

객체지향설계프로젝트

1조

최종 레포트

팀원 : 강승민, 박원준

## Diary for E.S

1. 개발 동기
2. 제공할 기능
3. 요구사항리스트
4. 유즈케이스 다이어그램
5. 세부 유즈케이스
6. 시퀀스 다이어그램
7. 화면정의서
8. 프로그램 구동 화면
9. 클래스 다이어그램
10. API 매뉴얼(Javadoc 사용) 초기페이지 스크린샷
11. 팀원 간 작업 분배 계획 및 결과
12. 자가 평가

## 개발 동기

- 지금까지 손으로 작성하던 일기 작성 방법이 너무 오래되었다고 생각되어 새로운 일기 작성 방법을 고안
- 학생과 선생님 모두에게 지금의 시스템보다 더욱 더 편리한 기능을 제공하여 학생에게는 일기 외의 공부에 대한 시간과 효율을 높이고 선생님 역시 업무 전반의 효율을 높이기 위함

## 제공할 기능

### 학생

1. 일기 작성
2. 일기에 사진 첨부
3. 실시간으로 일기 제출
4. 이전 일기 보기
5. 알림장 기능

### 선생님

1. 실시간으로 일기 검사
2. 좀 더 편리한 일기 검사 시스템
3. 알림장 기능
4. 학생의 계정 관리

### 서버

1. 모든 학생의 모든 일기를 보관 (사진 포함)
2. 일기의 수정사항 반영
3. 계정 관리

## 요구사항 리스트 (최종 수정 안)

### 학생

- 1) 아이디와 비밀번호를 입력하여 로그인 할 수 있어야 한다.
- 2) 일기를 작성하고 일기에 이미지를 첨부할 수 있어야 한다.
- 3) 검사한 내용을 확인할 수 있어야 한다.
- 4) 날짜를 선택하여 이전 일기를 열람할 수 있어야 한다.
- 5) 선생님이 작성한 공지사항을 확인할 수 있는 기능

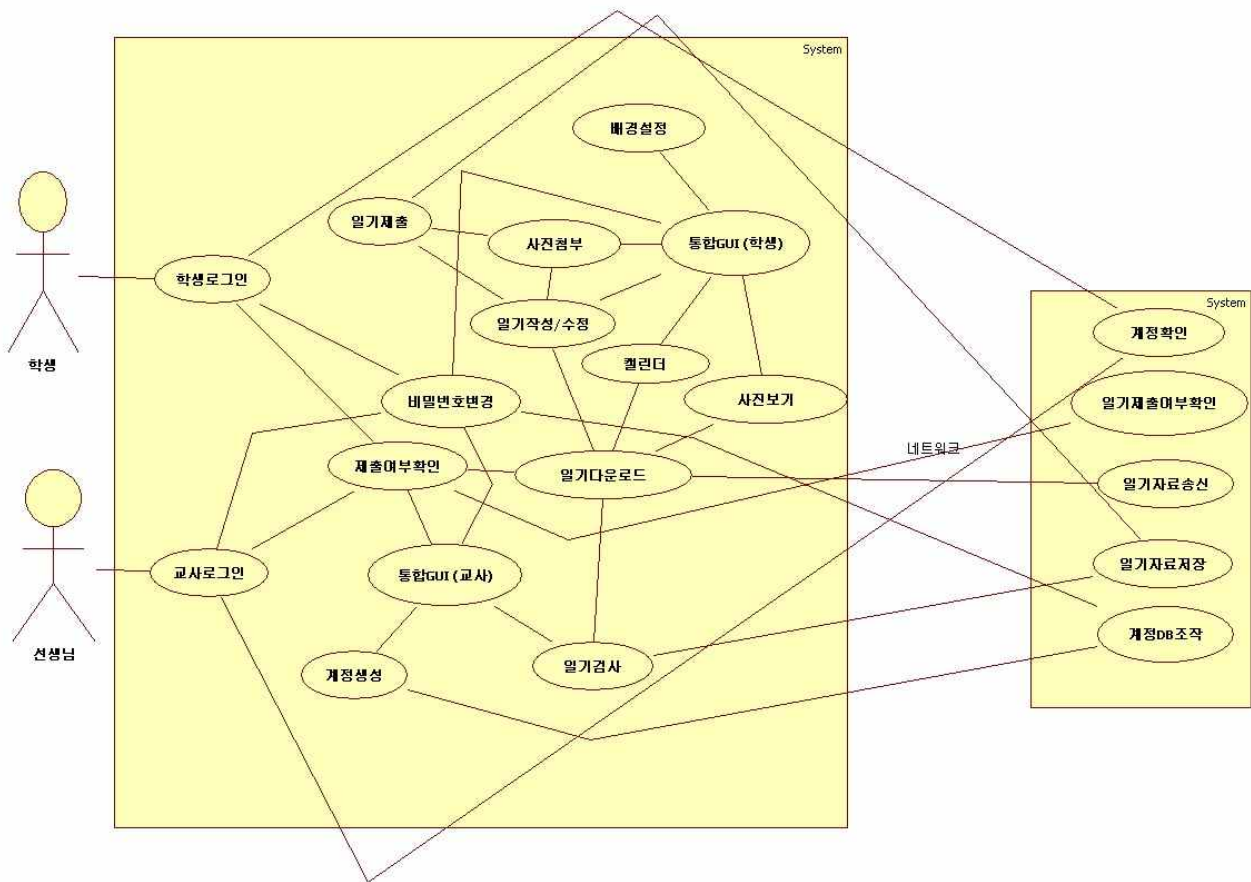
### 선생님

- 1) 아이디와 비밀번호를 입력하여 로그인 할 수 있어야 한다.
- 2) 담당할 학생의 계정을 생성할 수 있어야 한다.
- 3) 생성된 계정(학생)의 당일 일기 작성 여부, 검사 여부를 확인할 수 있어야 한다.
- 4) 작성된 일기를 검사할 수 있어야 한다.
- 5) 작성된 일기에 수정(주로 맞춤법 교정)을 할 수 있어야 한다.
- 6) 공지사항을 작성할 수 있어야 한다.

### 서버

- 1) 학생 측에서 보낸 데이터(일기와 이미지)를 저장할 수 있어야 한다.
- 2) 학생 측에서 요청한 데이터 (이전 일기 등)을 찾아 전송할 수 있어야 한다.
- 3) 선생님 측에서 수정된 데이터(검사 완료된 일기)를 수정할 수 있어야 한다.
- 4) 데이터베이스의 테이블에 데이터삽입을 할 수 있어야 한다.
- 5) 요청이 들어온 데이터와 테이블의 데이터가 맞는지 여부를 비교할 수 있어야 한다.

# Use Case Diagram



배경설정과 사진보기는 개발시간 부족으로 구현 실패

## 세부 유즈케이스

== > 유즈케이스 끝에 있는 (유즈케이스 XX - 000) 의 경우 유즈케이스 진행 도중 다른 유즈케이스에 해당하는 작업을 진행한 뒤 되돌아 오거나 다른 유즈케이스의 작업을 추가로 진행하는 것을 뜻합니다.

### == > 학생 유즈케이스

#### ● 학생 로그인

1. 로그인 폼에 ID, PW를 입력하고 버튼을 클릭한다. 로그인 타입은 '학생' 으로 맞춘다.
2. 서버에서 계정을 확인한다. (유즈케이스 서버 - 계정확인)
3. 로그인이 승인된다.
  - 3-1. 로그인이 승인되지 않는다. (패스워드 오류, 서버상의 문제, 인터넷 연결 안됨 등)
  - 3-2. 로그인 폼을 초기화 한다.
4. 로그인 폼을 닫고 해당 학생의 데이터를 가져오기 시작한다.
5. 일기 작성 GUI를 그린다.

