인수인계서

저작도구 유지보수

namespace: Softpower.SmartMaker.SmartMaker

솔루션 경로: ./TopBuild/TopBuild_80.sln

빌더 실행 오류

빌더 실행 최소 요구사항

- Window 10 64Bit
- Visual C++ 2015 재배포 패키지 설치

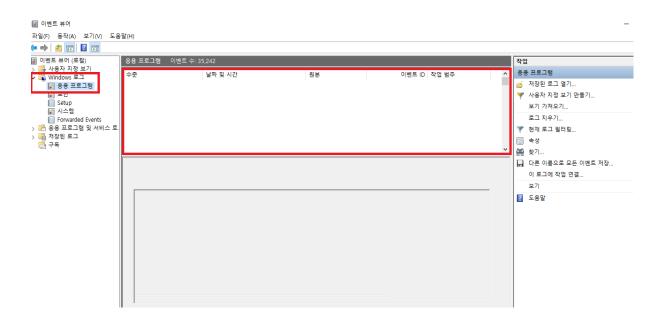
Bos 로그파일

- C:\SmartMaker\SmartBuilder\BOS\log\PQ_PB_YYYY-MM-DD.log 형태의 로그파일 생성 여부 체크 및 내부 내용 확인하여 처리

이벤트 뷰어

- 최소 요구사항 충족 및 Bos로그파일 생성 실패의 경우 윈도우 탐색기 → 이벤트 뷰어를 통해 .NET FrameWork 내부 오류를 의심해봐야 한다.

Windows 로그 → 응용 프로그램 → 에러 로그 확인



에러 로그 확인 및 원인 분석하여 처리하며, 만약 아무런 에러 로그도 없는경우 윈도우 자체에러로 의심되는 상황이면 아래 명령어 혹은 윈도우 복구 기능을 사용하여 시스템 복구이후 다시 테스트 해본다.

CMD

sfc /scannow

아톰 동작 이상

개별 아톰 동작에 버그가 발생한 경우 아톰 아톰 클래스 위치 확인 및 동작 논리를 확인하여 처리한다.

- AtomBase : 아톰 UI 관련 기능

- Atom: 아톰 DB동작 및 업무규칙 관련 기능

- Attrib: 아톰 시리얼라이즈 및 저장속성 관련 기능

편집 동작 이상

아톰 생성 / 이동 / 삭제 등의 편집 기능 수정이 필요한경우 아래 클래스를 참고하여 처리하도록 한다.

DMTFrame / DMTView / DMTDoc : 폼 프레임을 관리하는 부모 클래스들로 이 클래스 내부에서 키보드 / 마우스 이벤트 동작 및 전체적인 아톰 편집기능이 구현되어 있다.

업무규칙 동작 이상

ScriptDoc.cs / IScriptObject / CExeServer 을 상속받은 모든 객체는 업무규칙으로 접근 가능하다.

```
//폼 계열 업무규칙
public class LightJDoc : ScriptDoc {}

//아톰 계열 업무규칙
public class Atom : CObjectX, IScriptObject {}

//전역 업무규칙
public class CFuncServer : CExeServer {}
```

업무규칙은 실행 함수:

- SetProperty : ~~ 설정한다. - GetProperty : ~~을 참조한다.

- Action : ~~을 실행한다.

Atom.ProcessEvent (EVS TYPE.): 업무규칙 이벤트 함수를 호출하는경우

LightJDoc : 폼 객체에 해당하는 업무규칙 동작 (폼의 ...을 실행한다. / 폼의 ...을 설정한다.)

FuncServer: 전역 함수 호출 기능 (난수, 파일 등)

업무규칙

업무규칙 실행 방식

업무규칙은 기본적으로 이벤트를 기준으로 호출되며, 대상(Target)과 동작 방식(EVS_EVENT)이 주어지며 실행되도록 한다.

이벤트 동작 순서

- 1. 전처리 : 진행관리자 전처리 방식으로 설정된 기능이 동작한다.
- 2. 본처리 : 업무규칙 업무규칙에서 작성한 모든 문단은 본처리 단계에서 동작하도록 하다
- 3. 아톰기능 : 아톰 자체적인 기능으로 기능버튼의 저장 / 앞장 / 뒷장 기능이 존재한다.
- 4. 후처리 : 진행관리자 후처리 방식으로 설정된 기능이 동작한다.

각 단계에서 진행중지 / 문단의 실행 취소을 처리하여 반환값이 -1이 처리된경우, 이후 단계를 동작하지 않고 이벤트을 중지하도록 구현되어 있다.

업무규칙 컴파일

스마트메이커 자연어 업무규칙 컴파일 논리는 기본적으로 공백과 조사 (을/를/이/가 등)을 기준으로 처리하도록 되어있다.

컴파일 소스 구조

이 글에서는 유지보수를 위한 일부 클래스만 소개하며, 전체적인 구조 파악을 위해선 별첨되어 있는 과거 소스버전 - 업무규칙 설계도를 확인해야 한다.

[EBLBasePre.cs]

한글 / 일본어 컴파일 단계에서 모든 문장을 처리해주는 주요 클래스로 아래 함수를 통해 실행객체를 생성한다.

업무규칙 실행 객체가 최종적으로 만들어지는 단계로 한 문장에 한번씩 ComplexSentenceParsing 함수가 호출되며 문장 종류(문단선언 / 조건문 / 실행 / 연산)에 따라 내부 클래스로 이동하여 실행객체를 생성한다.

```
public void ComplexSentenceParsing (CEBLList oaExecute,
ProcessQ.DicInfo.GRAM_TYPE nType)
{
    switch (nType)
    {
            //문장 종류에 따라 파싱 및 실행객체 생성
    }
}
```

[WordSplitter_Jpn.cs]

일본어 자연어의 경우 별도 띄어쓰기 없이 작성되기 때문에 한글과 같은 논리로 동작하기 위해 강제로 띄어쓰기를 추가해주는 기능을 담당하는 함수이다.

해당 클래스의 경우 일본어 번역 담당자와 협업하며 구현했으며 사용되는 단어를 임의로 예외처리 했기 때문에 추후 보강이 필요할 수 있음

[CEBLBasePre_Eng]

영문 컴파일 단계에서 모든 문장을 처리해주는 주요 클래스로 아래 함수를 통해 실행객체를 생성한다.

영문버전의 경우 동사 위주로 처리하였으며 자연스러운 문장 구성을 위해 다양한 예외처리 기준이 추가되어 있음 (별첨 영문 엑셀 예제 참고)

문장에서 동사를 찾고, 해당 동사에 따른 개별적 SentenceManager을 통해 업무규칙 컴파일을 진행하도록 설계함

컴파일 단계

1. 문장 패턴화

- 각 문장별로 자연어 사전에 선언되어 있는 패턴과 일치하는 적합도가 가장높은 패턴을 지정하는 단계로 패턴을 탐색하지 못한경우 [문장을 분석할 수 없습니다.] 라는 문구와 함께 컴파일이 실패 처리된다.

2. 단어 변수화

- 사용자가 입력한 아톰명 / 폼명칭 / 변수명 등을 인식하는 단계로 예약어 > 전역변수 > 아톰명 > 변수명 순으로 인식하도록 되어있으며 아톰명 단계에서 인식하지 못한 모든 항목은 변수로 처리된다.

3. 패턴 - 변수 시리얼라이즈

- 변수로 인식된 항목중 아톰에 해당하는 항목과 아톰에 접근하는 속성이 존재할때 해당 아톰이 속성을 지원하지 않는경우 컴파일 에러를 발생시킨다.
- 변수명칭의 속성을 접근하는 경우 컴파일 단계에서는 인식할 수 없지만, 실행 단계에서는 동작 가능한 형태 (부모폼, 아톰변수)등이 존재하기 때문에 경고만 표기하고 컴파일 정상 완료 처리한다.
- 컴파일된 객체는 ArrayList에 담아 Serialize 처리가 가능하도록 관리한다.

4. 컴파일 완료

- 디버깅 모드에서 컴파일이 정상적으로 완료된경우 각 문장을 실행체 단계로 표기하여 콘솔 로그로 출력해준다. 한글 / 일본어 / 영문 업무규칙 컴파일 및 실행체 생성이 정상적으로 되었는지 판단하기 위해 사용되며, 한글 / 일본어 / 영문 각각 컴파일 후 실행결과를 비교하여 판단하는 방식으로 사용한다.
- 다음 예제는 폼_열림 시점에 입력란을 화면에 표시해주는 문단을 컴파일시 표기되는 데이터이다.

```
품을 열면 다음 문단을 실행한다.
{
입력란을 화면에 표시한다.
}
```

EXE / EXE_FUNC : 이벤트 문단의 시작과 끝을 알리는 항목으로 이벤트 문단의 경우이벤트가, 사용자 지정 문단의 경우 사용자 지정 문단명칭이 표시된다.

IM34에서 IM은 폼 이벤트를 뜻하며 34는 _열림의 의미를 가진다.

따라서 폼 열림 문단 내부 내용을 지정해주는 의미이다.

```
SmartMakerEdu.exe Error: 0 : _EXE / EXE_FUNC / IM34 / 1
//생략
SmartMakerEdu.exe Error: 0 : _EXE / EXE_ENDFUNC / IM34 / -1
```

IV200 ~ IV999 와 같이 IV로 시작하는 항목은 사용자가 만든 아톰명칭을 뜻하며 IV200.IA1.IP0의 경우 IV200아톰에 IA1 속성을 사용하겠다는 의미이다. (IP0의 경우 생략되는 경우가 많으며 배열인경우 0번째 항목을 의미함)

```
SmartMakerEdu.exe Error: 0 : _IFO / INFO / #ARGSTART / 3
SmartMakerEdu.exe Error: 0 : _EXE / EXE_EXPR / / 3
SmartMakerEdu.exe Error: 0 : _ITM / ITM_OBJ / IV200.IA1.IP0 /
3
SmartMakerEdu.exe Error: 0 : _IFO / INFO / #ARGEND / 3
```

BOSs2.Key 키워드

업무규칙 이벤트 번호 / 아톰 속성 번호 지정하는 Key파일로 단순 Txt 형태로 이루어져 있음 Key / Value 형태로 구성되며 빌더 실행시 해당 파일에서 데이터를 읽어와 처리한다.

