

Team Project - OOAD

Timetable

(instructor :: 유준범 교수님)

Konkuk univ. dept of CSE

Team 4

200611517 정훈섭

200711420 권준수

200711448 오희수

200710118 유희찬

Team Project - Timetable development

Contents

Requirement Satisfaction

Test Report

Changed Design

Expansibility

Requirement satisfaction

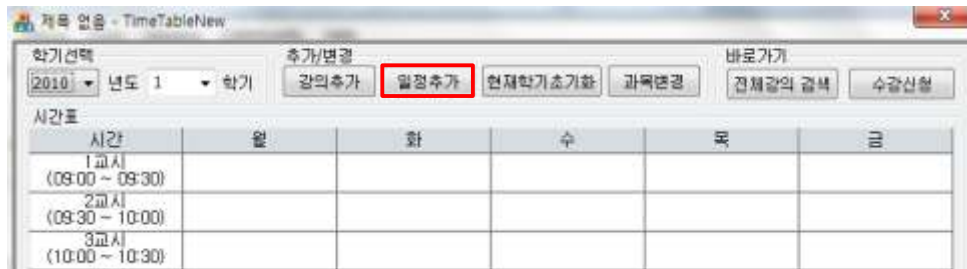
Analysis, Design, 그리고 Implementation 단계까지 진행되어 최종 'TimeTable' 시스템이 완성되었습니다. 이전에 Analysis 단계에서 조사하였던 requirement들이 실제 완성된 TimeTable에 제대로 적용이 되었는지 확인해 봅니다.

다음은 Analysis 단계에서 작성하였던 requirement description입니다.

Requirement specification

1. 학교 강의 외 개인 일정(언교원, 학원, 봉사활동 등) 추가 가능
2. Java의 jdk 추가 설치 없이 가볍게 파일 하나로 실행
3. 직관적으로 알 수 있는 사용 방법
4. 저장/불러오기 기능 추가
5. 학과별 강의 목록 출력(컴퓨터공학부 강의로 한정)
6. 강의계획서 보여주기