#### ● 일기 작성/수정

1. 로그인을 성공한다.
2. 그려진 GUI에서 일기 제목을 입력한다.
  - 2-1. 당일 일기를 이미 작성하였으나 검사 되지 않은 경우
  - 2-2. 작성한 일기를 불러온다. (유즈케이스 학생,교사 - 일기다운로드)
3. 일기 내용을 입력한다.
4. 첨부할 사진이 있다면 사진을 첨부한다 (유즈케이스 학생 - 사진첨부)
5. 날씨 등을 입력한다.
6. 제출 버튼을 누른다 (유즈케이스 학생 - 일기 제출)

#### ● 사진첨부

1. 로그인을 성공한다.
2. (유즈케이스 학생,교사 - 일기다운로드)를 해서 첨부한 사진이 없다면 사진 첨부 버튼을 활성화 한다.
3. 사진 첨부 버튼을 누른다.
4. 사진 파일을 선택하여 첨부한다.
  - 4-1. 사진파일이 제대로 첨부되지 않았다. (너무 큰 용량, 적합하지 않은 확장자 등)
  - 4-2. 에러 메시지를 띄우고 첨부 창을 닫는다.

## ●일기 제출

1. 로그인을 성공한다.
2. 일기의 내용을 모두 입력한다. (사진은 선택사항)
  - 2-1. 일기의 내용이 모두 입력되지 않은 경우
  - 2-2. 어떤 부분이 입력되어 있지 않은지에 대한 메시지를 출력한다.
  - 2-3. 일기를 제출하지 않고 끝낸다. (3번부터 진행하지 않음)
3. 일기 제출 버튼을 누른다.
4. 일기가 서버로 전송되어 저장된다. (유즈케이스 서버 - 일기자료 저장)
  - 4-1. 이미 일기를 작성했다면 내용을 수정한다 (유즈케이스 서버 - 일기자료 수정)
5. 일기를 성공적으로 제출했다는 메시지를 띄운다.
  - 5-1. 일기 제출에 실패했다는 메시지를 띄운다. (서버 문제, 인터넷 연결 문제 등)
  - 5-2. 일기를 학생의 컴퓨터에 저장한다.

## ●캘린더

1. 로그인을 성공한다.
2. 일기를 보고싶은 날짜를 선택한다.
3. (유즈케이스 학생,교사 - 일기 다운로드)

## == > 교사 유즈케이스

### ●교사 로그인

1. 로그인 폼에 ID, PW를 입력하고 버튼을 클릭한다. 로그인 타입은 '선생님' 으로 맞춘다.
2. 서버에서 계정을 확인한다. (유즈케이스 서버 - 계정확인)
3. 로그인이 승인된다.
  - 3-1. 로그인이 승인되지 않는다. (패스워드 오류, 서버상의 문제, 인터넷 연결 안됨 등)
  - 3-2. 로그인 폼을 초기화 한다.
4. 로그인 폼을 닫고 해당 교사의 학생 일기 제출 목록을 가져온다.
5. 교사용 GUI를 그린다.

### ●계정 생성

1. 로그인을 성공한다.
2. 계정 생성 버튼을 누른다.
3. 계정 생성 폼을 그린다.
4. 관리할 학생의 정보(학번, 이름 등), ID, 초기 비밀번호를 입력한다.
5. 계정 생성 완료 버튼을 누른다.
6. 서버로 데이터를 전송한다 (유즈케이스 서버 - 계정 DB 조작)
  - 6-1. 정보가 전부 입력되지 않은 경우
  - 6-2. 에러메시지를 띄우고 데이터 전송을 하지 않는다.

### ●제출여부확인

1. 로그인을 성공한다.
2. 로그인 성공 후 가져오는 제출 여부 목록에 따라 GUI에 제출 여부를 표시한다.
3. 제출 완료한 학생을 클릭하여 해당 학생이 작성한 일기를 검사한다 (유즈케이스 학생,교사 - 일기 다운로드), (유즈케이스 교사 - 일기 검사)

### ●일기 검사(맞춤법 검사)

1. 로그인을 성공한다.
2. 제출 여부에서 제출 완료한 학생을 클릭하여 가져온다. (유즈케이스 학생,교사 - 일기 다운로드), (유즈케이스 교사 - 일기 검사)
3. 일기를 읽고 도장 선택 버튼을 눌러 도장을 선택한다.
4. 학생에게 남길 말이나 맞춤법을 검사하여 기록한다.
5. 검사 완료 버튼을 눌러 서버로 데이터를 전송한다. (유즈케이스 서버 - 일기 자료 수정)

## == > 학생.교사 공통 유즈케이스

### ●비밀번호 변경

1. 로그인을 성공한다.
2. 비밀번호 변경 버튼을 누른다.
3. 새롭게 그려진 변경 폼에 현재 비밀번호와 바꿀 비밀번호를 입력한다.
4. 변경 버튼을 눌러 비밀번호를 변경한다. (유즈케이스 서버 - 계정 DB 조작)
5. 비밀번호 변경에 성공했다는 메시지를 띄운다.

5-1. 비밀번호 변경에 실패했다는 메시지를 띄운다. (데이터 전송 실패, 현재 비밀번호 입력 오류, 비밀번호 길이 오류 등)

5-2. 입력 폼을 초기화 한다. (6번 실행 안함)

6. 폼을 닫는다.

### ●일기 다운로드

1. 다운로드가 필요한 상황에 서버에 해당 일기 자료를 요청한다.
2. 서버에서 일기 자료를 받아온다.
  - 2-1. 서버에서 일기 자료를 받아오지 못한다. (일기 없음, 인터넷 연결 문제 등)
  - 2-2. 일기가 없다는 이벤트를 발생시킨다.
3. 일기 다운로드에 성공했다는 이벤트를 발생시킨다.

## == > 서버 유즈케이스

### ●계정 확인

1. 클라이언트에서 계정에 관련된 데이터를 수신한다.
2. DB에 접근하여 해당 계정이 있는지 여부를 검사한다.
3. 계정이 있다면 계정이 있다는 신호를 송신한다. 상황에 따라 추가적인 데이터도 송신한다.
  - 3-1. 계정이 없다면 계정이 없다는 신호를 송신한다.