Key: 정수형태로 이루어진 데이터이며, 좌측정렬에 우측 공백으로 처리하도록 한다.

Value: 문자형으로 이루어진 데이터로 공백을 허용하지 않는다.

이벤트 선언

- 폼_열림(): 폼에만 추가 가능한 이벤트 형태이다.
- _실행 () : 이와 같은 행태인경우 _앞에 아톰등의 객체가 추가되는 형태로 모든 아톰에 추가될 수 있는 이벤트 이다.

자연어 패턴 추가 (한글 / 일본어)

자연어 패턴의 경우 한글 / 일본어 모두 동일한 어순을 가지고 있기 때문에 같은 방식으로 컴파일 및 처리하도록 한다.

자연어 패턴:

1. 조사의 종류

```
은/는 : S (주격조사 Subject)
이/가 : H (보격조사 Help)
을/를 : O (목적격조사 Object)
이라는/라는 : D (지칭조사 Designate)
의 : B (by)
에 : N (on)
에서 : I (IN)
로/으로 : W (with)
보다 : T (than)
서술어 : V (Verb)
가짜서술어 : X - 동사형태지만 서술어역활을 하지 않는 동사
파라메터문장 : P (Parameter)
조건문장 : C (Condition)
이벤트 : F (Function Event)
```

커서 : R (CURSOR) -> 프로시저에서 사용

문장 : A

그외 : E (etc)

2. XML 요소의 구성

<ID>: 패턴 고유번호 - 서술어별로 매겨지며 불변이다. 이 고유번호는 팝업메뉴의 우선순위 설정에

사용된다.

<GR TYPE> : 문장의 종류를 판단한다. 대입문인지.. 함수인지.. 반복문인지 등등..

<CHAR> : 문장의 패턴을 나타낸다. 문장성분의 조합, 주어+목적어+서술어 = S + O + V

<RUN> : 기호식 스크립트의 문법을 나타낸다. 사전정보를 만들 때 참고하므로 함부로 바꾸면 안 된다

<COND> : 조건 및 문장에 꼭 필요한 예약어이다 문장성분의 위치와 일치해야 한다.

<EXAMPLE> : 자연어 스크립트 예문이다.

<HELP EX> : 파라메터에 대한 명칭이 붙어있는 예문이다. 추후 기능개선이 필요할 듯 하여 Example과

분리한 상태, Dic에는 HELP EX우선 저장된다.

<SHOW> : FALSE이면 PopupHelp에 보이지 않는다, 지향하지 않는 문법, 여러가지 중복되는 문법 등을

사용자에게 보이지 않기 위해

<RESULT> : 함수가 실행한 후 반환값 (현재 사용 안함)

<KEYWORD> : 편집기에서 F1을 눌렀을 때 Html Help에서 검색할 검색어

3. XML 요소의 해석

tterns/

ID>1</ID> -> '표시하다' 에서 ID 1

<GR TYPE>EXPR</GR TYPE> -> 문장종류 : 연산식

<CHAR>O + N + V</CHAR> -> 문장성분 : 목적어+...에+서술어

<RUN>메시지(OP, VP)</RUN> -> 기호식으로 변경했을 때의 문법

<COND>문장, 화면</COND> -> 목적어의 체언은 '문장', '...에'의 체언은 화면

<EXAMPLE>문장(...)를 화면에 표시한다('예아니오', '질문').</EXAMPLE>

<SHOW>FALSE</SHOW>

</Patterns>

아래 업무규칙 예제가 동작하는 자연어 패턴이다.

최종적으로 만들어지는 실행객체는 [입력란.화면감춤 = 참] 형태로 만들어진다.

자연어 업무규칙 특성상 사용자가 좀더 쉽고 직관적인 문장을 구사해도 동작하도록 처리하는것이 목적이기 때문에 자연어 패턴 적용 문법을 제공한다.

실행객체 : 입력란.화면감춤 = 참

원문: 입력란의 화면감춤은 참을 참조한다.

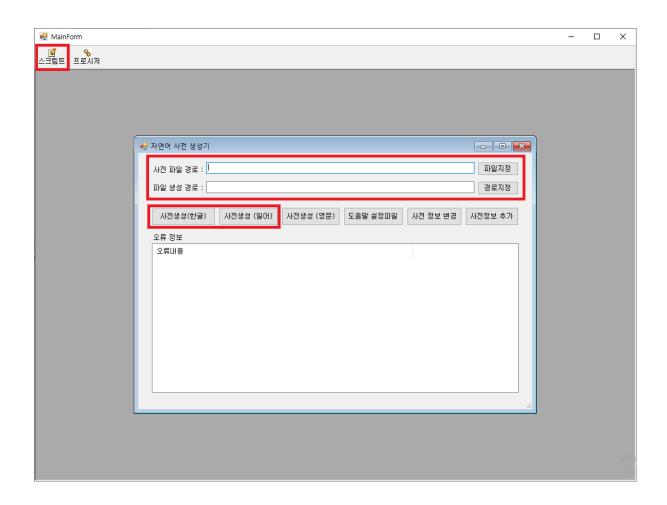
자연어 패턴 적용 : 입력란의 화면감춤 속성을 설정한다.

<Patterns>

<ID>4</ID>
<GR_TYPE>ASSIGN</GR_TYPE>
<CHAR>E + O + V</CHAR>
<RUN>E = 참</RUN>
<COND>, 속성</COND>
<EXAMPLE>... 속성을 설정한다.</EXAMPLE>
<SHOW>FALSE</SHOW>
<KEYWORD>속성설정해제</KEYWORD>
</Patterns>

실행경로:.\TopBuild\DictionaryCreater\bin\Debug\DictionaryCreater.exe

- 1. 실행경로에서 DictionaryCreater.exe파일을 관리자권한으로 실행
- 2. 스크립트 항목 클릭 \rightarrow 사전파일 경로, 생성 경로 지정
- 3. 사전생성 (한글 / 일어) 버튼 클릭
- 4. 생성된 ProcessQ.dic / ProcessQ_J.dic 파일 exe와 동일 경로에 두고 빌더 실행시 적용



DB동작 이상

기본검색키 동작

- 기본검색키란 : 폼 최초 동작시 폼에 설정되어 있는 DB동작들 중 기본으로 동작해야되는 항목을 말하며, 일반적으로 데이터속성 → 기본검색키가 체크되어 있는 항목들이 동작한다.
- 즉시실행: 검색창, 차트와 같이 조회도구 계열에 속하는 아톰들에 있는 옵션으로 별도 동작 없이도 폼 최초 실행시 데이터를 검색해오는 기능을 뜻한다.
- 참조정보키 : 상세폼 보기로 폼을 열었을때 부모폼이 자식폼으로 전달하는 값을 의미한다.

기본검색키 동작이 실행되면 아래와 같은 순서로 동작한다.

- 1. 데이터 입력도구 계열 데이터 초기화
- 2. 동작 Flag에 따라 SELECT / DELETE / UPDATE / INSERT 동작 진행
- 3. [업무규칙] 검색완료 / 저장완료 이벤트 호출

폼 최초 실행시 다음과 같은 순서로 동작한다.

- 1. 참조정보키 설정
- 2. [업무규칙] 폼_준비완료 이벤트 동작
- 3. 기본검색키_조회
- 4. 즉시실행 (검색창, 차트 등)
- 5. [업무규칙] 폼_열림

위 순서는 안드 / IOS / 빌더 / 웹 모두 공통으로 지정된 순서이기 때문에 임의로 변경해서는 안된다.

일반 사용자가 실수하기 쉬운 항목이며, 관련 사항으로 버그가 접수되는 경우 사용자가 위 순서를 고려하여 폼을 제작하였는지 반드시 확인이 필요한다.

예시) 검색창_찾아오기 DB처리객체에 조건을 설정하고, 업무규칙으로 해당값을 지정했는대도 적용되지 않는 상황

검색창 - 즉시실행 설정되어 있음

```
[DB처리객체]
검색창_찾아오기 / WHERE 10 = #조건값입력란 /
[업무규칙]
폼을 열면 다음 문단을 실행한다.
{
조건값입력란은 "10"이다.
}
```

위와 같은 상황일때 검색창 즉시실행의 순서는 **4**이고, 업무규칙 폼 열림의 순서는 **5**이기 때문에 조건값입력란의 값이 ""인 상황에서 조회 동작을 하고, 이후에 입력란에 값을 설정하게 된다.

사용자는 입력란에 값이 들어가 있는대도 불구하고 검색창에서 정상적인 데이터를 가져오지 못한다고 판다하여 에러로 접수할 확률이 매우 높다.

위와 같은 상황에서는 즉시실행 4번 동작 이전에 조건값입력란에 값을 설정해야만 정상적으로 동작함으로 $1 \sim 3$ 번 이내에서 해당값이 설정되어야 한다.

아래 업무규칙은 처음과 다르게 폼_준비완료 시점인 2번 항목에서 설정했기 때문에 정상적으로 동작할 것이다.

```
폼의 준비를 완료하면 다음 문단을 실행한다.
{
조건값입력란은 "10"이다.
}
```

■ DB이외 항목값 유지

데이터 초기화 과정에서 데이터속성이 설정되어 있지 않은 입력란등에 값을 보존하고 싶을때 사용하는 속성이다.

■ 저장후 화면 보존

기본검색키_저장 동작 완료 후 데이터를 초기화 하지 않고 유지해주는 기능으로 연속입력, 연속저장시 사용한다.

DB처리객체

- 자동생성 DB처리객체
 - 별도 호출이 불가능하며, 폼 실행시 자동으로 만들어지는 DB처리객체이다.
 - 데이터입력도구 입력도구 데이터속성 설정시 만들어지는 DB처리객체로 기본검색키로 설정된 항목을 기준으로 자동으로 만들어진다. (버튼 앞장 / 뒷장 / 저장 / 삭제 등으로 호출 가능)
 - 데이터조회도구 검색창 / 스크롤 등에 데이터속성 설정시 만들어지는 DB처리객체로 조건, 정렬 등의 항목 변경이 가능함 (스크롤 다시보기, 검색버튼 등으로 조회 가능)

- 사용자지정 DB처리객체 사용자가 직접 UI을 조작해 만든 DB처리객체로 진행관리자, 업무규칙 등에서 호출 가능하다.