R-1 학교 강의 외 개인 일정 추가 기능

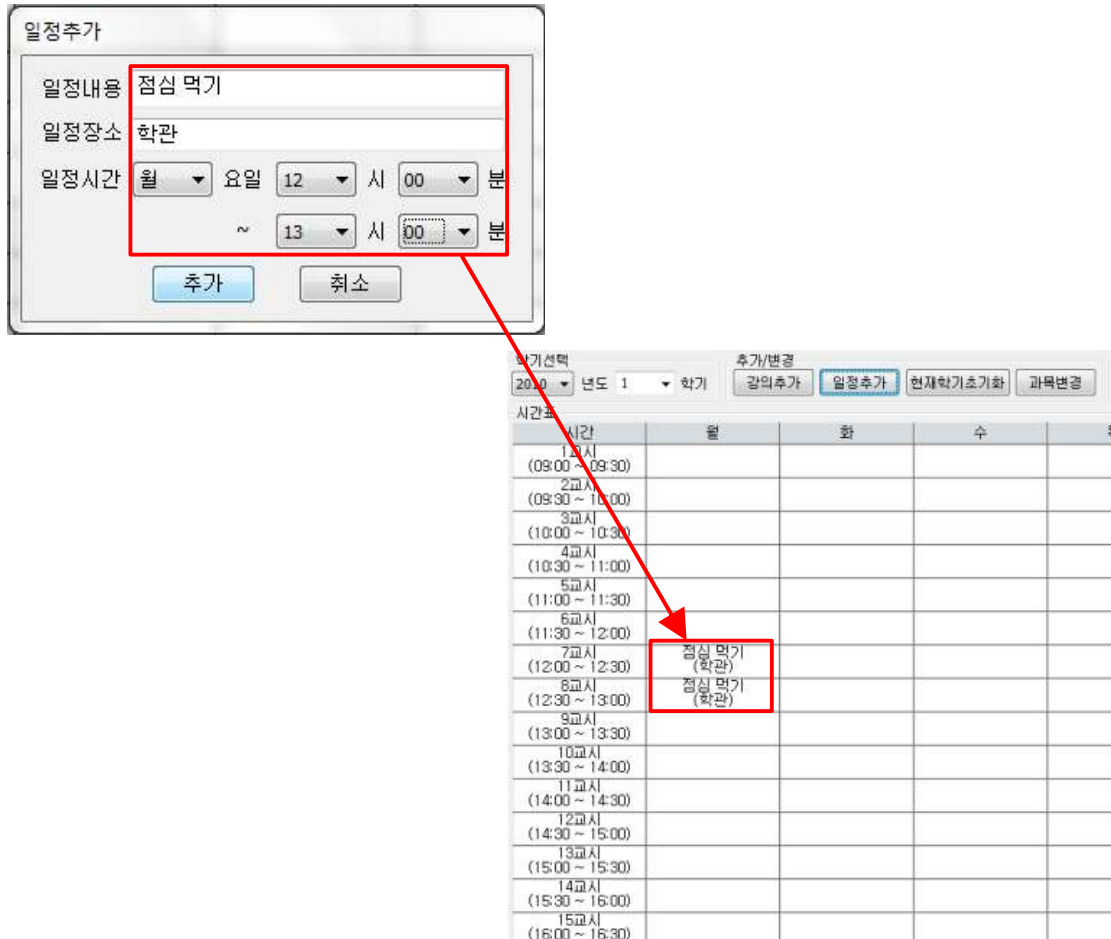


프로그램을 실행해 보면 상단 중앙에 추가/변경 메뉴들이 모여 있습니다. 이 중에서 우리가 원하는 '일정추가' 버튼을 클릭합니다.

The '일정추가' dialog box contains the following fields and controls:

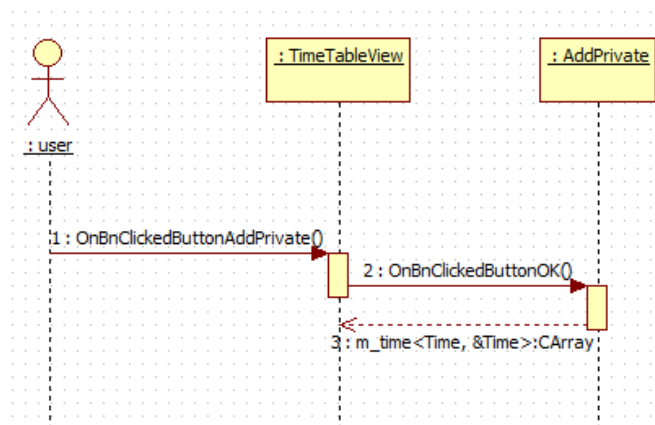
- '일정내용' (Schedule Content): A text input field.
- '일정장소' (Schedule Location): A text input field.
- '일정시간' (Schedule Time): A dropdown menu.
- '요일' (Day of the Week): A dropdown menu.
- '시' (Hour): A dropdown menu.
- '분' (Minute): A dropdown menu.
- '~' (Tilde): A symbol indicating a range.
- '시' (Hour): A dropdown menu.
- '분' (Minute): A dropdown menu.
- '추가' (Add): A button.
- '취소' (Cancel): A button.

팝업창이 뜨면 일정의 내용, 장소와 시간을 입력하여 추가를 클릭하면 다음과 같이 일정이 추가된 것을 볼 수 있습니다.



일정 내용, 장소는 콤보 박스를 이용하지 않아, 제시된 몇 가지 중에서 선택하는 것이 아닌 사용자가 자유로이 입력할 수 있습니다.

수행 내용을 좀 더 자세히 보면, 다음 sequence diagram과 같이 수행됩니다.



사용자가 '일정추가' 버튼을 누르게 되면, TimeTableView 객체의 OnBnButtonAddPrivate() 함수가 호출되게 됩니다. 화면 상에는 '일정 추가' 팝업창이 뜨고 사용자로부터 일정을 입력 받을 준비를 합니다. 일정을 입력받고, 사용자가 '확인' 버튼을 누르게 되면 AddPrivate 객체의 OnBnClickedButtonOK() 함수가 호출되어지고, TimeTableView로 사용자가 입력한 일정의 정보 (시간, 장소 등등..) 가 전달되어 집니다. TimeTableView 객체에서는 이 데이터를 이용하여 화면에 일정을 출력하게 됩니다.