### ●일기 자료 저장/송신/제출여부 확인

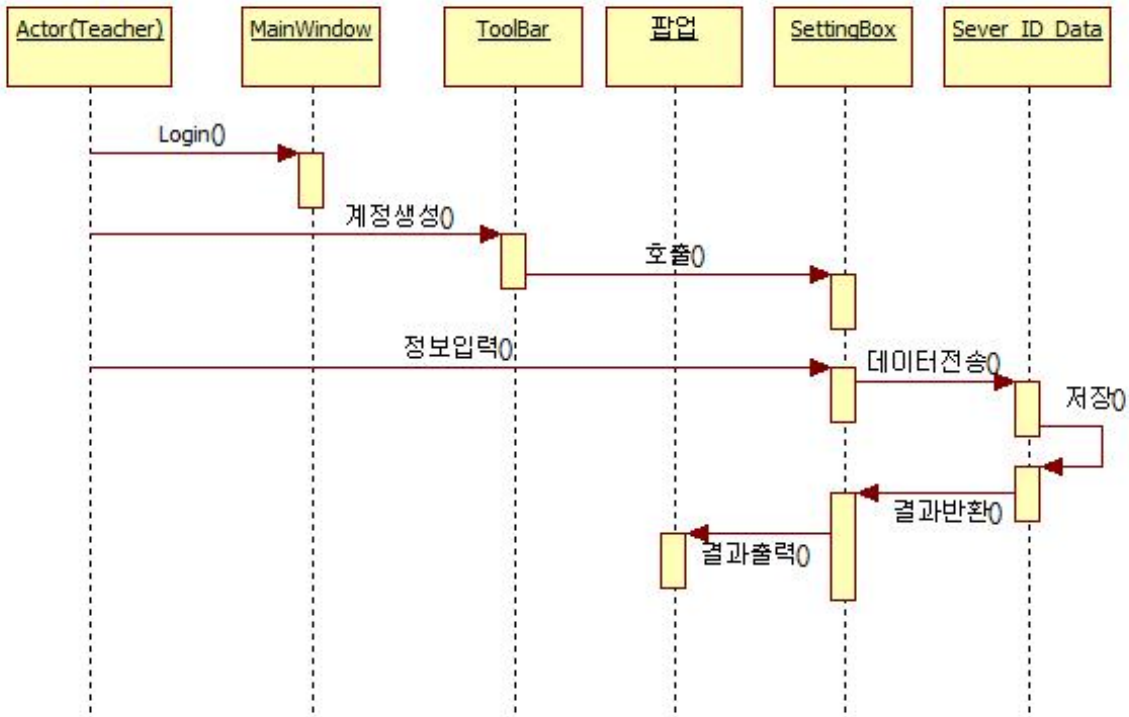
1. 클라이언트에서 일기 관련 데이터와 계정 데이터를 수신한다.
2. 해당 계정에 같은 날짜의 일기 파일이 있는지 검사한다.
3. 일기파일이 없다면 새로운 파일을 만들고 그곳에 일기를 저장한다.
  - 3-1. 일기 파일이 있다면 해당 파일을 백업하고 새롭게 파일을 만들어 저장한다.
  - 3-2. 사진 파일이 있다면 같은 위치에 사진 파일도 업로드 한다.
4. 일기 저장이 완료되었다는 신호를 송신한다.

### ●계정 DB 조작

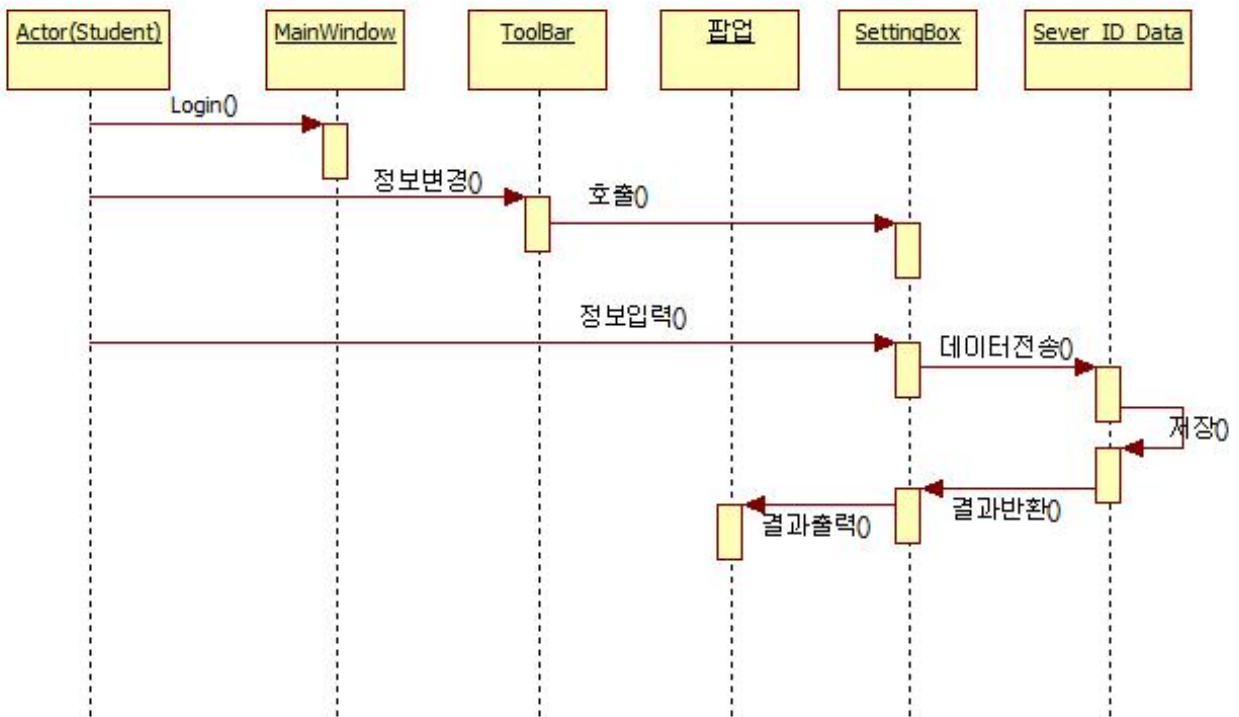
1. 클라이언트로부터 요청 내용(계정생성이나 비밀번호 변경 등)과 관련 데이터를 수신한다.
2. 요청 내용에 따라 DB에 접근하여 관련된 데이터를 수정한다.
3. 요청이 처리되었다는 신호를 송신한다.
  - 3-1. 요청이 실패되었다는 신호를 송신한다.