디버깅 방법

모든 DB처리객체 동작시 CDocQueryMgr.cs - MakeQuery 함수를 통해 실행될 쿼리가 만들어지기 때문에 해당 부분에 브레이크를 추가하여 쿼리가 정상적인지 여부 체크 및 디버깅이 가능하다.

```
protected string MakeQuery communic parameter, and adaptation, and Minimage, makes, and Minimage, an
```

DB테이블 자동설계 및 생성

namespace: Softpower.SmartMaker.SD

솔루션 경로: /SD/SD.sin

F10키를 누를경우 현재 폼에 설정되어 있는 데이터속성 정보를 취합해 테이블을 생성해주는 기능이다.

//Softpower.SmartMaker.TopDBManager80.ERDGenManager.ERDSendQueryWindow
//Softpower.SmartMaker.SD.ErdGen.ErdGenWindow.ERDGenWindow
class ERDSendQueryWindow : Windows {}
class ERDGenWindow : UserControl {}

ERDSendQueryWindow : TopBuild 프로젝트에 있는 레퍼 Window로 단순 윈도우 객체역할만 처리해주며 내부에 표시되는 Control은 SD 프로젝트에 있는 ERDGenWindow 을 사용한다.

ERDGenWindow : 실질적인 ERD을 화면에 그려주고, 테이블 생성 동작하는 컨트롤이다. 테이블 생성 버그 및 기능 보강 필요시 해당 클래스를 수정해야 한다.

AWS 서버 - 서버설치

namespace: Softpower.SmartMaker.AwsServer

솔루션 경로: ./AwsServer/AwsServer.sln

AWS에서 제공하는 EC2 기능을 활용하여 스마트서버를 자동으로 설치해주는 기능이다.

AWS의 경우 최초 가입시 1년 무료 사용기간을 제공하기 때문에 그걸 활용한 방식으로 해당 프로젝트 내부에는 AWS SDK을 참조하여, 자동 생성해주는 기능을 담당한다.

//Softpower.SmartMaker.AwsServer.ServerWizard

배포 - 패키지 생성

namespace: Softpower.SmartMaker.AndroidPackage

솔루션 경로: ./Android/Android.sln

```
class PackageMainWindow : Window {
    this.DataContext =
        Softpower.SmartMaker.AndroidPackage.Package.ValueDocument
}
```

패킹 버튼 클릭시 PackageMainWindow 화면이 표시되며, DataContext로 ValueDocument 객체를 소유하고 있다. 패킹동작에서 표시되는 모든 페이지는 동일한 ValueDocument 객체를 공유하여 설정 및 실행시 데이터 일관성 유지하기가 간편하도록 설계됨

로컬 DB변환:

로컬 서버접속 환경에서 만들어진 앱의 경우 현재 접속되어 있는 DB정보를 가져와 SQLite로 컨버전 하는 작업이 진행된다 이때 AndroidSQL 프로젝트를 활용하여 SQLite DB파일을 생성한다.

배포 - 테스트 배포 / 전용 스토어

namespace: Softpower.SmartMaker.AndroidPackage

솔루션 경로: ./Android/Android.sln

각 메뉴 클릭시 UpLoadMainWindow / UpLoadListMainWindow 클래스를 표시해주며, DataContext로 ValueDocument 을 공유하여 모든 동작에 일관된 데이터를 유지하도록 처리함

배포 - 웹서버 배포 (제너레이터)

QPM, QWP \rightarrow HTML로 컨버전 하는 부분의 경우 웹 담당자 분들이 유지보수 진행

웹배포 UI, 파일선택 논리, 디시리얼라이즈 논리에 대해서 변경이 필요한 경우 아래 프로젝트 정보에 따라 수정 필요

- UIX

namespace : WonderfulSoft.mBizGenerator 솔루션 경로 : ./mBizCenter/mBizCenter.sIn

- UIX2

namespace : Softpower.SmartMaker.SmartGenerator 솔루션 경로 : ./SmartGenerator/SmartGenerator.sln

메이커스토어

namespace: MakerStore

솔루션 경로: ./MakerStore/MakerStore.sln

메이커스토어는 QPM으로 구성되어 있는 내부 마켓 프로그램이며 별도 소스코드로 이루어져 있지 않다. 업그레이드 관리 및 내부 정보 유출 방지를 위해 QPM을 리소스 형태로 참조하는 프로젝트를 만들고 이 프로젝트 내부에 모든 폼이 들어가 있는 형태이다.

윈도우 패킹

namespace: mBizSolution

솔루션 경로 : ./mBizSolution/mBizSolution.sln

윈도우 패킹의 경우 사용자가 지정한 모든 QPM파일을 리소스 형태로 참조하는 개별 exe를 생성해주고, 생성된 exe와 연동되는 DLL은 자사가 사용하는 빌더 DLL과 동일하게 일치시켜 동작하도록 하는 기능이다.

디버깅용 프로젝트인 SmartSos내부에 테스트할 exe 로컬 경로를 아래와 같이 지정하고 디버깅하여 처리하도록 한다.

mBizEngine.MainWindow.LastOpenModelPath = @"E:\temp\display_4284060059.exe";

출력물 모델

namespace: Softpower.SmartMaker.SmartMaker

솔루션 경로:./TopBuild/TopBuild 80.sln

프로젝트: TopReportAtom / TopReportLight / TopReportProcess / TopReportPDF

출력물 모델은 기본적으로 전달받은 데이터를 PDF 형태로 출력하는 기능이다.

윈도우 환경에서 PDF 생성을 위해 ITextSharp 라이브러리를 활용했으며, 라이선스 이슈 때문에 라이선스로 부터 자유로운 낮은 버전을 사용하여 기능 구현함

 주의사항: 현재 사용중인 라이브러리에서는 72DPI 기준으로 동작하며, 사용하는 좌표체계가 다르다. (Y축 반전), 관련 부분을 처리하기 위해 컨버전해주는 코드들이 존재한다.

TopReportPDF-PDFDrawManager : 모든 아톰을 PDF양식에 맞춰 그려주는 역할을 하는 클래스이다. PDF 생성이 이상하거나, 기능 보강을 위해선 이 클래스에서 처리해야한다.

출력물 모델도 일반 모델가 동작하는 방법 및 유지보수는 동일하며, 런처 단계에서 동작하는경우 스마트서버에서 PDF을 만들도록 동작함

신규 개발 업무

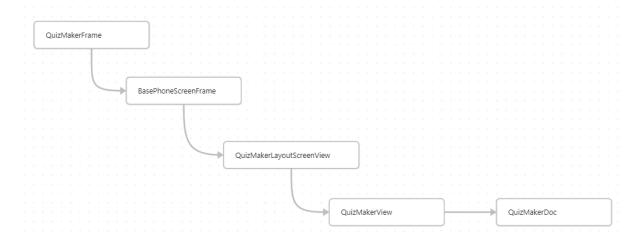
퀴즈메이커

문제은행 서비스를 위해 기획 및 설계되어 있는 기능으로 사용자가 작성한 문제을 **DB**에 저장 및 수정하는 기능이다.

DB에 저장된 정보의 경우 JSON 포맷 형태로 저장되어 있으며 빌더 디시리얼라이즈 기능을 통해 폼 열기가 가능한 구조

퀴즈메이커 내부 항목의 경우 아톰들로 이루어져 있으며 정답 출제 동작시 퀴즈블록을 동적으로 생성 / 내부 아톰 전체를 퀴즈블록에 포함하여 동작처리하는 것으로 문제풀이 기능을 제공한다.

관련 클래스



상속 구조

- QuizMakerFrame : DMTFrame

- QuizMakerLayoutScreenView: BaseScreenView

QuizMakerView : DMTViewQuizMakerDoc : DMTDOC

동작 방식

- 퀴즈메이커 자체적으로 제공해야되는 편집기능들을 위해 기존 **DMT** 계열 클래스를 상속받아 편집기능 영역만 추가함
- 실제 실행 동작의 경우 기존 DMTDoc / DMTFrame과 동일한 아톰 기반 동작 논리를 통해 동작되도록 설계

서버

빌더 동작 방식

빌더에서 에듀태크 관련 서버와 통신해야되는 기능들의 경우

Softpower.SmartMaker.TopApp.EduTech.EduTechServletManager 해당 클래스에 전부 선언되어 있다.

EduTechServletManager.Instance.GetQuizMakerQuizList() 등과 같이 사용한다

API 스펙

QuizMaker.do

문제은행 컨텐츠 데이터 및 정답 데이터 자체가 중요하다고 판단되어 기본적으로 파라미터 암호화 처리 하여 동작하도록 기능 설계됨

공통 암호화 방식

key : QuizMaker.do?action=getKey value : Dictionary<string, string>

사용 함수 : EduTechServletManager.Instance.GetEncryptParam(Dictionary<string, string>map, string key)

[GET] action = getKey

현재 접속중인 세션을 기준으로 암호화 키값 발급받는다

필수여부	예시	설명

[결과값]

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "2804393e135c4ed6",
    "RESULT_MSG": "SUCCESS"
}
```

[GET] action = curriculumlist

db에 저장된 문제유형에 전체 정보를 받아온다. 상세 정보의 경우 list을 통해 가져온다

필수여부	예시	설명

```
{
    "RESULT_COLUMN_NAME": [
        "Q03"
],
    "RESULT_COLUMN_TYPE": [
```

[GET] action = list

로그인이 유지된 상태에서만 동작하는 기능으로, db에 저장된 문제정보를 조회한다.

	필수 여부	예시	설명
param	Y	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다 - curriculum : 필수 : DB Q03에 해당하는 값으로 교과목 기준으로 조회한다 - index : 선택 : 블록검색에 사용되는 값으로 기본값 0으로 동작하며 100행씩 조회하도록 처리되어 있음 - level : 선택 : DB Q06에 해당하는 값으로 문제 난이도별 조회 동작시 사용한다. - type : 선택 : DB Q05에 해당되는 값으로 문제유형 기준 조회 동작시 사용한다.

