

E2E 성적 관리 시스템

1. 시스템 소개

가. 본 시스템은 E2E 어학원의 성적 관리 시스템이다

나. 기본적인 UI 는 winform을 이용하여 작성하였다.

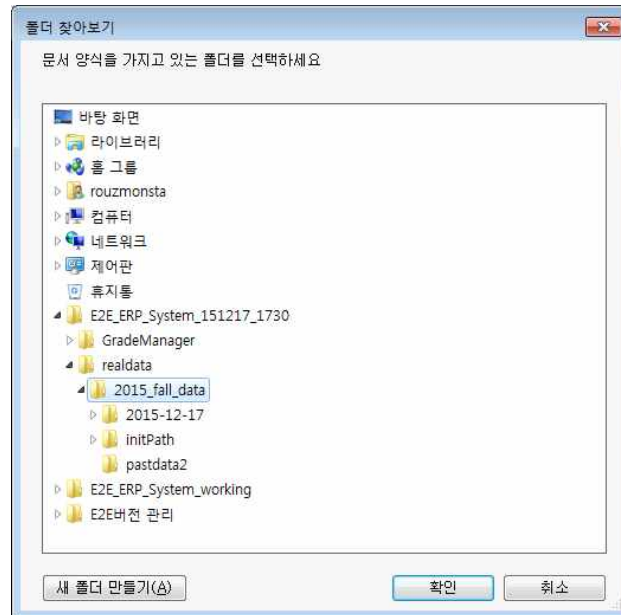
다. 개발 환경 소개

- 1) C# .Net Framework 4.1 및 winform을 이용하여 작성
- 2) Windows 7 Ultimate K
- 3) Visual Studio 2012

2. 시스템 외부의 전반적인 흐름

가. 시스템 최초 실행 시, raw data가 있는 폴더를 선택할 수 있는 창이 출력된다.

1) 화면 캡처



나. 폴더 선택 후 확인 버튼을 클릭하면, 리포트 생성을 위한 화면이 출력된다.

다. 전체적인 구성은 크게 6개의 탭으로 나뉘어져 있다.



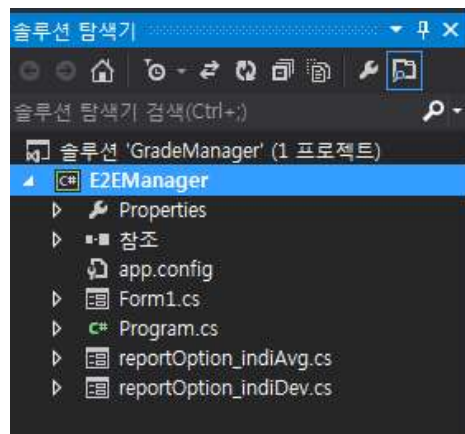
라. 탭의 제목과 기능은 아래와 같다.

- 1) ClassReport - Story
 - 가) Story class에 대한 Class report를 출력하는 탭
 - 2) StudentReport - Story
 - 가) Story class에 대한 Student report를 출력하는 탭
 - 3) ClassReport - Step
 - 가) Step class에 대한 Class report를 출력하는 탭
 - 4) StudentReport - Step
 - 가) Step class에 대한 Student report를 출력하는 탭
 - 5) ClassReport - IBT
 - 가) IBT class에 대한 Class report를 출력하는 탭
 - 6) StudentReport - IBT
 - 가) IBT class에 대한 Student report를 출력하는 탭
 - 7) 기타 기능
 - 가) 학생 무결성 체크와 성적 입력용 시트 생성을 위한 탭
- 마. 시스템 사용 시 주의 사항은 아래와 같다
- 1) 새로운 학기 시작 시 초기 데이터 구축은 기존 데이터를 copy & paste 하는 방식으로 우선 구축한다.
 - 2) raw data로 이용되는 파일명은 최종 class명과 동일해야 한다.
 - 3) raw data 내부의 시트 포맷은 기존의 형식 및 열 순서를 유지하도록 한다.
 - 4) 최초 성적 입력 파일 생성 및 새로운 학생 추가 시, 무결성 체크 기능을 이용하여 서로간의 정보에 문제가 없는지 확인한다.

3. 시스템 전반적 흐름(내부)

가. 전체적인 시스템 분기에 대한 설명

- 1) 전반적인 Form은 크게 3가지로 나뉜다



가) Form1.cs

- (1) 해당 시스템의 메인 form으로, 총 7개의 탭으로 구성되어 있다.

나) reportOption_indiAvg.cs

- (1) 평균에 대한 학생 개인 리포트 출력 시, 필요한 값을 입력하는 form이다.

다) reportOption_indiDev.cs

(1) 편차에 대한 학생 개인 리포트 출력 시, 필요한 값을 입력하는 form이다.