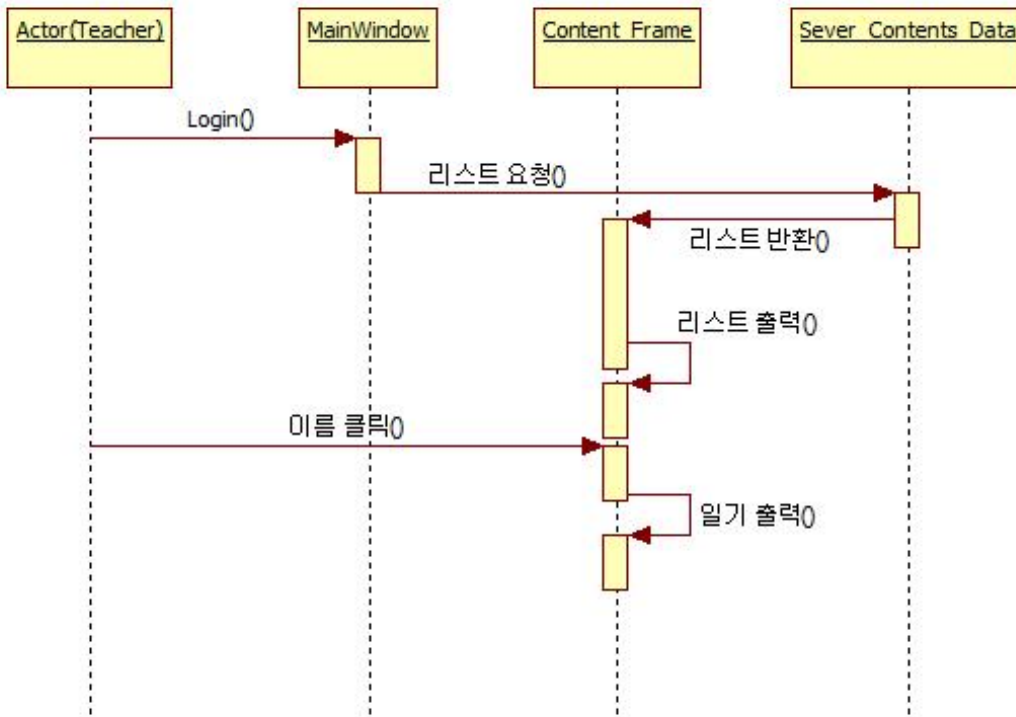
## 시퀀스 다이어그램



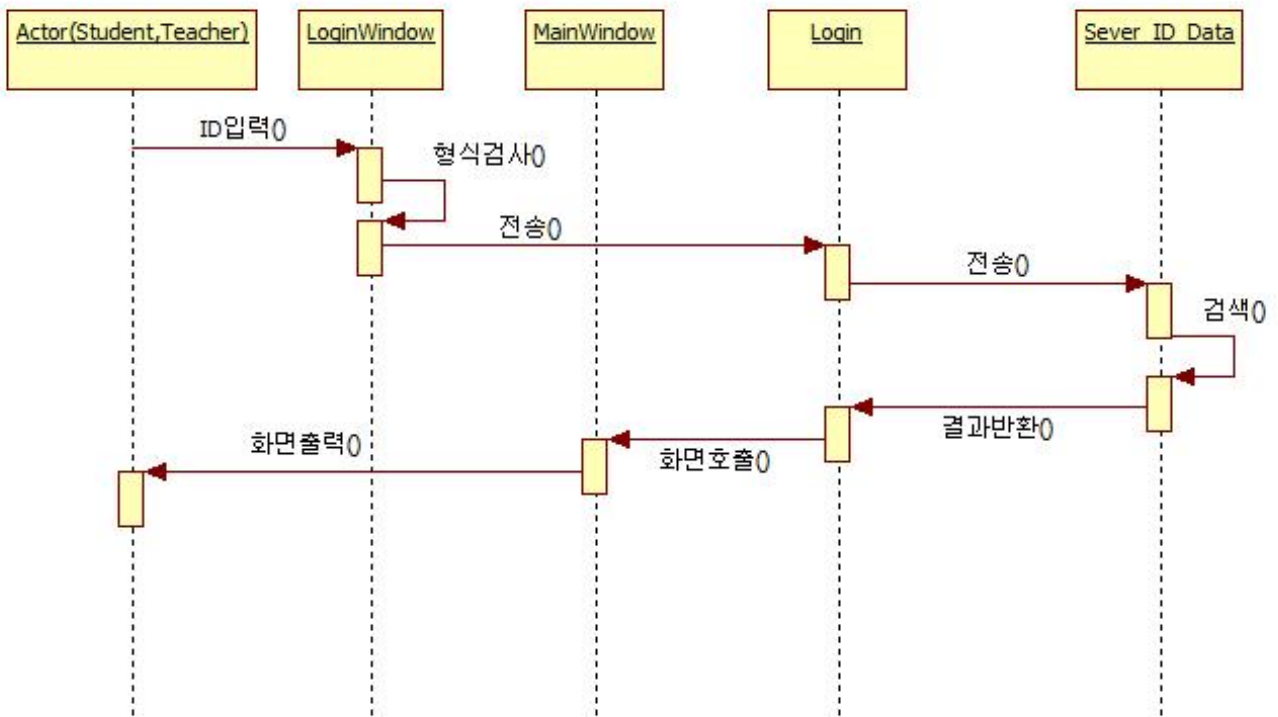
▲(교사용) 계정 생성 시퀀스



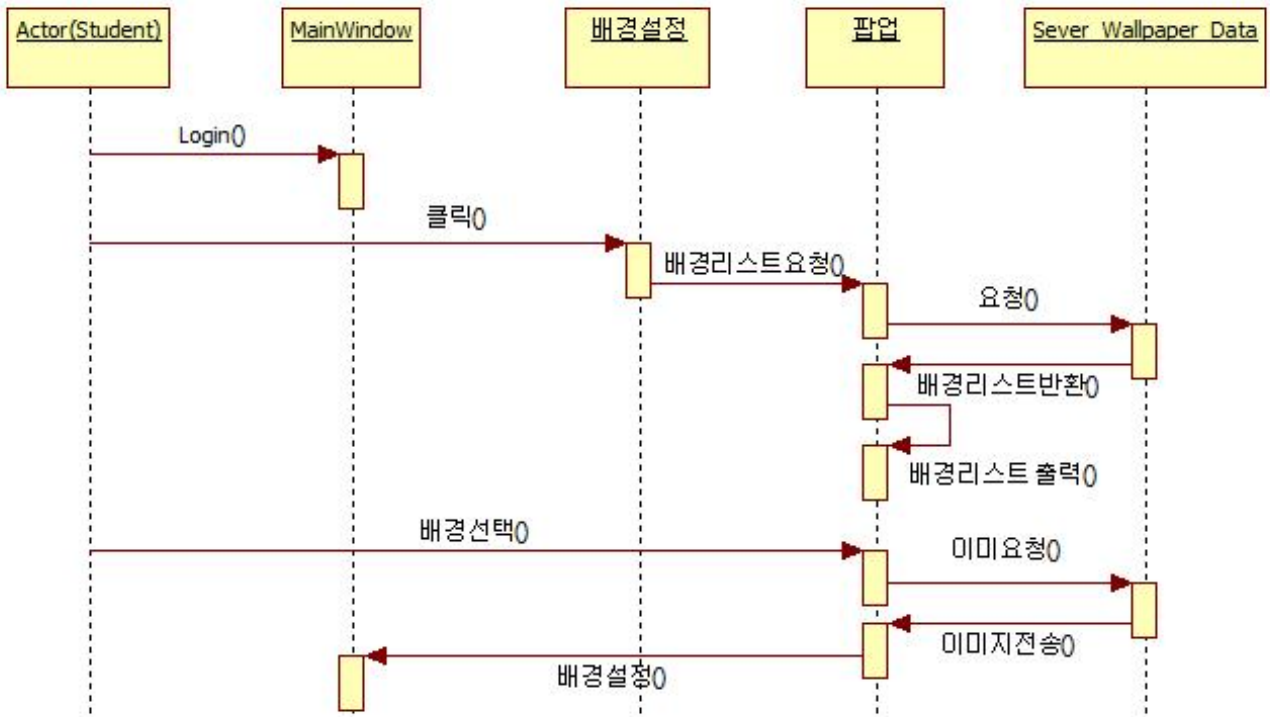
▲(공통) 계정 정보 변경 (비밀번호 등) 시퀀스



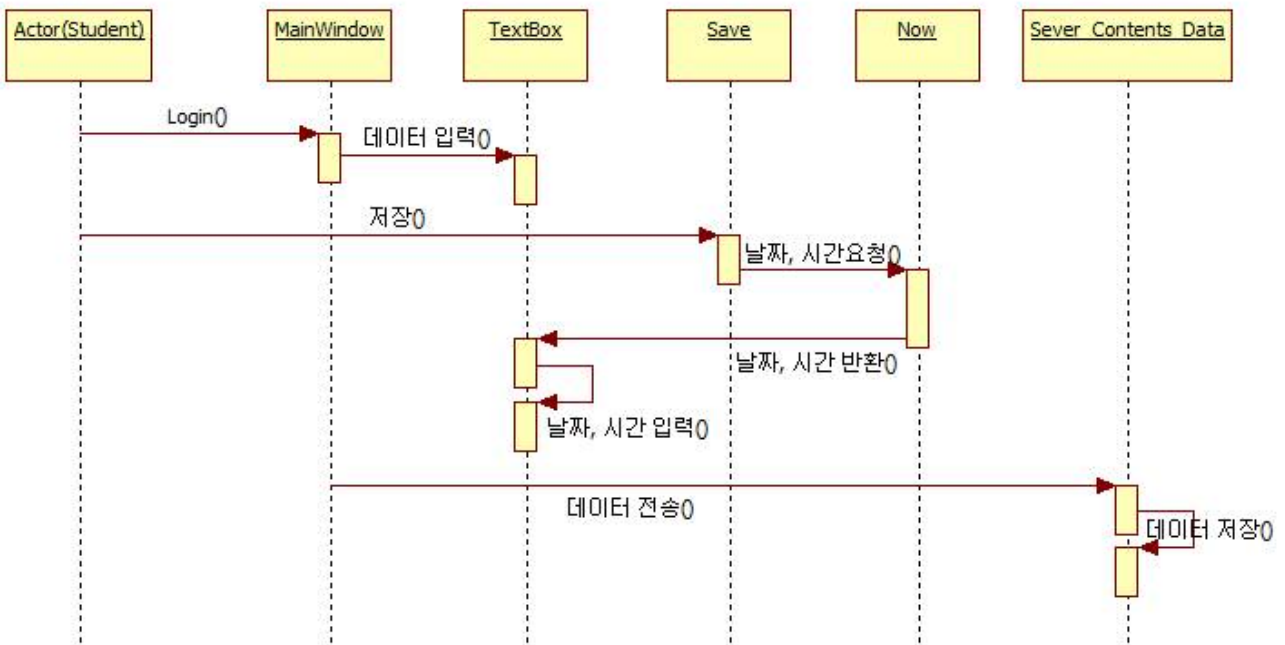