```
{
    "RESULT_COLUMN_NAME": [
        "Q01",
        "Q02",
        "Q05",
```

[GET] action = insert

신규 문제은행 정보를 추가하는 기능으로 퀴즈코드가 자동으로 발급된다

	필수 여부	예시	설명
param	Y	[암호화값]	테이블 스팩에 따라 Q02 ~ Q14 / QC2 ~ QC13 까지 항목들을 전달해주며 공통 암호화 값으로 처리 - contentCount : 일관형등 content가 여러개 생성되어야 되는 경우 사용

필 수 여 부	예시	설명
丁		

RESULT	Υ	2412000001	빌더 제너레이터 등에서 사용 가능하도록 문제번호 리턴처리
_DATA			

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "2412000001",
    "RESULT_MSG": "SUCCESS"
}
```

[GET] action = update

관리번호 기준 문제를 수정 처리하도록 한다

	필수 여부	예시	설명
param	Υ	[암 호화 값]	테이블 스팩에 따라 Q02 ~ Q14 / QC2 ~ QC13 까지 항목들을 전달해주며 공통 암호화 값으로 처리

[결과값]

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "",
    "RESULT_MSG": "SUCCESS"
}
```

[GET] action = updateQuizContent

관리번호 기준 문제은행 컨텐츠를 수정하는 기능으로 / Insert 이후 웹배포 Html, js, 문항, 답항 값등을 별도로 저장하기 위해 사용한다

	필수 여부	예시	설명
param	Y	[암 호화 값]	QC1 / QC3 / QC11 / QC12 / QC13 - 각 항목에 해당하는 데이터를 공통 암호화 값으로 처리

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "",
```

```
"RESULT_MSG": "SUCCESS"
}
```

[GET] action = delete

관리번호 기준 문제를 삭제하기 위해 존재하는 기능

	필수 명부	예시	설명
param	Y	암 항 장 값	Q01 값 암호화 하여 전달

[결과값]

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "",
    "RESULT_MSG": "SUCCESS"
}
```

[GET] action = select

현재 접속중인 세션을 기준으로 암호화 키값 발급받는다

	필수 여부	예시	설명
param	Y	[암 호화 값]	Q01 값 암호화 하여 전달

```
{
    "RESULT_COLUMN_NAME": [
        "Q01",
        "Q02",
        "Q03",
        "Q04",
        "Q05",
        "Q06",
        "Q07",
        "Q08",
        "Q09",
        "Q10",
        "Q11",
```

```
"Q12",
"CHAR",
```

[GET] action = selectqcontent 웹 배포 이후 퀴즈를 **HTML**로 표시하기 위해 사용

- 추후 권한체크 논리 보강 필요할것으로 예상됨

	필수 여부	예시	설명
--	----------	----	----

QC1	Υ	2024120 001	문제번호	
-----	---	----------------	------	--

[결과값]

[GET] action = deployqqm

현재 접속중인 세션을 기준으로 암호화 키값 발급받는다

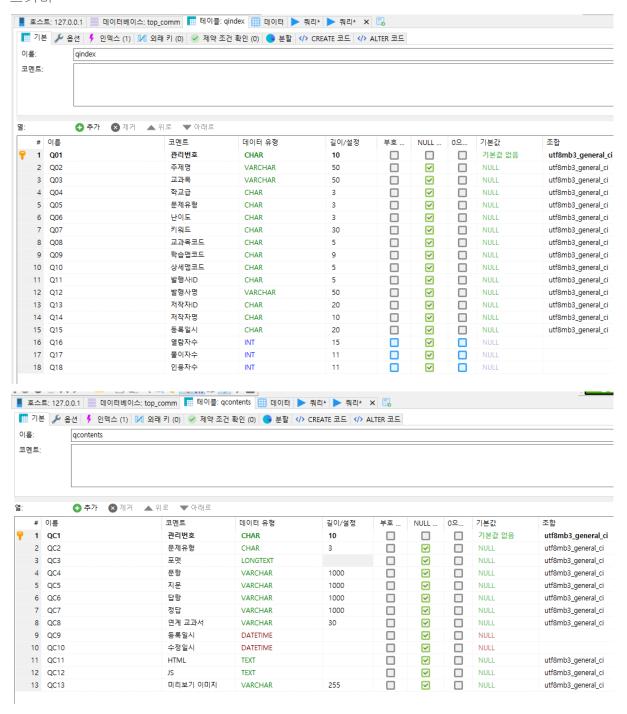
	필수 여부	예시	설명
param	Υ	[암호화값]	path 값 암호화 하여 전달
FILE	Y	Multi part 로 파일 데이터 전달	업로드할 파일 전달

[결과값]

[GET] action = getQuizCode 신규 퀴즈 코드를 발급받는 기능

```
"RESULT_CODE": "1000",
    "RESULT_DATA": "2412000001",
    "RESULT_MSG": "SUCCESS"
}
```

스키마



```
-- 호스트:
                    127.0.0.1
-- 서버 버전:
                     10.6.8-MariaDB - mariadb.org binary distribution
-- 서버 OS:
                     Win64
-- HeidiSQL 버전:
                       12.0.0.6468
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET NAMES utf8 */:
/*!50503 SET NAMES utf8mb4 */;
/*!40103 SET @OLD TIME ZONE=@@TIME ZONE */;
/*!40103 SET TIME ZONE='+00:00' */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN KEY CHECKS=0 */;
/*!40101 SET @OLD SQL MODE=@@SQL MODE,
SQL MODE='NO AUTO VALUE ON ZERO' */;
/*!40111 SET @OLD SQL NOTES=@@SQL NOTES, SQL NOTES=0 */;
-- 테이블 top_comm.qindex 구조 내보내기
CREATE TABLE IF NOT EXISTS 'gindex' (
 `Q01` char(10) NOT NULL COMMENT '관리번호'.
 `Q02` varchar(50) DEFAULT NULL COMMENT '주제명',
 `Q03` varchar(50) DEFAULT NULL COMMENT '교과목',
 `Q04` char(3) DEFAULT NULL COMMENT '학교급'.
 `Q05` char(3) DEFAULT NULL COMMENT '문제유형',
 `Q06` char(3) DEFAULT NULL COMMENT '난이도',
 `Q07` char(30) DEFAULT NULL COMMENT '키워드'
 `Q08` char(5) DEFAULT NULL COMMENT '교과목코드'
 `Q09` char(9) DEFAULT NULL COMMENT '학습맵코드'
 `Q10` char(5) DEFAULT NULL COMMENT '상세맵코드'.
 `Q11` char(5) DEFAULT NULL COMMENT '발행사ID',
 `Q12` varchar(50) DEFAULT NULL COMMENT '발행사명'.
 `Q13` char(20) DEFAULT NULL COMMENT '저작자ID',
 `Q14` char(10) DEFAULT NULL COMMENT '저작자명',
 `Q15` char(20) DEFAULT NULL COMMENT '등록일시',
 `Q16` int(15) DEFAULT NULL COMMENT '열람자수',
 `Q17` int(11) DEFAULT NULL COMMENT '풀이자수'
 `Q18` int(11) DEFAULT NULL COMMENT '인용자수'.
 `Q19` char(5) DEFAULT NULL COMMENT '콘텐츠평점',
 PRIMARY KEY ('Q01')
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
-- 내보낼 데이터가 선택되어 있지 않습니다.
/*!40103 SET TIME ZONE=IFNULL(@OLD TIME ZONE, 'system') */;
/*!40101 SET SQL MODE=IFNULL(@OLD SQL MODE, ") */;
/*!40014 SET FOREIGN_KEY_CHECKS=IFNULL(@OLD_FOREIGN_KEY_CHECKS, 1) */;
/*!40101 SET CHARACTER SET CLIENT=@OLD CHARACTER SET CLIENT */;
/*!40111 SET SQL NOTES=IFNULL(@OLD SQL NOTES, 1) */;
```

- -----