나. 전반적인 winform call에 대한 설명

- 1) 본 프로그램은 엑셀 응용 프로그램과 winform을 연결한 프로그램이다. 따라서 프로그램 구동 시에는 엑셀 응용 프로그램에 대하여 실행 상태를 유지하여야 한다. 따라서 winform 초기화 시 excel application에 대한 object를 실행 후 invisible 상태로 둔다.
- 2) 리포트 출력을 위한 데이터 입력은 form에 명시된 번호 순으로 한다.
- 3) report 생성 버튼 클릭 시, 실제 리포트 생성을 위한 루틴으로 접근한다.
- 4) 먼저, 리포트 생성에 필요한 데이터의 무결성을 체크한다.
 - 가) 정확한 항목에 체크가 이루어졌는지 확인
 - 나) 입력된 숫자 범위가 정확한지 확인
 - 다) 필요한 데이터가 모두 입력되었는지 확인
- 5) 무결성 체크가 완료되면, 출력 대상에 대한 정보를 리스트 자료 구조를 이용하여 작성한다.
- 6) class report와 student report에서 출력 대상에 대한 정보는 콤보박스를 이용하여 입력받는다.
- 7) 최종적으로 콤보 박스를 통하여 입력된 level, class, (name) 을 level list, class list, (name list) 로 구성한다. level과 class, list의 매칭 문제는 동일한 인덱스로 접근하는 것으로 해결하였다.
 - 가) 최초 원품 구축 시에 dictionary 구조들을 이용하여 level과 class, name code(동명이인 처리를 위한 학생 코드), name 간 연결 구조를 구축한다.
 - 나) 구축된 dictionary 구조를 이용하여 list 구축 시에 활용하였다.
 - 다) '전체'를 대상으로 리포트 출력 시, dictionary 구조를 이용하여 모든 데이터들을 추출, list 구축에 활용하였다.
- 8) list를 이용하여 리포트 대상 구축을 완료 후, report format file의 시트를 이용하여 리포트별 시트를 추출하여 복사 후 사용한다.
 - 가) 리포트 포맷 파일은 대상 폴더 내의 ReportFormat.xlsx에 각 시트로 구분되어 있다.
- 9) 시트 복사 후, 리포트 대상들과 각 시트명으로 구성된 raw data 엑셀 파일을 이용하여 리포트에 필요한 결과들을 계산한다.
- 10) 계산된 결과는 각각의 리포트에 맞게 셀에 데이터 입력 후, reformatting 작업을 수행한다. reformatting 작업 시에는 리포트의 종류 별로 서식이 다르기 때문에 모두 다른 방식으로 포맷을 적용하였다.
- 11) 리포트 완성 후에는 '작업 완료'라는 메시지 창을 띄운다. 리포트 생성 중 오류 발생 시에는 오류 메시지를 출력한다.
- 12) 리포트 완성 후에는 사용하고 있는 Excel worksheet object와 excel workbook object에 대해서 저장 및 memory release를 해준다. 이는 ExcelDispose라는 커스텀 함수 혹은 직접적인 releaseObject 함수 콜을 이용하여 접근하였다.
- 13) 리포트 완성 리스트의 파일 이름을 더블클릭하면 해당 리포트 창을 연다.

다. 내부적으로 쓰이는 자료구조 및 대표 함수에 대한 설명

- 1) 평가 항목
 - 가) 기본 평가 항목은 Extensive, Intensive, Spoken 총 3개로 구성되어 있다.

나) 이는 최초 Step class에 대한 시스템 구축 시 backbone을 구성한 것이며, IBT class와 Story class에서는 다르게 적용된다.

다) 즉, 내부적 명칭은 Extensive, Intensive, Spoken으로 사용하지만, 실제 매핑을 통하여 Story와 IBT class에서는 다른 이름으로 사용된다.(내부 코드 참고)

2) finalData 자료구조

가) 최종 리포트 출력을 위한 자료 및 계산 결과를 담기 위한 custom class

나) 최종 리포트 성적은 finalTest.xlsx에 반별로 점수가 입력되어 있다.

다) 항목은 article1, article2, article3, article4로 총 4개로 구성되어 있다.

라) 클래스에 따라 finalData가 없는 경우도 존재한다.

마) finalData class 내부에 article별 세부 항목 매핑 정보가 기술되어 있다.(코드 참고)

3) classData 자료구조

가) 일반적으로 리포트에 사용되는 모든 계산 결과를 저장할 수 있는 통합 자료구조이다.

(1) 평가 항목별 평균

(2) 세부 평가항목별 평균

(3) 평가 항목별 편차

(4) 세부 평가항목별 편차

(5) 전체 항목 수

(6) 평가 항목의 통합 평균

(7) 그 외

나) 클래스가 대상으로 지정될 경우, 클래스 전체에 대한 결과를 저장하며, 학생이 대상으로 지정될 경우, 해당 학생의 결과를 저장하게 된다.

(1) calculateClassResult라는 함수를 통하여 접근

4. 시스템 사용을 위한 기본 환경 설정

가. 본 시스템은 Excel application을 연결하여 사용하는 시스템이다. 따라서 응용프로그램간 연결을 위해서 microsoft access database를 설치하여 사용하여야 한다

1) 해당 시스템에서 사용한 라이브러리는 Microsoft Excel 2010을 기준으로 작성하였다. 따라서 Microsoft Access Database 2010을 설치하여 사용한다.

2) 다른 버전의 Office 이용 시, 상하위 호환에 문제가 생길 시, 프로젝트 참조에서 참조된 라이브러리 확인이 필요하다.

3) .Net Framework 4.X 기준으로 작성되었기 때문에, 최소 .Net Framework 4.1 이상이 설치되어 있어야 한다.

4) x86, x64 모두 동작할 수 있도록 구동 대상을 any cpu로 설정하였다. XP 환경에서 구동 시에는 any cpu가 아닌, x86, x64 중 하나의 모드를 선택하여 컴파일 후 사용하도록 한다.

5. 작성자

가. 이름 : 강 상 욱

나. 이메일: muzholiq@gmail.com

다. 최종 수정일: 2015.12.26