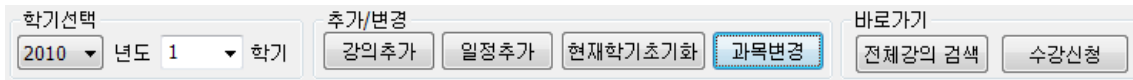
R-2 (Java jdk 등) 추가 설치 없이 가볍게 파일 하나로 실행

Java를 이용할 경우 사용하는 컴퓨터에 jdk가 설치되어 있지 않으면 실행이 되지 않습니다. 물론, jdk가 자동적으로 설치되고, jdk를 한 번 설치하면 특별히 업데이트가 필요하지 않는 이상 제대로 작동을 하나, jdk에 대한 지식이 없는 (다수의) 사용자가 느끼기에 번거롭고 어렵게 다가갈 수 있습니다. 이를 해결하기 위해 프로젝트 진행을 MFC를 이용하여 개발하였습니다.

R-3 직관적으로 알 수 있는 사용 방법

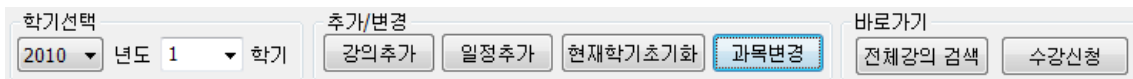


복잡하게 많은 기능을 구현하지도 않았을뿐더러 추가된 기능들도 기존에 있던 건국대학교 종합강의시간표나 다른 시간표 프로그램과 유사하게 작성하여 처음 써보는 사용자라도 **이전의 '경험' 을 바탕으로** 사용법을 '직관적' 으로 알 수 있도록 제작했습니다.

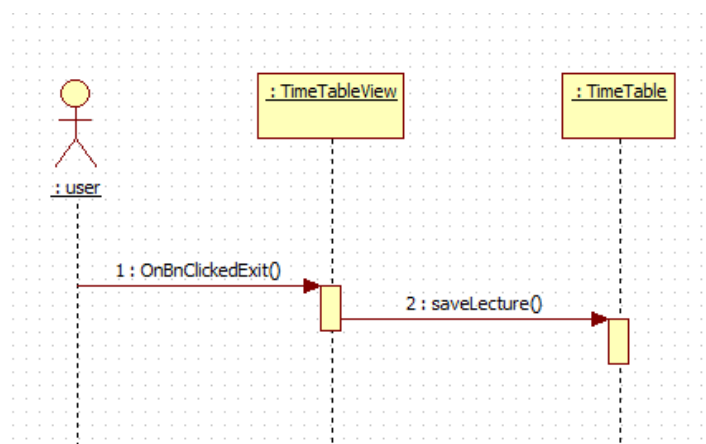


또한 위 그림과 같이 실제 사용자가 선택을 하여 기능이 수행되게 하는 메뉴는 **한 곳에 모아둠**으로써 사용자들이 해매지 않고 원하는 메뉴를 선택할 수 있게 했습니다.

R-4 저장/불러오기 기능 추가



메뉴에서 보이는 바와 같이 현재 프로그램에서는 저장/불러오기 버튼이 보이지 않습니다. 그렇다고 저장/불러오기 기능을 지원하지 않는 것은 아닙니다. 사용자 편의를 위하여 저장/불러오기를 따로 버튼을 클릭하여 저장하는 것이 아닌, 사용자가 프로그램을 종료할 때 자동으로 저장되고, 또 프로그램을 실행할 때 자동으로 불러옵니다.

