|  |  |
| --- | --- |
| 팀명 | 삼촌 |
| 팀장 이름 | 남 정호 |

# **SW 기본정보**

## SW명

* **Samchon Framework**

## SW 개요



### Samchon Framework

오픈 소스 프로젝트 Samchon Framework. C++의, 클라우드 서버 및 분산처리 시스템을 구현하기 위한, 프레임워크입니다. Samchon Framework는 크게 다음과 같은 목적으로 사용됩니다.

* + 1. 성능과 메모리 관리가 중요한 중대형 클라우드 시스템 구축
    2. 기존의 C++ 솔루션을 클라우드 서비스로 빌드
    3. 표준적인 메시지 프로토콜과 데이터 표기법을 이용, 통합 시스템을 수월히 구축
    4. 복잡한 네트워크 시스템을 객체지향적 관점에서 수월히 구축
    5. 운영체제에 독립적인 (크로스 컴파일이 가능한) C++ 라이브러리의 사용

Samchon Framework는 문자 그대로 프레임워크입니다. 또한, 프레임워크는 단독적으로 기능 테스트를 할 수가 없습니다. 이에Samchon Framework를 활용했던 프로젝트 및 예제들과 함께 기능 테스트를 진행하고자 합니다.

즉, 활용 사례들을 통하여 Samchon Framework의 기능 테스트를 대리하고자 합니다. 이 아래로부터는Samchon Framework를 활용했던 프로젝트 및 예제들에 대한 개요들이 적혀 있습니다.

### OraQ

Samchon Framework를 사용하여 구축한 프로젝트로써, PACS 미디어의 입출납을 관리하며 PACS 미디어 기록을 보관하고 열람할 수 있는 클라우드 솔루션입니다.

* **주 기능**
* 회원 및 서버 관리
* Media(CD)로부터 PACS 미디어를 OraQ 서버에 기록 및 관리
* OraQ 서버의 기록의 PACS 및 비표준 파일서버로의 이전을 관리
* PACS Media 뷰어 제공

### Samchon Simulation

Samchon Framework가 사용된 (Samchon Framework의 모태가 된) 프로젝트입니다.

간단한 시세 조회에서부터 남트리를 이용하여 최적의 주식 거래 조건을 도출하고, 몬테카를로 시뮬레이션을 통하여 주가의 흐름 추이를 예측해 볼 수 있는 클라우드 솔루션입니다.

* **주 기능**
* 시세 조회
  + 주가 조회
  + 재무정보 조회
  + 주가 및 재무정보 상호비교
* 종목 검색
* 시뮬레이션
  + 백 테스팅 시뮬레이션
  + 몬테카를로 시뮬레이션
* 부가 기능
  + 문서로의 출력
  + 게시판

### Hansung Timetable

모의 수강신청을 하고, 더불어 해당 시간표를 출력해볼 수 있는 프로그램입니다. 클라우드 시스템의 구축과 관련은 없는 프로젝트이되, Samchon Framework가 사용된 프로젝트이기에, Samchon Framework의 기능테스트 및 유효성 검증을 위해 기입하게 되었습니다.

NamTree를 통하여, 학생들의 입맛에 맞는 복합조건 검색을 할 수 있습니다.

### Examples

앞서의 프로젝트들이 Samchon Framework를 솔루션 단위로써 (주로 클라우드 서비스 단위) 기능 테스트를 대리하기 위하여 쓰였다면, 지금부터 언급되는 예제들은, 각 모듈 단위에 대한 기능 테스트를 대리하기 위해 작성된 예제입니다.

#### Packer

Samchon Framework의 CaseGenerator를 사용한 예제입니다.

M개의 물품을 N 종류의 포장지에 담을 때, 가장 포장지를 적게 쓰는 경우를 찾아줍니다.

#### TSP Solver

Traveling Salesman Problem을 유전자 알고리즘으로 풀어나가는 예제입니다.

#### Entity

Samchon Framework의 표준 데이터 표기법을 위해 설계된, Entity 모듈을 사용하는 예제입니다.

#### Invoke

Samchon Framework의 메시지 프로토콜을 사용하기 위해 제작된 Invoke 모듈, 그에 대한 가이드를 제공하기 위해 작성된 예제입니다.

#### ParallelSystem

Samchon Framework의 분산처리시스템 모듈을 사용하여, 트리 구조의 병렬처리 시스템을 구현한 예제입니다.