```
127.0.0.1
-- 서버 버전:
                      10.6.8-MariaDB - mariadb.org binary distribution
-- 서버 OS:
                     Win64
-- HeidiSQL 버전:
                       12.0.0.6468
/*!40101 SET @OLD CHARACTER SET CLIENT=@@CHARACTER SET CLIENT */;
/*!40101 SET NAMES utf8 */:
/*!50503 SET NAMES utf8mb4 */:
/*!40103 SET @OLD TIME ZONE=@@TIME ZONE */;
/*!40103 SET TIME ZONE='+00:00' */;
/*!40014 SET @OLD FOREIGN KEY CHECKS=@@FOREIGN KEY CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD SQL MODE=@@SQL MODE,
SQL MODE='NO AUTO VALUE ON ZERO' */;
/*!40111 SET @OLD SQL NOTES=@@SQL NOTES, SQL NOTES=0 */;
-- 테이블 top comm.qcontents 구조 내보내기
CREATE TABLE IF NOT EXISTS 'gcontents' (
 `QC1` char(10) NOT NULL COMMENT '관리번호',
 `Q_1` int(11) NOT NULL COMMENT '관리번호2',
 `QC2` char(3) DEFAULT NULL COMMENT '문제유형',
 `QC3` longtext DEFAULT NULL COMMENT '포맷',
 `QC4` varchar(1000) DEFAULT NULL COMMENT '문항'
 `QC5` varchar(1000) DEFAULT NULL COMMENT '지문'
 `QC6` varchar(1000) DEFAULT NULL COMMENT '답항',
 `QC7` varchar(1000) DEFAULT NULL COMMENT '정답',
 `QC8` varchar(30) DEFAULT NULL COMMENT '연계 교과서',
 `QC9` datetime DEFAULT NULL COMMENT '등록일시',
 `QC10` datetime DEFAULT NULL COMMENT '수정일시',
 'QC11' text DEFAULT NULL COMMENT 'HTML',
 'QC12' text DEFAULT NULL COMMENT 'JS'.
 `QC13` varchar(255) DEFAULT NULL COMMENT '미리보기 이미지',
 PRIMARY KEY ('QC1', 'Q 1') USING BTREE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
-- 내보낼 데이터가 선택되어 있지 않습니다.
/*!40103 SET TIME ZONE=IFNULL(@OLD TIME ZONE, 'system') */;
/*!40101 SET SQL_MODE=IFNULL(@OLD_SQL_MODE, ") */;
/*!40014 SET FOREIGN KEY CHECKS=IFNULL(@OLD FOREIGN KEY CHECKS, 1) */;
/*!40101 SET CHARACTER SET CLIENT=@OLD CHARACTER SET CLIENT */;
/*!40111 SET SQL NOTES=IFNULL(@OLD SQL NOTES, 1) */;
```

부문명칭	Top_comm						
테이블 명칭	qindex						
필드명	자료형	필드 길이	기본갑	명칭	비고		
Q01	char	10	Х	관리번호			
Q02	varchar	50	NULL	주제명			
Q03	varchar	50	NULL	교과목			
Q04	char	3	NULL	학교급			
Q05	char	3	NULL	문제유형			
Q06	char	3	NULL	난이도			
Q07	char	30	NULL	키워드			
Q08	char	5	NULL	교과목코드			
Q09	char	9	NULL	학습맵코드			
Q10	char	5	NULL	상세맵코드			
Q11	char	5	NULL	발생사ID			
Q12	varchar	50	NULL	발행사명			
Q13	char	20	NULL	저작자DI			
Q14	char	10	NULL	저작자명			
Q15	char	20	NULL	등록일시			
Q16	int	10	NULL	열람자수			
Q17	int	10	NULL	풀이자수			
Q18	int	10	NULL	인용자수			
Q19	char	5	NULL	콘텐츠평점			

부문 명칭	Top_Comm						
테이블 명칭	qcontents						
필드명	자료형	필드 길이	기본갑	명칭	비고		
QC1	char	10	х	관리번호			
Q_1	int		х	관리번호2			
QC2	char	3	NULL	문제유형			
QC3	longtext	x	NULL	포맷			
QC4	varchar	1000	NULL	문항			
QC5	varchar	1000	NULL	지문			
QC6	varchar	1000	NULL	답항			
QC7	varchar	1000	NULL	정답			
QC8	varchar	30	NULL	연계 교과서			
QC9	datetime	х	NULL	등록일시			
QC10	datetime	Х	NULL	수정일시			

QC12 text				
JQC12 LEXI	Х	NULL	JS	
QC13 varchar	255	NULL	미리보기 썸네일	

퀴즈메이커 CMS

프로젝트 정보

- 작업 브랜치 : Develop_QuizMaker

- 명칭 : SmartCMS

- Namespace : Softpower.SmartMaker.SmartCMS

퀴즈메이커 CMS 클래스 구성도



시험지 생성

시험지 제작, 웹고시 생성, 강의안 생성, 앱웹 서비스 기능



- 1. Index 윈도우에서 DB에 저장될 기초 정보 설정 (ExaminationInfo)
- 2. CMSWindow에서 실제 출제될 문제 종류 설정
- 3. 설정된 최종 데이터를 기반으로 시험지 생성 기능 구현

산출물 관리 기능

생성된 고사 정보를 화면에 표시해주고, 세부 정보 확인 가능하도록 처리해주는 기능 상세정보창에서 고사 선택 후 시험지 생성 버튼을 통해 QEB로 변환 작업을 처리하도록 한다.



시험지 생성 기능 (QEB 변환 기능) 생성된 고사 정보를 바탕으로 QEB을 생성하는 기능 원본 문제 크기와, 용지 크기가 일치한다는 보장이 없기 때문에 아래 조건에 따라 문제 크기를 임의로 변경하고, 사용자가 아톰 크기를 다시 편집하는 방식으로 설계되어 있음

1열 배치

- 컨텐츠가 QEB 중앙에 배치될 수 있도록 좌표 조정
- 원본 문제 크기 그대로 유지

2열 배치

- 컨텐츠 표시 영역 (QEB 넓이 / 2 여백)에 원본크기을 맞춰 전체적인 비율을 계산하여 재배치하도록 기능 구현
- 2열 배치, 영역 넘어가는 경우 다음 열 / 다음 페이지 로 배치되도록 처리

[TopProcess] CMSGeneratorManager.cs

위치 : Softpower.SmartMaker.TopProcess.CMSGenerator.CMSGeneratorManager

ExecuteGenerator() 함수를 통해 시험지 정보를 QEB로 변환해주는 기능

DB

스키마

름: !멘트:	qexam								
	추가 ⊗ 제	거 ▲ 위로 ▼ 아래로							
#	이름	코멘트	데이터 유형	길이/설정	부호	NULL	00	기본값	조합
- 1	Qe01	고사번호	CHAR	6				기본값 없음	utf8mb3_general_c
2	Qe02	고사제목	CHAR	40		~		NULL	utf8mb3_general_ci
3	Qe03	대상자	CHAR	30		~		NULL	utf8mb3_general_ci
4	Qe04	처리유형	CHAR	14		~		NULL	utf8mb3_general_ci
5	Qe05	목적	CHAR	60		~		NULL	utf8mb3_general_ci
6	Qe06	출제자 ID	CHAR	10		~		NULL	utf8mb3_general_ci
7	Qe07	출제자명	CHAR	20		~		NULL	utf8mb3_general_ci
8	Qe08	교과목	CHAR	5		~		NULL	utf8mb3_general_ci
9	Qe09	학교학년	CHAR	3		~		NULL	utf8mb3_general_ci
10	Qe010	문항수	CHAR	4		~		NULL	utf8mb3_general_ci
11	Qe011	총점	INT	4		~		NULL	
12	Qe012	문제구성	TEXT			~		NULL	utf8mb3_general_ci
13	Qe013	정답열	VARCHAR	256		~		NULL	utf8mb3_general_ci
14	Qe014	매체속성	CHAR	12		~		NULL	utf8mb3_general_ci
15	Qe015	풀이속성	CHAR	12		~		NULL	utf8mb3_general_ci
16	Qe016	채점속성	CHAR	12		~		NULL	utf8mb3_general_ci
17	Qe017	고시일자	CHAR	8		\checkmark		NULL	utf8mb3_general_ci
18	Qe018	출제일시	CHAR	15		~		NULL	utf8mb3_general_ci
19	Qe019	사용기록	VARCHAR	255		\overline{V}		NULL	utf8mb3 general ci

생성 쿼리

```
127.0.0.1
-- 서버 버전:
                     10.6.8-MariaDB - mariadb.org binary distribution
-- 서버 OS:
                    Win64
-- HeidiSQL 버전:
                       12.0.0.6468
/*!40101 SET @OLD CHARACTER SET CLIENT=@@CHARACTER SET CLIENT */;
/*!40101 SET NAMES utf8 */:
/*!50503 SET NAMES utf8mb4 */:
/*!40103 SET @OLD TIME ZONE=@@TIME ZONE */;
/*!40103 SET TIME ZONE='+00:00' */;
/*!40014 SET @OLD FOREIGN KEY CHECKS=@@FOREIGN KEY CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD SQL MODE=@@SQL MODE,
SQL MODE='NO AUTO VALUE ON ZERO' */;
/*!40111 SET @OLD SQL NOTES=@@SQL NOTES, SQL NOTES=0 */;
-- 테이블 top comm.qexam 구조 내보내기
CREATE TABLE IF NOT EXISTS 'gexam' (
 `Qe01` char(6) NOT NULL COMMENT '고사번호'.
 `Qe02` char(40) DEFAULT NULL COMMENT '고사제목',
 `Qe03` char(30) DEFAULT NULL COMMENT '대상자',
 `Qe04` char(14) DEFAULT NULL COMMENT '처리유형',
 `Qe05` char(60) DEFAULT NULL COMMENT '목적',
 `Qe06` char(10) DEFAULT NULL COMMENT '출제자 ID',
 `Qe07` char(20) DEFAULT NULL COMMENT '출제자명'.
 `Qe08` char(5) DEFAULT NULL COMMENT '교과목',
 `Qe09` char(3) DEFAULT NULL COMMENT '학교학년'.
 `Qe010` char(4) DEFAULT NULL COMMENT '문항수',
 `Qe011` int(4) DEFAULT NULL COMMENT '총점',
 `Qe012` text DEFAULT NULL COMMENT '문제구성',
 `Qe013` varchar(256) DEFAULT NULL COMMENT '정답열'.
 `Qe014` char(12) DEFAULT NULL COMMENT '매체속성',
 `Qe015` char(12) DEFAULT NULL COMMENT '풀이속성'.
 `Qe016` char(12) DEFAULT NULL COMMENT '채점속성',
 `Qe017` char(8) DEFAULT NULL COMMENT '고시일자',
 `Qe018` char(15) DEFAULT NULL COMMENT '출제일시'.
 `Qe019` varchar(255) DEFAULT NULL COMMENT '사용기록',
 PRIMARY KEY ('Qe01')
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3;
-- 내보낼 데이터가 선택되어 있지 않습니다.
/*!40103 SET TIME_ZONE=IFNULL(@OLD_TIME_ZONE, 'system') */;
/*!40101 SET SQL MODE=IFNULL(@OLD SQL MODE, ") */;
/*!40014 SET FOREIGN KEY CHECKS=IFNULL(@OLD FOREIGN KEY CHECKS, 1)
/*!40101 SET CHARACTER SET CLIENT=@OLD CHARACTER SET CLIENT */;
/*!40111 SET SQL NOTES=IFNULL(@OLD SQL NOTES, 1) */;
```

테이블 스펙

부문명칭	Top_comm					
테이블 명칭	qexam					
필드명	자료형	필드 길이	기본갑	명칭	비고	
QE01	char	6		고사번호		
QE02	char	40	NULL	고사제목		
QE03	char	30	NULL	대상자		
QE04	char	14	NULL	처리유형		
QE05	char	60	NULL	목적		
QE06	char	10	NULL	출제자ID		
QE07	char	20	NULL	출제자명		
QE08	char	5	NULL	교과목		
QE09	char	3	NULL	학교학년		
QE10	char	4	NULL	문항수		
QE11	int	10	NULL	총점		
QE12	text	65535	NULL	문제구성		
QE13	text	65535	NULL	정답열		
QE14	char	12	NULL	매체속성		
QE15	char	12	NULL	풀이속성		
QE16	char	12	NULL	채점속성		
QE17	char	8	NULL	고사일자		
QE18	char	15	NULL	출제일시		
QE19	varchar	255	NULL	사용기록		

API 스펙

/QuizMakerCMS.do

CMS (Contetns Management System)을 위해 제공되는 기능으로 퀴즈메이커 조회, qexam 신규 항목 추가 기능 등을 제공해준다.

[GET] action = getKey

현재 접속중인 세션을 기준으로 암호화 키값 발급받는다

필수여부	예시	설명

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "2804393e135c4ed6",
    "RESULT_MSG": "SUCCESS"
}
```