▲(교사용) 일기 제출 여부 확인 시퀀스



▲(공통) 로그인 시퀀스

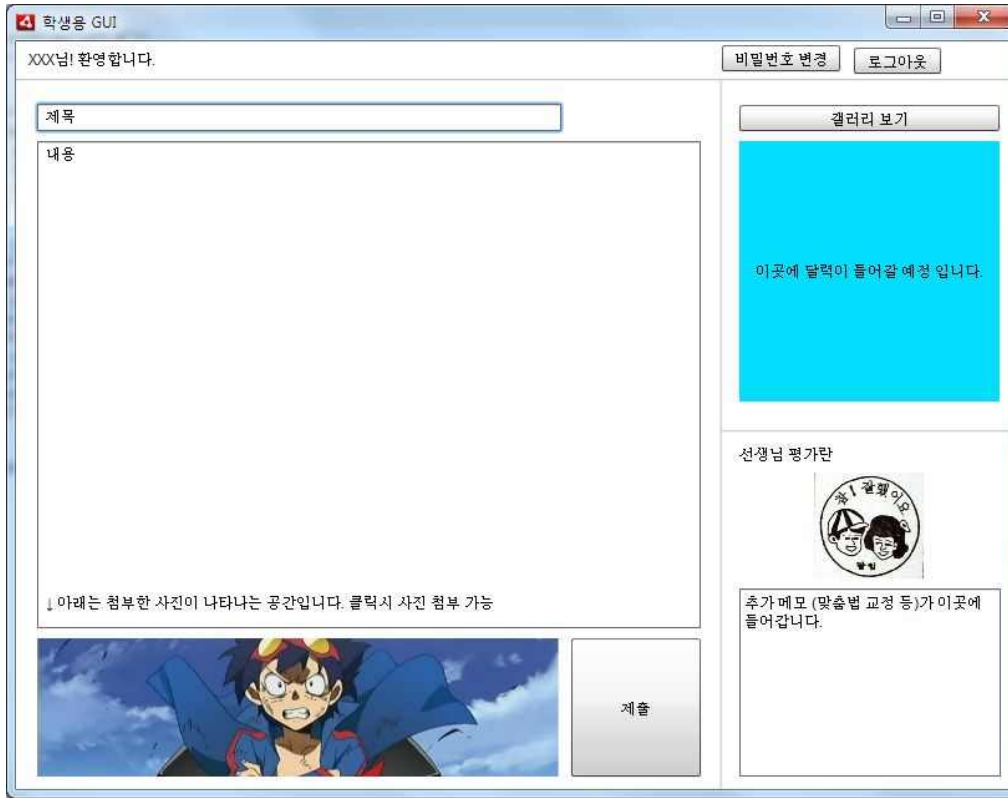


▲(학생용) 일기 배경 설정 시퀀스

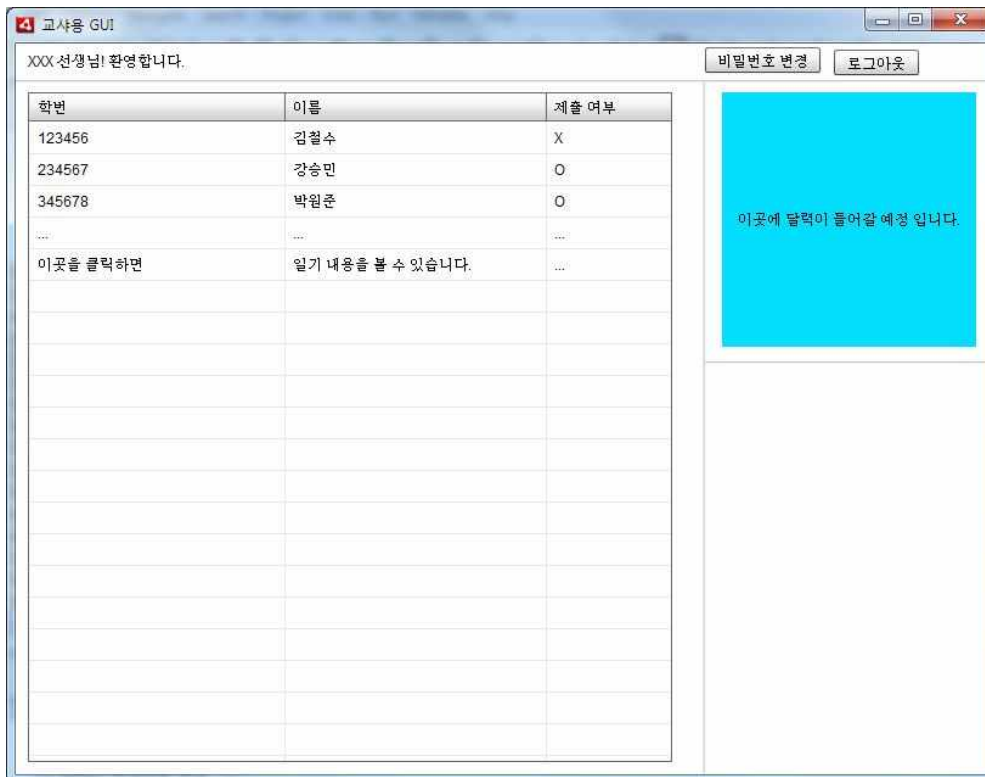


▲(학생용) 일기 작성 시퀀스

## 화면 정의서



▲ 학생용 GUI 화면 정의서 (일기 작성 화면)

