙 제정), 행정(정책 수행), 사법(분쟁 중재, 규칙 위반 판단)의 권력을 나누고, 사용자들에게 참정권을 나눠줘서 이 프로그램에 여러 형태의 정치 시스템을 도입할 수 있다. 사용자들에게 경제적 체

협뿐만이 아니라 정치적 체험을 할 수 있도록 함으로써 경제 시뮬레이션 기능을 포함하는 좀 더 복잡한 사회 시뮬레이터를 구현할 수 있다.

4. 개발단계별 기간 및 투입인원수

개발단계	개발시간	인원	비고
개념 설계	99. 5. 1~01. 6.31	1	시장 : 박상찬 교수님 지도 경제 : 유평일 교수님 지도
시스템설계	01. 7. 1~00. 8.31	1	CBD : 배두환 교수님 지도
프로그래밍	01. 9. 1~00.10.31	4	파트별 프로그램 제작
구현 계	4 개월	5	

5. 사용 또는 개발언어, TOOL

용도	개발 언어	비고
설계	UML	COOL:Joe 1.1
Server-Side Application	Java(Beans), SQL	
Application Interface	JSP, HTML	나모 웹에디터

6. 사용시스템

용도	CPU	RAM	OS
설계 & 디버그	Duron 800	640	Win. 2000 Prof.
개발 1	Pentium2, 400	448	Win. 2000 Adv.
개발 2	Pentium3, 550	256	Win. 2000 Prof.
디자인	Celeron, 650	384	Win. 2000 Prof.
프로그램 운용	Dual, Pentium3 700	768	Linux, 2.4.2-2