조회

[GET] action = list 퀴즈 정보 조회 기능

	필수여부	예시	설명
param	Υ	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다 누락된 항목이 존재하는 경우 정상적으로 처리되지 않기 때문에 반드시 모든 항목값을 전달해야되며, "으로 전달시 조건을 비교하지 않도록 서버측에서 처리함 - curriculumCode - schoolGradeCode - quizCode - quizLevelCode - studyMapCode - writerCode - dateTimeCode - keyWord1 ~ keyWord5

항목 상세 설명

각 항목들은 공백으로 전달시 비교하지 않도록 서버쪽에서 처리되어 있음

- curriculumCode : Q03 - 교과목 - schoolGradeCode : Q04 - 학교급

- quizCode : Q05 - 문제유형 - quizLevelCode : Q06 - 난이도 - studyMapCode : Q09 - 학습맵코드 - writerCode : Q13 - 저작자ID

- dateTimeCode : Q15 - 등록일시

D1 : 최근 1개월D2 : 최근 6개월D3 : 최근 1년

keyWord1 ~ keyWord5 - Q02, Q07 Like로 검색하며 하나의 조건만 충족되더라도
 전체 검색되도록 처리

[서버측 실행 쿼리]

```
SELECT Q01, Q02, Q03, Q04, Q05, Q06, Q09, Q12, Q15, Q18 FROM QINDEX WHERE

(" = ? OR Q03 = ?) AND

(" = ? OR Q04 = ?) AND

(" = ? OR Q05 = ?) AND

(" = ? OR Q06 = ?) AND

(" = ? OR Q09 = ?) AND

(" = ? OR Q13 = ?) AND

(" = ? AND " = ? AND " = ? AND " = ? AND " = ?) OR

(" != ? AND (Q07 LIKE ? OR Q02 LIKE ?)) OR

(" != ? AND (Q07 LIKE ? OR Q02 LIKE ?)) OR

(" != ? AND (Q07 LIKE ? OR Q02 LIKE ?)) OR

(" != ? AND (Q07 LIKE ? OR Q02 LIKE ?)) OR

(" != ? AND (Q07 LIKE ? OR Q02 LIKE ?))

)
```

```
{
    "RESULT_COLUMN_NAME": [
        "Q01",
        "Q02",
        "Q03",
        "Q04",
        "Q05",
        "Q06",
        "Q09",
        "Q12",
        "Q15",
        "Q18"
    ],
    "RESULT_COLUMN_TYPE": [
        "CHAR",
        "VARCHAR",
        "VARCHAR",
        "CHAR",
        "CHAR",
```

[GET] action = select_quiz 퀴즈 정보 정답 조회 기능

	필수여부	예시	설명
param	Υ	[암호화 값]	code : 정답을 조회할 퀴즈 코드를 , 를 통해 복수 조회 가능하도록 처리

[서버측 실행 쿼리]

```
SELECT QC7 FROM QCONTENTS WHERE QC1 = ? OR QC1 = ?
```

```
{
    "RESULT_COLUMN_NAME": [
    "QC7"
```

```
"RESULT COLUMN TYPE": [
    "RESULT DATA": [
"[{\"Values\":[{\"AnswerValueType\":0,\"GroupName\":\"\",\"AtomName\":\
"답항1\",\"AnswerValue\":true},{\"AnswerValueType\":0,\"GroupName\":\"\"
\"AtomName\":\"답항2\",\"AnswerValue\":true},{\"AnswerValueType\":0,\"G
roupName\":\"\",\"AtomName\":\"답항3\",\"AnswerValue\":true},{\"AnswerVa
lueType\":0,\"GroupName\":\"\",\"AtomName\":\"답항4\",\"AnswerValue\":fa
lse},{\"AnswerValueType\":0,\"GroupName\":\"\",\"AtomName\":\"답항5\",\"
AnswerValue\":false}],\"Name\":\"\",\"Point\":0.0,\"QuizType\":1,\"Quiz
AnswerEngineType\":0,\"ActionType\":0,\"IsDirection\":false}]"
"[{\"Values\":[{\"AnswerValueType\":0,\"GroupName\":\"\",\"AtomName\":\
"답항1\",\"AnswerValue\":true},{\"AnswerValueType\":0,\"GroupName\":\"\"
.\"AtomName\":\"답항2\",\"AnswerValue\":false},{\"AnswerValueType\":0,\"
GroupName\":\"\",\"AtomName\":\"답항3\",\"AnswerValue\":false},{\"Answer
ValueType\":0,\"GroupName\":\"\",\"AtomName\":\"답항4\",\"AnswerValue\":
false},{\"AnswerValueType\":0,\"GroupName\":\"\",\"AtomName\":\"답항5\",
."AnswerValue\":false}],\"Name\":\"\",\"Point\":0.0,\"QuizType\":1,\"Qu
   "RESULT CODE": "1000",
    "RESULT MSG": "SUCCESS"
```

[GET] action = select_quiz_content 퀴즈 컨텐츠정보 조회 기능

	필수여부	예시	설명
param	Y	[암호화 값]	code : 정답을 조회할 퀴즈 코드를 [,] 를 통해 복수 조회 가능하도록 처리

```
SELECT QC1, Q_1, QC2, QC3, QC4, QC5, QC6 FROM QCONTENTS WHERE QC1 = ? OR QC1 = ?
```

```
"RESULT COLUMN NAME": [
    "RESULT COLUMN TYPE": [
       "LONGTEXT",
       "VARCHAR",
       "VARCHAR"
    "RESULT DATA": [
"H4sIAAAAAAAEAO1dW28jyXX+KwPCDw5W3u1udpPiAAZCiuJt1E2Jd3I1GPRNzaaa3dy+iJ
eJqH3YAAHWDzHqTYzAa2ycGI6NGHCcNbB5yJ/J4478H3Kq+k5S0nBHmtWCZUM7rFNVp06d0
nVOVX1V5OtM2ZK9qWq6neVMzTynDjJNW9d0UzTCjLZq66IheFNJtTPPM5mDjDLvqbajW2bm
OXOQqZbrvAY5r9GnE91xM88/fp1pSrw4G+CPL9QlyvUbyNRNN8sAk55oeJCmr6OPURGazmU
S1FPVPLIMC5r4yWGW4/I0NAq0suiM2+7S8KUGQ19X3DFw/JC6vr4+uKddaqd26Vw+n2fo3F
s0/BI1/W69Z5NSBCIEEnAHmZroylH737mn29r4Cb3JfafuMFub4rY2FWm0YpkuyhHEKSr/5
nc/v/ny02ff/vdXb37xBdRs6yukqMMPQdcJue7t9nbVfj+ybB+Ch5CFodZ12WnAtquWTwpW
n4qa6s97/PFURLae+Tt1H41V+bKvSpnnF6LhqLhxaD5TLvGi46rYJxRtW1z6koT8qZTtbSu
Q0kzdYRz3xBKVoJmDTNF1bV1KiJxPVVDmYQEKlbWmzZko6y6oqaYCyqltzVS7JyKH9uY/v7
n56ldv/vwPf/31N8CmpV4kc4HShIToBjrgLUU1giFBKnF6uqNLSPuBcHWnrDsiUEBe1/Ywp
SeaujNOFCl6rhUXi8h+wyXXPF6ocoLeddSers403d",
"[{\"Name\":\"문제번호\",\"ValueType\":1,\"Value\":\"[N]\"},{\"Name\":\"
지시문\",\"ValueType\":1,\"Value\":\"글을 알맞게 띄어 읽을 때 좋은 점을 모르
고르세요.\"}]",
"[{\"Name\":\"답항1\",\"ValueType\":1,\"Value\":\"①\"},{\"Name\":\"답항2
\",\"ValueType\":1,\"Value\":\"②\"},{\"Name\":\"답항3\",\"ValueType\":1
\"Value\":\"③\"},{\"Name\":\"답항4\",\"ValueType\":1,\"Value\":\"④\"}
{\"Name\":\"답항5\",\"ValueType\":1,\"Value\":\"⑤\"}]"
```

```
],
],
"RESULT_CODE": "1000",
"RESULT_MSG": "SUCCESS"
}
```

[GET] action = select_quiz_answer 퀴즈 정보 정답 조회 기능

	필수여부	예시	설명
param	Y	[암호화 값]	code : 정답을 조회할 퀴즈 코드를 , 를 통해 복수 조회 가능하도록 처리

[서버측 실행 쿼리]

SELECT QC7 FROM QCONTENTS WHERE QC1 = ? OR QC1 = ?