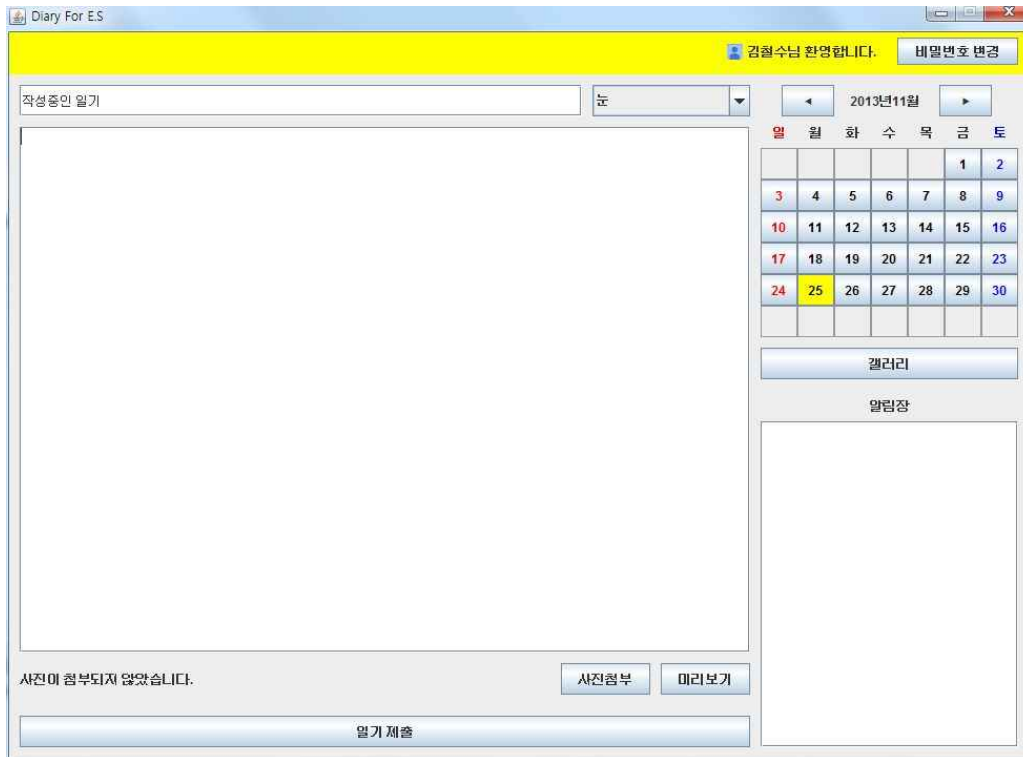


▲ 교사용 GUI 화면 정의서 (제출 여부 확인)

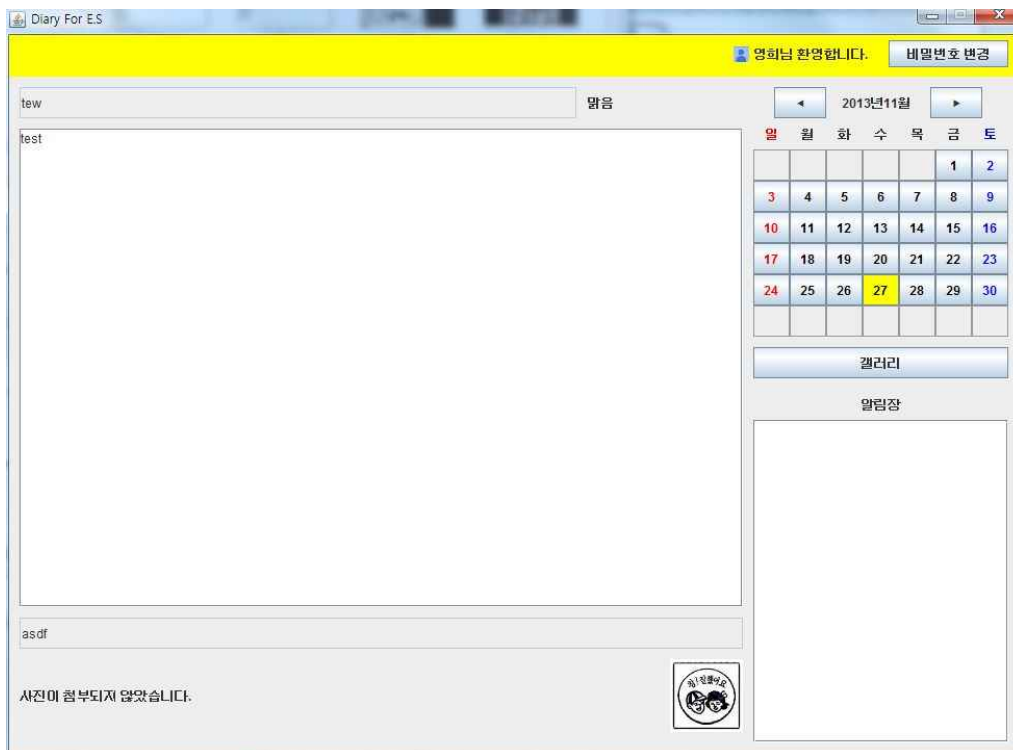


▲ 교사용 GUI 화면 정의서 (학생 일기 검사)