## Test 환경정보

### Samchon Framework

* Server
  + OS: C++ 빌드가 가능한 모든 운영체제
  + DBMS: ODBC를 지원하는 DBMS
  + Boost, ODBC 라이브러리[[1]](#footnote-1)
* Client
  + Web Browser
  + JavaScript 또는 Flex를 사용할 수 있는 환경

### OraQ

* Server
  + OS: Windows
  + Framework: Samchon Framework
  + DBMS: SQL Server with File-stream
  + 기타 PACS 미디어 처리에 관련된 라이브러리
* Client
  + Web Browser
  + Flash Player

### Samchon Simulation

* Server
  + OS: Linux/CentOS
  + Framework: Samchon Framework
  + DBMS: MySQL
* Client
  + Web Browser
  + Flash Player

### Hansung Timetable

* Client
  + OS: Adobe-AIR를 설치할 수 있는 운영체제
  + Platform: Adobe-AIR

### Examples

* System
  + OS: C++를 빌드할 수 있는 환경
  + Framework: Samchon Framework

# **SW 기능리스트**

## Samchon Framework

|  |  |  |
| --- | --- | --- |
| **대분류** | **중분류** | **소분류** |
| 라이브러리 | 유틸리티 | 컨테이너 |
| 기타 |
| 임계영역 | |
| 데이터 입출력 | XML Object |
| SQL 드라이버 |
| 이벤트 | |
| 파일트리 | |
| 수학 | 일반 |
| 경우의 수 |
| 유전자 알고리즘 |
| 프로토콜 | 일반 | 인터페이스 |
| Invoke |
| Entity |
| ExternalSystem |
| 서비스 | 클라우드 서비스 |
| SQL 가이드 |
| 마스터 | Distributed System |
| Parallel System |
| 슬레이브 | Distributed System |
| Parallel System |
| 남트리 | 조건 객체 | |
| 파일 객체 | |
| 학습데이터 | |

## OraQ

|  |  |  |
| --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험 항목)** |
| 관리자 | 서버 관리 | 삭제 규칙 설정 |
| 외부 연동서버 정의 |
|
| 회원 및 부서관리 | 부서관리 |
| 회원 권한변경 |
| 회원 목록 및 검색 |
| 회원 로그 | 회원로그 보기 |
| 미디어 임포트 | 미디어 | 미디어 불러오기 |
| 미디어 내용 보기 |
| 미디어 서버에 기록 |
| 환자 정보 | 환자정보 표기 |
| 환자정보 불러오기 |
| 부가기능 | 사진 촬영 |
| 비표준파일 열람 |
| 미디어 리스트 | 미디어 목록 | 스터디 리스트 |
| 비표준파일 리스트 |
| 접근 기록 |
| 미디어 정보 |
|
| 외부 서버에 등록 | PACS 서버에 등록 |
| File 서버에 등록 |
| 미디어 열람 | PACS 미디어 열람 |
| 비표준파일 열람 |
| 미디어 뷰어 | 이미지 보기 | 썸네일 |
| 이미지 뷰어 |
| 부가기능 | 주석 달기 |

## Samchon Simulation

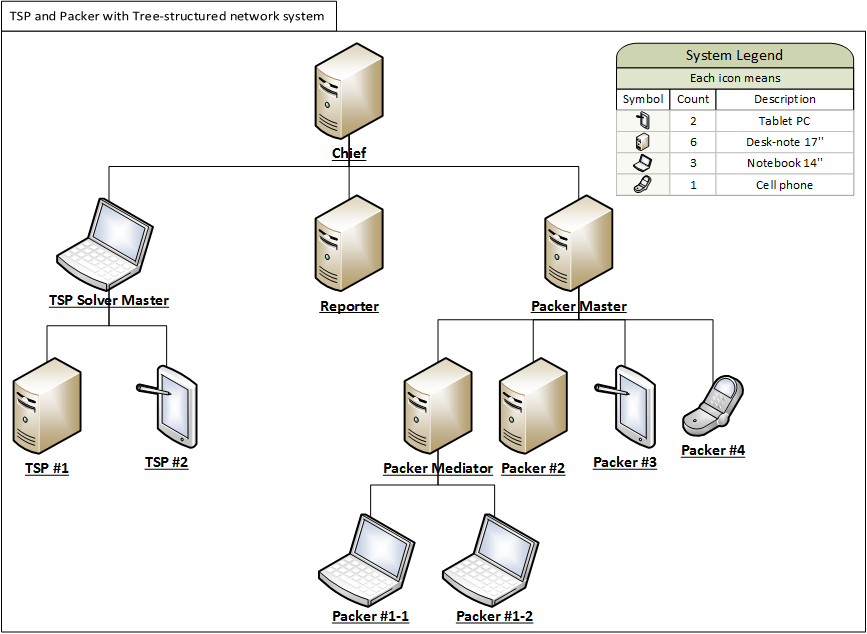
|  |  |  |
| --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험항목)** |
| 공통기능 | 문서화 | 문서로 출력 |
| 그림으로 출력 |
| 기업 목록 | 기업목록 출력 |
| 기업 검색 |
| 기업 선택 |
| 시세 조회 | 주가 조회 | 일일주가 조회 |
| 재무 요약정보 조회 |
| 재무정보 조회 | 투자지표 조회 |
| 재무정보 조회 |
| 비교그래프 |
| 비교정보 조회 | 비교기업 구성 |
| 상대주가 조회 |
| 상대투자지표 조회 |
| 시뮬레이션 (공통) | 설정 정보 | 설정정보 저장 |
| 설정정보 불러오기 |
| 파라미터 창 |
|
| 백 테스팅 (시뮬레이션) | 거래 조건 구성 | 알고리즘 리스트 관리 |
| 매수 알고리즘 구성 |
| 매도 알고리즘 구성 |
| 남트리 파일 관리 |
| 모의 거래 | 시뮬레이션 수행 |
| 데이터 시각화 | 개별 거래결과 조회 |
|
| 시뮬레이션 결과 도표 |
| 수익률 그래프 |
| 몬테카를로 (시뮬레이션) | 파라미터 설정 | 스크립트 입력 |
| 시뮬레이션 | 시뮬레이션 수행 |
| 데이터 시각화 | 시뮬레이션 결과 도표 |
| 개별 분포도 조회 |
|