[GET] action = select 고사 정보 조회 기능

	필수여부	예시	설명
param	Y	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다 누락된 항목이 존재하는 경우 정상적으로 처리되지 않기 때문에 반드시 모든 항목값을 전달해야되며, "으로 전달시 조건을 비교하지 않도록 서버측에서 처리함 - title - examinationTarget - studyMap - schoolGrade - dateTimeCode

항목 상세 설명

각 항목들은 공백으로 전달시 비교하지 않도록 서버쪽에서 처리되어 있음

- title : 고시 제목 - QE02

- examinationTarget : 고사 대상 - QE04

- studyMapCode : 교과목 - QE08 - schoolGrade : 학년 - QE09 - dateTimeCode : Q15 - 등록일시

D1 : 최근 1개월D2 : 최근 6개월D3 : 최근 1년

```
SELECT QE01, QE02, QE03, QE08, QE09, QE06, QE07, QE18 FROM QEXAM WHERE

QE02 LIKE ? AND

(" = ? OR QE04 = ?) AND

(" = ? OR QE08 = ?) AND

(" = ? OR QE09 = ?)
```

```
"고사제목",
"대상자",
"교과목",
```

생성

[POST] action = insert

시험지 생성 기능

	필수여부	예시	설명
param	Y	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다
			- QE02 ~ QE19 테이블 스펙에 맞춰 적절한값 설정

[서버측 실행 쿼리]

PK조회

```
SELECT MAX(QE01) FROM QEXAM
```

PK Insert 처리 (실패시 반복해서 처리하여 최대 30번 시도)

INSERT INTO QEXAM (Qe01) VALUES (?);

PK Insert 성공한 값으로 Update 처리

```
UPDATE QEXAM SET

QE02 = ?,
QE03 = ?,
QE04 = ?,
QE05 = ?,
QE06 = ?,
QE07 = ?,
QE08 = ?,
```

```
QE09 = ?,
QE010 = ?,
QE011 = ?,
QE012 = ?,
QE013 = ?,
QE014 = ?,
QE015 = ?,
QE016 = ?,
QE017 = ?,
QE018 = ?,
QE019 = ?,
WHERE QE01 = ?;
```

RESULT_DATA로 퀴즈코드를 리턴해준다.

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "000001",
    "RESULT_MSG": "SUCCESS"
}
```

수정

[POST] action = update

시험지 수정 기능

	필수여부	예시	설명
param	Y	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다
			- QE01 ~ QE19 테이블 스펙에 맞춰 적절한값 설정

[서버측 실행 쿼리]

```
UPDATE QEXAM SET

QE02 = ?,
QE03 = ?,
QE04 = ?,
QE05 = ?,
QE06 = ?,
QE07 = ?,
```

```
QE08 = ?,
QE09 = ?,
QE10 = ?,
QE11 = ?,
QE12 = ?,
QE13 = ?,
QE14 = ?,
QE15 = ?,
QE16 = ?,
QE17 = ?,
QE19 = ?
WHERE QE01 = ?;
```

RESULT_DATA로 퀴즈코드를 리턴해준다.

```
{
    "RESULT_CODE": "1000",
    "RESULT_DATA": "000001",
    "RESULT_MSG": "SUCCESS"
}
```

삭제

[POST] action = delete

시험지 삭제 기능

	필수여부	예시	설명
param	Υ	[암호화 값]	아래 항목들을 포함한 Map을 공통 암호화 방식에 따라 암호화 처리하여 전달한다 - QE01

[서버측 실행 쿼리]

```
DELETE FROM QEXAM WHERE QE01 = ?
```

[결과값]

RESULT_DATA로 퀴즈코드를 리턴해준다.

```
{
    "RESULT_CODE": "1000",
```

```
"RESULT_DATA": "000001",
"RESULT_MSG": "SUCCESS"
```

미공개 프로젝트

스마트메이커 형상관리기

name space: Softpower. Smart Maker. Project Manager Process

솔루션 경로 : ./ProjectManagerProcess/ProjectManagerProcess.sln

프로젝트 : ProjectManagerProcess

QPM, QWP, 리소스 등의 정보를 관리하기 위해 추가된 툴입니다.

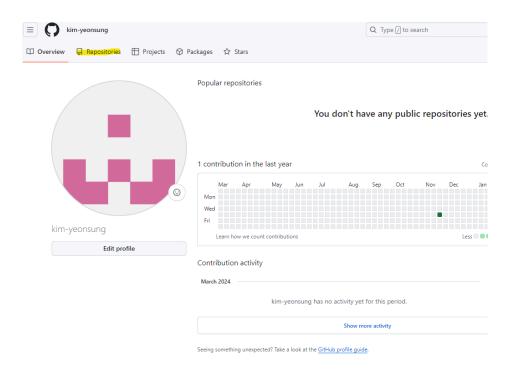
스마트메이커 형상관리 툴의 경우 원칙적으로 동시 편집을 제공하지 않습니다. 만약 임의로 파일변경 등을 통해 여러 사용자가 같은 파일을 수정한경우 업로드 / 다운로드가 정상적으로 동작하지 않거나 편집한 내용이 삭제될 수 있기 때문에 주의가 필요합니다

서버 : GitHub, Git, GitLab 모두 사용 가능

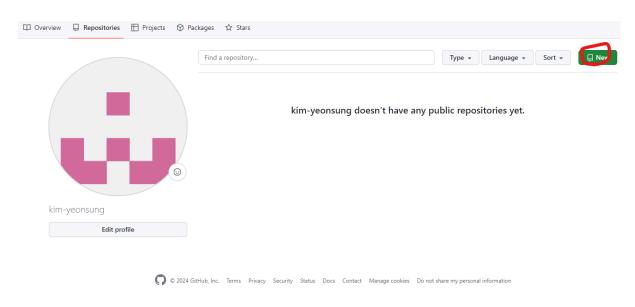
클라이언트: WPF, LibGit2Sharp 라이브러리 사용

GitHub를 통한 Repositories 생성

1. GitHub로그인 이후 Repositories 선택



2.New 버튼을 통해 Repositories 생성



3.공개 / 비공개 여부 선택 및 Repositories 생성 완료

Repository name : 저장소 명칭

Description : 설명

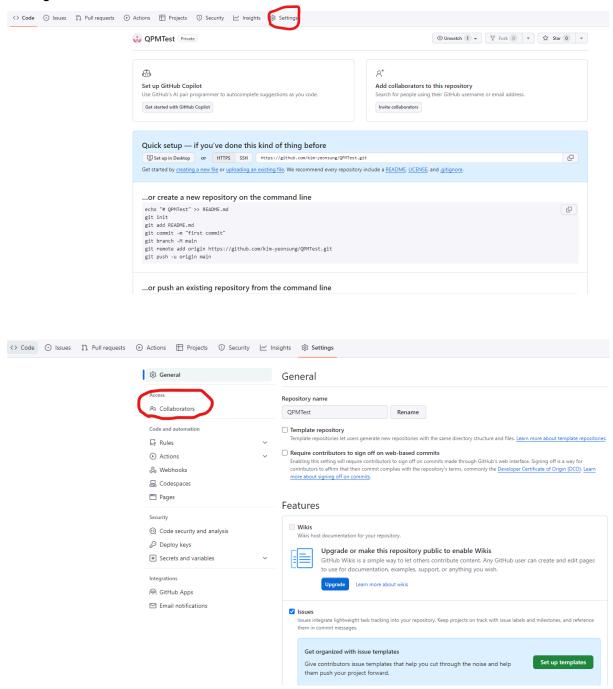
public : 공개 (GitHub사이트를 통해 조회 및 사용 가능)

private : 비공개 (GitHub사이트를 통한 조회 불가, 특정 사용자만 사용 가능)

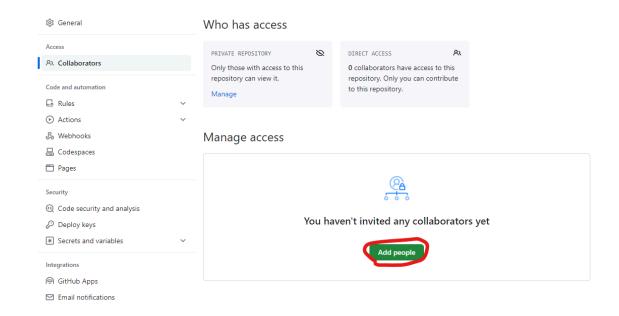
Owner	*	
		Repository name *
	kim-yeonsung 🔻	/
		▲ New repository name must not be blank
Great	epository names are	short and memorable. Need inspiration? How about $\mbox{miniature-octo-carnival}$?
Descri	ption (optional)	
-	Public	
0 1		rnet can see this repository. You choose who can commit.
	∩ Private	
\circ	_	n see and commit to this repository.
Initiali	ze this repository wi	th:
	d a README file	
Thi	s is where you can write	a long description for your project. <u>Learn more about READMEs.</u>
Add .g	itignore	
	itignore nore template: None 🔻	
.gitigi	nore template: None 🔻	from a list of templates. <u>Learn more about ignoring files.</u>
.gitigi Choose	nore template: None 🔻	from a list of templates. <u>Learn more about ignoring files.</u>
.gitigi Choose Choos	which files not to track to	from a list of templates. <u>Learn more about ignoring files.</u>

- 4. Repositories 생성이후 접근 가능한 사용자 설정 및 토큰값 발급
- 4-1 프로젝트 접근 사용자 설정

Setting -> Access -> Collaborators



Add people를 통해 프로젝트 편집 사용자를 지정한다



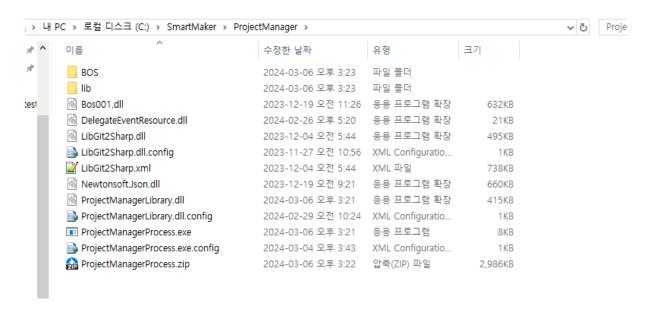
GitHub 토큰 발급 및 사용 방법

git ID, PW의 경우 민감한 정보이기 때문에 형상관리 툴에서 직접 입력하지 않고 Token을 사용하는 방식을 권장한다. 아래 링크를 통해 개발자 각자 토큰값을 발급받고 발급받은 토큰값을 형상관리 툴에 등록하여 사용하는 방식을 권장한다

GitHub토큰 발급 방법: https://hoohaha.tistory.com/37

형상관리툴 사용 방법

기본 설치 경로의 경우 : C:\SmartMaker\ProjectManager 경로에 생성된다. (파일 목록의 경우 달라질 수 있음)



시작 프로그램: ProjectManagerProcess.exe

환경설정 파일 위치: %AppData%\SmartMaker\Git\ConfigDefine.json

(예시 : C:\Users\[사용자 계정]\AppData\Roaming\SmartMaker\Git\ConfigDefine.json)

프로그램 시작시 환경설정 파일이 없는경우 환경설정 입력 화면으로 이동하며 기본적인 GitHub 계정정보를 입력한다

5 프로젝트 관리	×
프로젝트 URL :	https://github.com/kim-yeonsung/QPMTest
로컬 폴더 경로 :	C:\github
로그인 방식 :	○ ID방식 ● Token방식
ID:	
PW:	
Token : ghp_r4	1y2pXaj7Z2NUQ5odKD5ADqceXoWx3nLhkF
	확인

프로젝트 URL : Git 저장소 경로를 입력한다. ex) <u>https://github.com/kim-yeonsung/QPMTest</u> 로컬 폴더 경로 : 로컬에 저장될 저장소 경로를 설정한다. ex) C:\GitHub

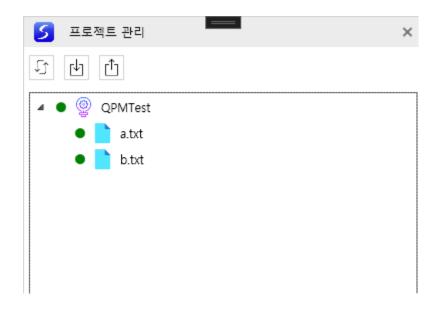
로그인 방식: ID방식, Token방식중 하나를 선택한다. ID방식의 경우 ID, PW가 GitHub에 등록된 계정과 정확히 일치해야하며, Token방식의 경우 토큰값만 일치하면 된다.

토큰 방식이더라도 작업중 표시를 위해 ID는 필수적으로 입력해야한다.

설정 완료후 확인버튼 클릭시 다음과 같은 경로에 작업이 진행된다.

[로컬폴더 경로] \ [프로젝트명칭] -> C:\GitHub\QPMTest

경로에 폴더 생성 및 정보 다운로드 (Clone처리)가 진행되기 때문에 최초 1회는 해당 경로가 비어있어야한다.



파일 더블클릭시 실행되며, QPM 등과 같은 모델 확장자는 열려있는 스마트메이커와 연동되어 동작한다.

폼 변경시 색상으로 표시되며 각 색상은 다음과 같은 의미를 가진다

빨간색 - 삭제

노란색 - 편집

파란색 - 신규 추가

파일 탐색기에서 이동, 추가, 삭제 처리한 내용도 감지할 수 있다.

버튼 종류: 새로고침 / 다운로드 / 업로드 순

새로고침 버튼 : 의 경우 로컬에 반영된 사항이 UI에 표시되지 않을때 누르는 버튼이다.

다운로드 : 다른 사용자가 서버에 반영한 내용을 받기 위한 버튼으로 중복 수정된 내용이 있는경우 오류가 발생할 수도 있다.

업로드 : 로컬에서 편집한 내용을 서버에 반영하는 버튼으로 클릭시 별도 창으로 이동하여 수정내역 작성 및 업로드를 진행할 수 있다.

수전 내용 반영 창은 다음과 같으며 체크박스 선택 / 해제를 통해 서버에 반영될 내용을 결정할 수 있다.

5 수정 내용 반영	×
□ • b.txt	
변경 내용을 입력하세요	
변경 내용 반영(Commit)	

내부 인프라 관련

서버 트래픽 감지

namespace: SmartServerTrafficManager

솔루션 경로: ./SmartServerTrafficManager.sln

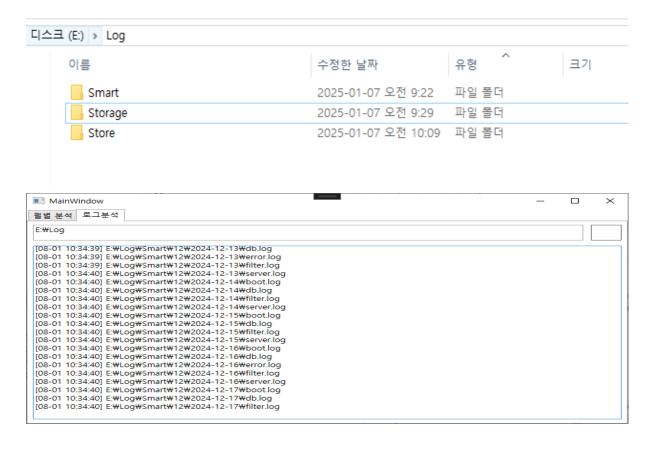
프로젝트 : SmartServerTrafficManager

스마트서버 로그을 인식하여 각 서블릿당 몇회 호출되었는지 측정해주는 프로그램 입니다.

로그분석

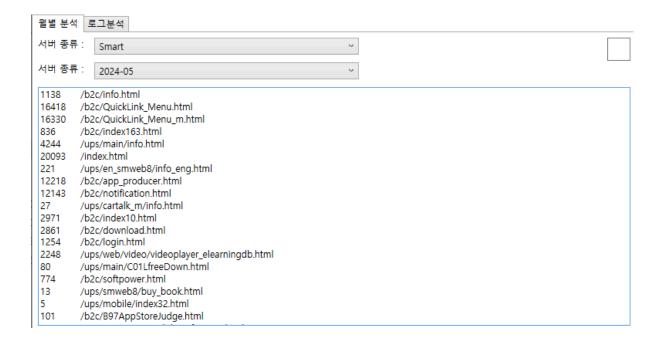
각 서버별 폴더 생성 후 해당 폴더에 로그파일을 넣고 우측에 있는 버튼 클릭시 자동으로 로그을 인식해줍니다.

Smart : 홈페이지 서버Storage : 스토리지 서버Store : 메이커스토어 서버



월별 분석

분석할 서버 및 일자를 선택 후 우측 버튼을 눌러 로그파일 분석 실행



분석 데이터 일부