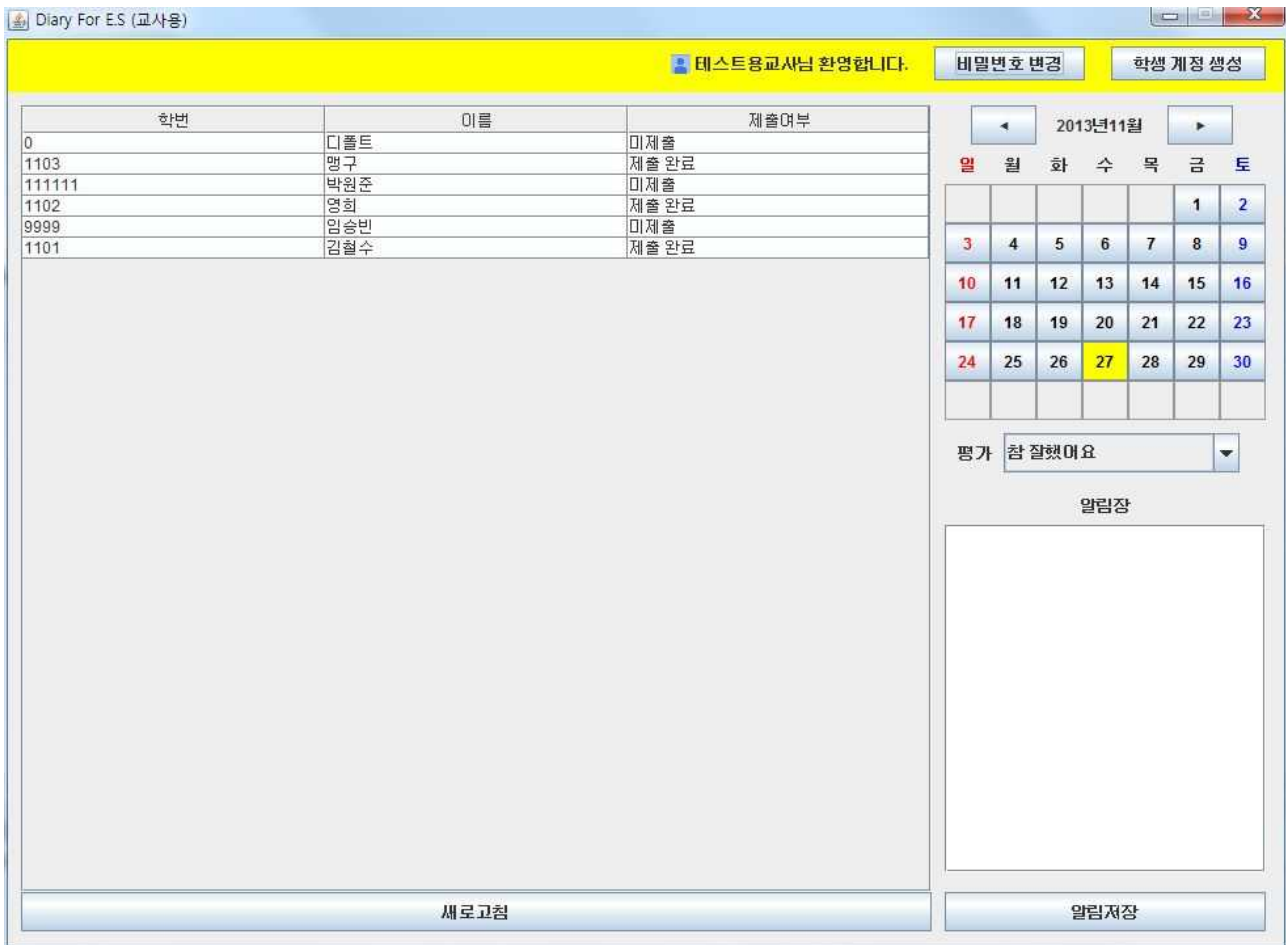
## 실제 구현 모습



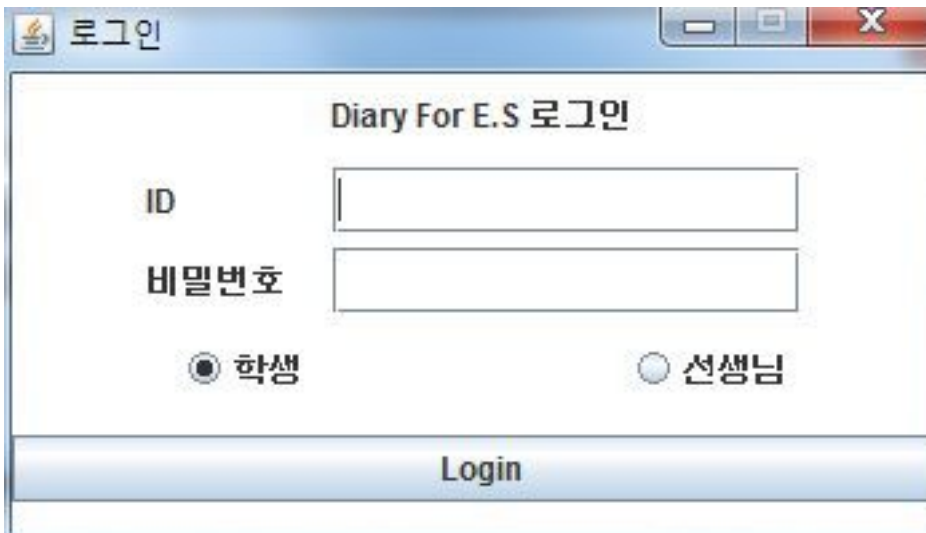
### ▲ 학생용 GUI에서 일기를 작성중인 모습



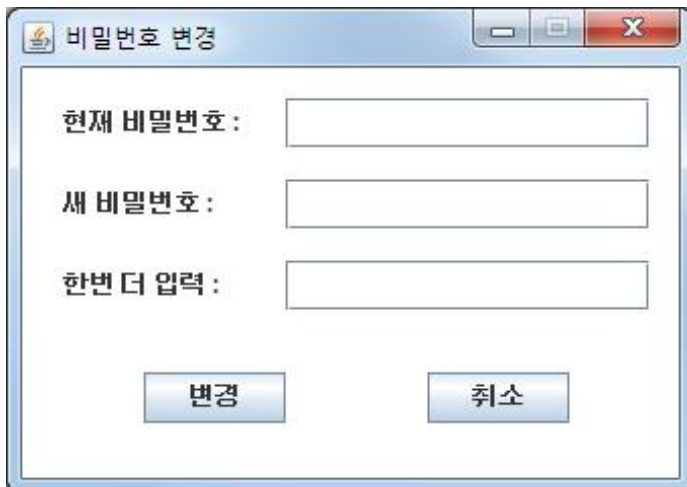
### ▲ 학생용 GUI에서 검사가 완료된 일기를 보는 모습



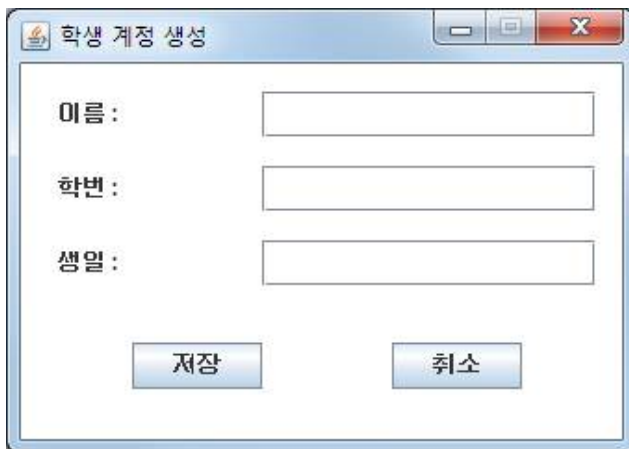
▲교사용 GUI에서 일기 제출 여부를 확인하는 모습



▲로그인 창의 모습



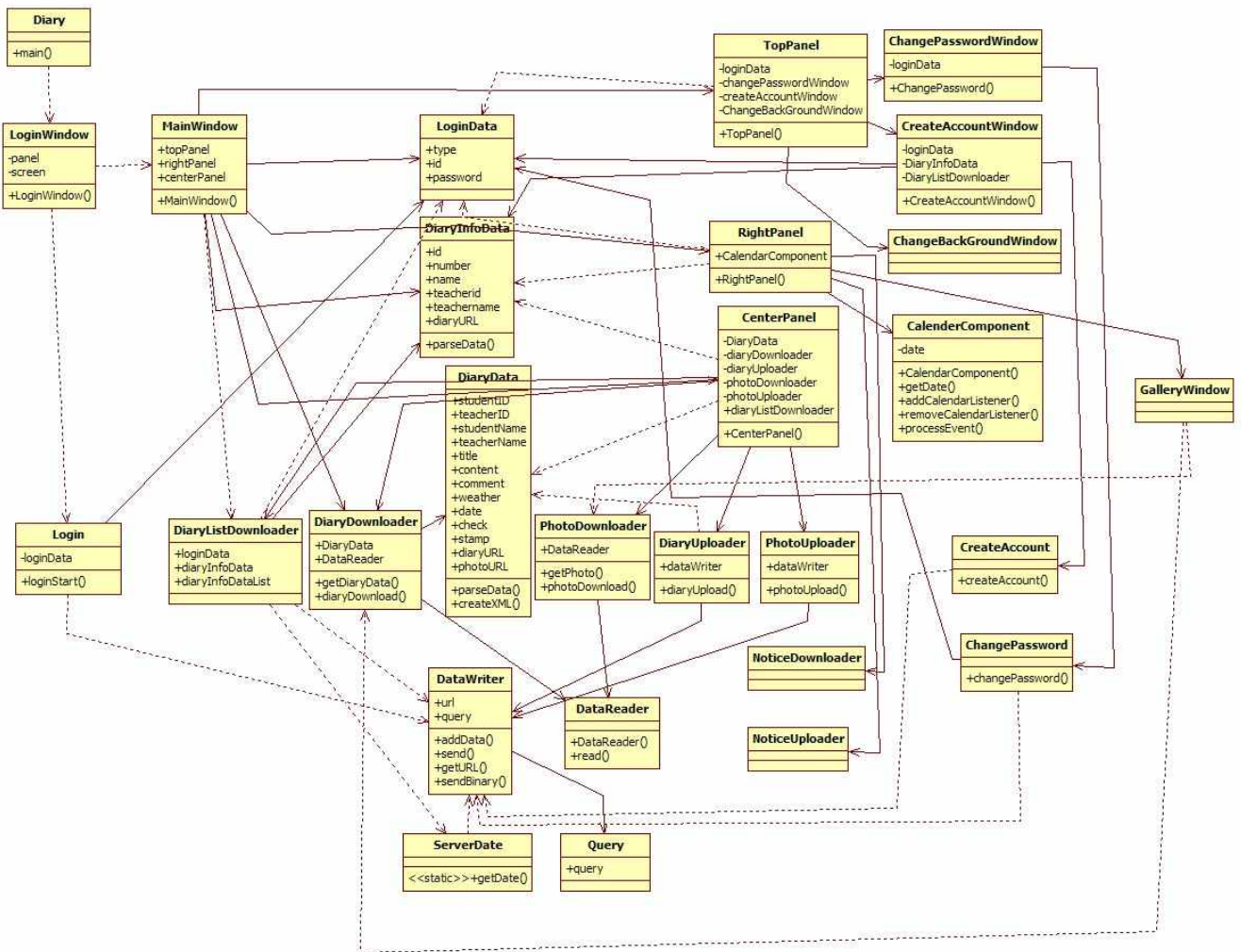
▲비밀번호 변경 GUI 구현 모습



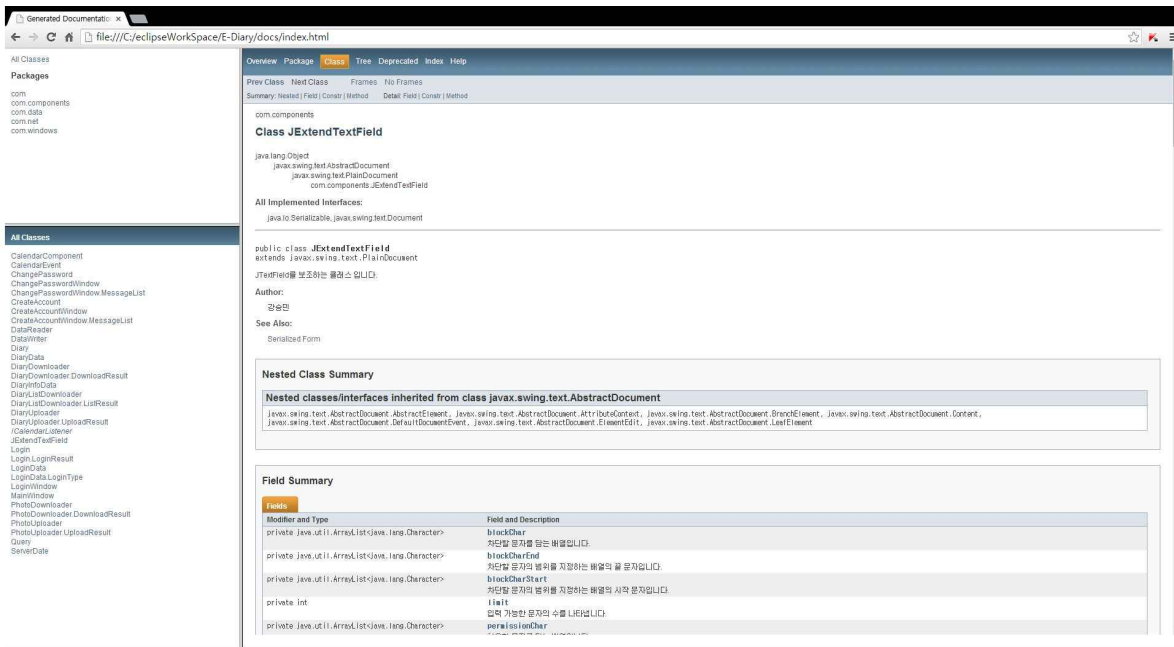
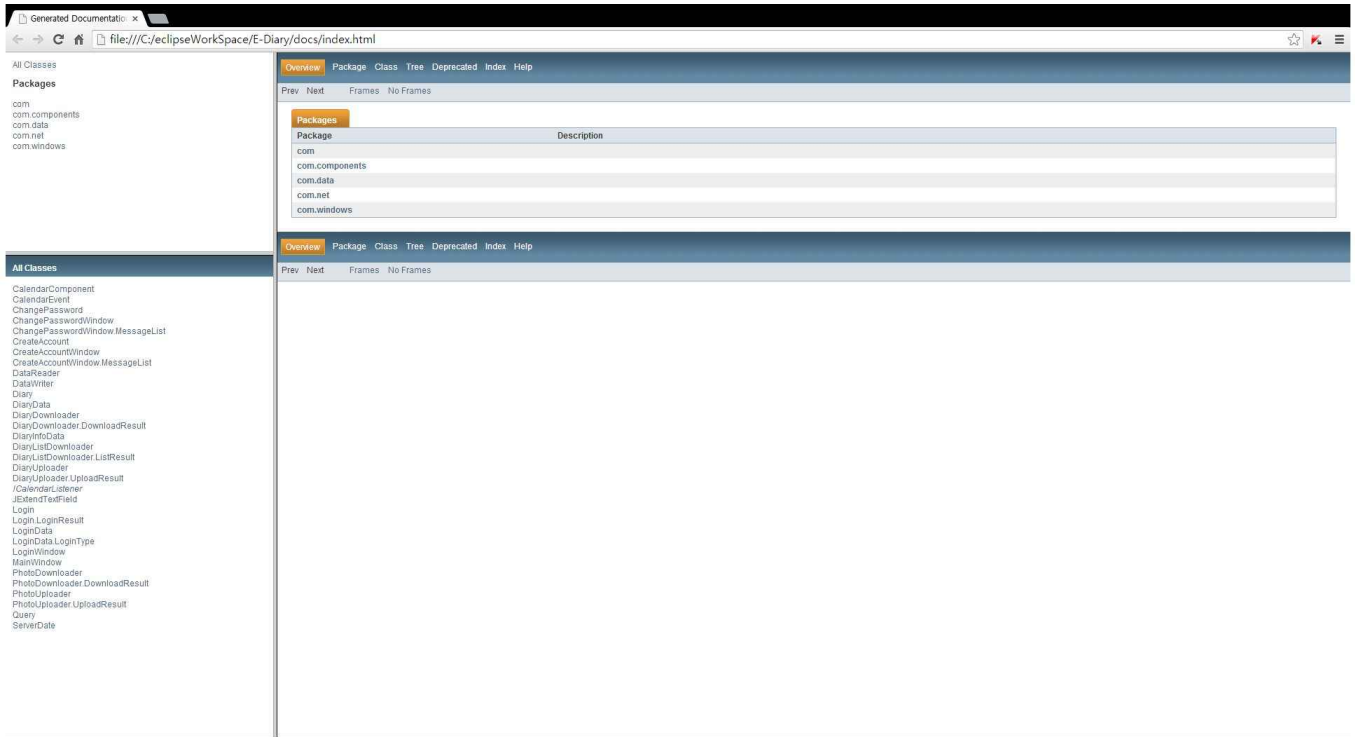
▲학생 계정 생성 GUI의 구현 모습



# 클래스 다이어그램



# API 매뉴얼 화면



# 팀원 간 작업 분배 계획 및 결과

## 작업 분배 계획

강승민

### 클라이언트

- components.JExtendsTextField
- components.CalendarComponent
- components.CalendarEvent
- components.ICalendarListener
- data.DiaryData
- data.DiaryInfoData
- data.LoginData
- net.ChangePassword
- net.CreateAccount
- net.DataReader
- net.DataWriter
- net.DiaryDownloader
- net.DiaryListDownloader
- net.PhotoDownloader
- net.DiaryUploader
- net.PhotoUploader
- net.Login
- net.ServerDate
- net.Query
- net.NoticeDownloader
- net.NoticeUploader

### 서버

- changepassword.php
- createaccount.php
- diarylist.php
- login.php
- servertime.php
- upload.php

## 박원준

Diary

windows.ChangePasswordWindow

windows.CreateAccountWindow

windows.LoginWindow

windows.MainWindow 및 하위 패널과 ActionListener들

## 작업 분배 결과

초기에 계획했던 작업 분배 계획대로 잘 분배하여 작업을 할 수 있었다.

승민은 네트워크 (com.net 패키지) 및 코딩에 난이도가 높은 부분 (CalendarComponent)을 담당하였고, 원준은 GUI (com.windows)와 이벤트 부분 (각종 ActionListener들)을 도맡아 작업을 수행할 수 있었다.