## Hansung Timetable

|  |  |  |
| --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험 항목)** |
| 시간표 검색 | 조건 객체 | 시간표 검색을 위한 조건식 구성 |
| 파일 객체 | 남트리 파일의 관리 |
| 시간표 구성 | 강의 목록 | 강의목록 파싱 |
| 강의목록 도표에 출력 |
| 시간표 출력 | 모의수강 신청 |
| 프로그램 상에서의 출력 |
| 문서로의 출력 |

## Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류** | **시험 항목** |
| Math | Packer | | 경우의 수를 이용한 포장 분배 |
| TSP Solver | | 유전자 알고리즘을 이용한 TSP 해결 |
| Protocol | Entity | | Entity 모듈 활용 예제 |
| Invoke | | Invoke 모듈 활용 예제 |
|
| Parallel | Master | 병렬처리에 대한 예제 |
| Slave |

병렬처리 예제에 관하여는, 다음 페이지의 네트워크 시스템을 시연  


# **기능 시험항목**

## Samchon Framework

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류** | **이를 사용한 예제** | **Pass & Fail** |
| 라이브러리 | 유틸리티 | 컨테이너 | 프로젝트 전반 |  |
| 기타 |
| 임계영역 | |
| 데이터 입출력 | XML Object | Entity 모듈 |  |
| SQL 드라이버 | DB 입출력이 명시된 경우 |  |
| 이벤트 | | 1) OraQ 타 서버로의 전송 2) 삼촌 시뮬레이션, 진행바 |  |
| 파일트리 | | 1) 삼촌 시뮬레이션 설정파일 2) 남트리 함수파일 일람 |  |
| 수학 | 일반 | 1) 삼촌 시뮬레이션 中 몬테카를로 시뮬레이션 |  |
| 경우의 수 | Examples - Packer |  |
| 유전자 알고리즘 | Examples - TSP |  |
| 프로토콜 | 일반 | 인터페이스 | 프로토콜 전반  (추상 객체 및 인터페이스 |  |
| Invoke |  |
| Entity |  |
| ExternalSystem | Examples -  병렬처리에 대한 예제 |  |
| 서비스 | 클라우드 서비스 | 클라우드 서비스 전반 1) OraQ 2) Samchon Simulation |  |
| SQL 가이드 | OraQ, 로그 기록 |  |
| 마스터 | Distributed System | Examples -  병렬처리에 대한 예제 |  |
| Parallel System |  |
| 슬레이브 | Distributed System |  |
| Parallel System |  |
| 남트리 | 조건 객체 | | Hansung Timetable 1) 시간표 복합 조건검색  Samchon Simulation 1) 종목 검사 2) 백 테스팅 |  |
| 파일 객체 | |  |
| 학습데이터 | |  |