```
20093 /index.html
16418 /b2c/QuickLink_Menu.html
16330 /b2c/QuickLink_Menu_m.html
12218 /b2c/app_producer.html
12143 /b2c/notification.html
4244 /ups/main/info.html
2971 /b2c/index10.html
2861 /b2c/download.html
2571 /b2c/C01LfreeDown.html
2248 /ups/web/video/videoplayer_elearningdb.html
1610 /b2c/B11eduEnterApp.html
...
```

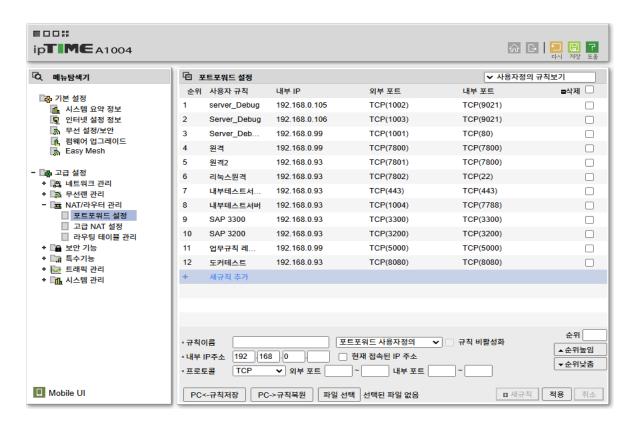
트래픽별 최종 합계와 월별 트래픽 분석하여 엑셀 시트로 정리



포트 포워딩

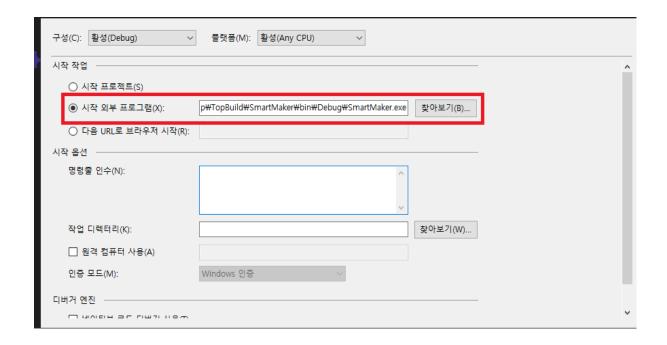
포트 포워딩을 통해 리뷰시 원격 / 서버 디버깅등에 유용하게 활용 가능함.

포트포워딩 기초 개념 (IP주소와 포트포워딩이란, 포트포워딩 설정하기(iptime))



참고 자료

외부 EXE 디버깅 방법



시작 외부 프로그램으로 SmartMaker.exe | SmartMakerEdu.exe 설정, 디버깅할 프로젝트의 DLL 및 PDB 파일을 exe경로에 추가하고 디버깅 실행시 정상적으로 디버깅 가능

WPF MVVM 디자인 패턴

https://wookshin.github.io/2023/08/28/wpf-mvvm-binding.html

GIT 브랜치 관리 전략 (Git - Flow)

https://puleugo.tistory.com/107

내부 공유 문서

- 개발환경 셋팅
 - 개발환경 세팅(빌더) Git
 - 개발환경 구축 (서버) GIT
- 스마트서버 Edutech Basic 라이선스 SmartServer Edutech Basic 라이선스 정책 (안)
 - 에듀테크 베이직 버전 관리자 도구 사용방법 가이드
- 스마트메이커 솔루션:
 - 스마트메이커 솔루션
- 푸시 동작 방식

 https://iridescent-dish-501.notion.site/1682e4aadf4a80c28ea7ed89a0b777a2?pvs=7
- 스마트서버 윈도우 서비스 등록
 https://iridescent-dish-501.notion.site/1772e4aadf4a80c7b9a2d6c921d14397?pvs=74
- 리눅스 마리아 DB 수동 설치 https://iridescent-dish-501.notion.site/DB-1842e4aadf4a806ab4cbc063e735f828
- Nginx 설정 (window)

 https://iridescent-dish-501.notion.site/Nginx-window-1842e4aadf4a8098b7baea11c8e
 2737d?pvs=74