## OraQ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험 항목)** | **사용 모듈** | **Pass/Fail** |
| 관리자 | 서버 관리 | 삭제 규칙 설정 | Entity 및 SQL 모듈 |  |
| 외부 연동서버 정의 |  |
| ExternalSystem 모듈 |
| 회원 및 부서관리 | 부서관리 | Entity 모듈, SQL 모듈 |  |
| 회원 권한변경 |  |
| 회원 목록 및 검색 |  |
| 회원 로그 | 회원로그 보기 |  |
| 미디어 임포트 | 미디어 | 미디어 불러오기 | DLL 라이브러리, Entity 모듈 |  |
| 미디어 내용 보기 |  |
| 미디어 서버에 기록 | 상동 및 SQL 모듈 |  |
| 환자 정보 | 환자정보 표기 | Entity 모듈 |  |
| 부가기능 | 사진 촬영 | 바이너리 송신 |  |
| 비표준파일 열람 | 바이너리 수신 |  |
| 미디어 리스트 | 미디어 목록 | 스터디 리스트 | Entity 및 SQL 모듈 |  |
| 비표준파일 리스트 |  |
| 접근 기록 |  |
| 미디어 정보 |  |
| 바이너리 수신 |
| 외부 서버에 등록 | PACS 서버에 등록 | Event 모듈 및 실시간 연동 |  |
| File 서버에 등록 |
| 미디어 열람 | PACS 미디어 열람 | 해당 없음 (페이지 이동) |  |
| 비표준파일 열람 | Entity 모듈 바이너리 수신 |  |
| 미디어 뷰어 | 이미지 보기 | 썸네일 |  |
| 이미지 뷰어 |  |
| 부가기능 | 주석 달기 | Entity 모듈 |  |

## Samchon Simulation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험항목)** | **사용 모듈** | **Pass/Fail** |
| 공통기능 | 문서화 | 문서로 출력 | Entity 모듈 |  |
| 그림으로 출력 |  |  |
| 기업 목록 | 기업목록 출력 | Entity 모듈 |  |
| 기업 검색 |  |
| 기업 선택 |  |  |
| 시세 조회 | 주가 조회 | 일일주가 조회 | Entity 모듈,  SQL 모듈 |  |
| 재무 요약정보 조회 |  |
| 재무정보 조회 | 투자지표 조회 |  |
| 재무정보 조회 |  |
| 비교그래프 |  |
| 비교정보 조회 | 비교기업 구성 |  |
| 상대주가 조회 |  |
| 상대투자지표 조회 |  |
| 시뮬레이션 (공통) | 설정 정보 | 설정정보 저장 | FileTree 모듈 |  |
| 설정정보 불러오기 |  |
| 파라미터 창 | Application 모듈 |  |
| Entity 모듈 |
| 백 테스팅 (시뮬레이션) | 거래 조건 구성 | 알고리즘 리스트 관리 |  |
| 매수 알고리즘 구성 | NamTree 모듈 |  |
| 매도 알고리즘 구성 |  |
| 남트리 파일 관리 |  |
| 모의 거래 | 시뮬레이션 수행 |  |
| 데이터 시각화 | 개별 거래결과 조회 | Application 모듈 |  |
| Entity 모듈 |
| 시뮬레이션 결과 도표 |  |
| 수익률 그래프 |  |
| 몬테카를로 (시뮬레이션) | 파라미터 설정 | 스크립트 입력 | FileTree 모듈 |  |
| 시뮬레이션 | 시뮬레이션 수행 |  |  |
| 데이터 시각화 | 시뮬레이션 결과 도표 | Entity 모듈 |  |
| 개별 분포도 조회 |  |
| Application 모듈 |

## Hansung Timetable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류 (시험 항목)** | **사용 모듈** | **Pass/Fail** |
| 시간표 검색 | 조건 객체 | 시간표 검색을 위한 조건식 구성 | NamTree |  |
| 파일 객체 | 남트리 파일의 관리 |  |
| 시간표 구성 | 강의 목록 | 강의목록 파싱 | library |  |
| 강의목록 도표에 출력 | Entity |  |
| 시간표 출력 | 모의수강 신청 |  |
| 프로그램 상에서의 출력 |  |
| 문서로의 출력 |  |

## Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **대분류** | **중분류** | **소분류** | **시험 항목** | **사용 모듈** | **Pass/Fail** |
| Math | Packer | | 경우의 수를 이용한 포장 분배 | Case generator |  |
| TSP Solver | | 유전자 알고리즘을 이용한 TSP 해결 | Genetic Algorithm |  |
| Protocol | Entity | | Entity 모듈 활용 예제 | Entity 모듈 |  |
| Invoke | | Invoke 모듈 활용 예제 |  |
| Invoke 모듈 |
| Parallel | Master | 병렬처리에 대한 예제 | ExternalSystem, DistributedSystem 및 ParallelSystem 모듈 |  |
| Slave |

1. Boost, ODBC 라이브러리:

   삼촌 프레임워크를 개작할 때만 필요합니다. 단순히 삼촌 프레임워크를 사용하고자 하는 경우 (DLL이나 SO를 사용하는 경우) 에는 필요하지 않습니다. [↑](#footnote-ref-1)