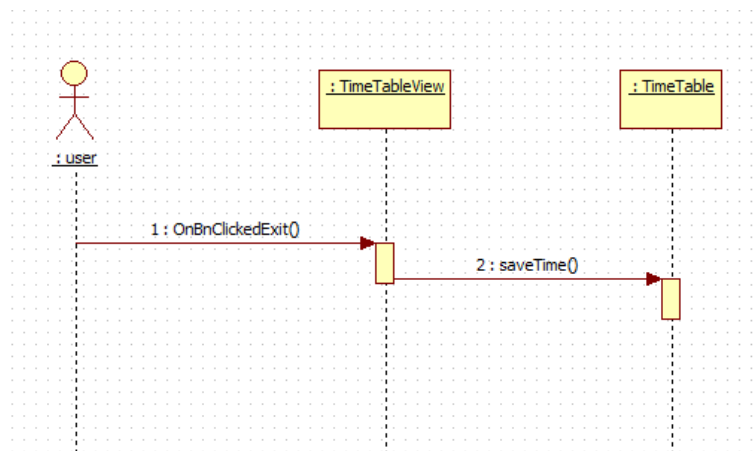


위의 sequence diagram과 같이, 사용자가 프로그램을 '종료' 하게 되면 TimeTableView 객체에서는 TimeTable 객체의 saveLecture() 를 호출함으로써 현재 TimeTable의 정보를 저장하게 됩니다.

R-5 학과별 강의 목록 출력

위의 그림에서처럼 콤보박스를 내리면 단과대가 나오고, 단과대를 선택하면 해당 단과대의 강의 목록을 아래의 콤보박스에서 찾을 수 있습니다. 이번 Demo에서는 제공하지 않지만 첫 배포 시에는 DB에는 컴퓨터공학부 강의만 들어갈 예정입니다. 사용자는 별도의 입력 없이 클릭만으로 손쉽게 강의를 추가할 수 있습니다.

또한, 학기 시작 후에 강의실이나 시간이 변경이 되었을 경우가 생길 수 있습니다. 사용자는 배포자가 새로운 DB를 제공하기 이전에 직접 스스로 과목 변경 메뉴를 선택하여 이미 저장된 과목을 변경하여 사용할 수도 있습니다.

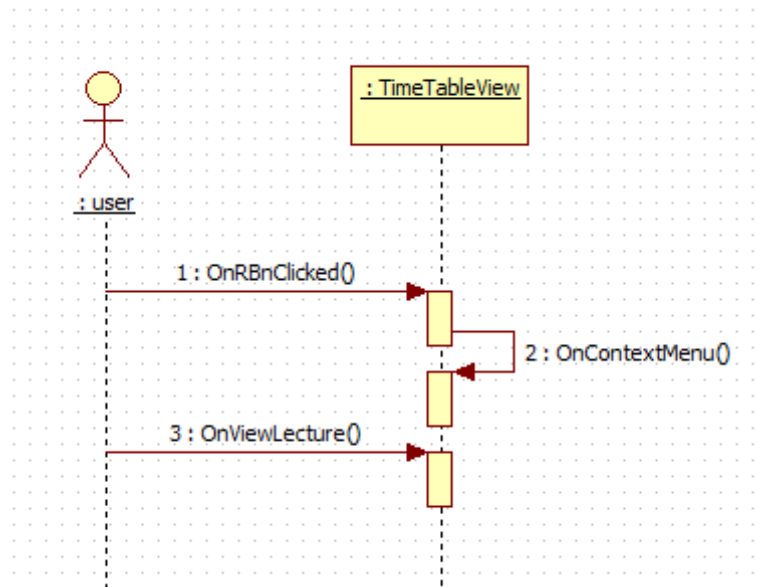


Sequence diagram에서 처럼 '강의 정보 변경' 을 통해 변경된(혹은 추가된) 강의 정보도 위의 저장/불러오기 기능 때처럼 프로그램이 종료될 때, saveTime() 함수가 호출되어 같이 저장되어 집니다.

R-6 강의계획서 보여주기

4교시 (10:30 ~ 11:00)			
5교시 (11:00 ~ 11:30)	소프트웨어공학 (새502/유준범)		
6교시 (11:30 ~ 12:00)	소프트웨어공학 (새502/유준범)	강의계획서보기	
7교시 (12:00 ~ 12:30)	소프트웨어공학 (새502/유준범)	강의삭제	
8교시 (12:30 ~ 13:00)	소프트웨어공학 (새502/유준범)		
9교시			

시간표를 보다가 강의의 강의계획서를 보고 싶으면 해당 강의의 칸을 오른쪽 클릭을 하여 나오는 메뉴에서 '강의계획서보기' 를 클릭합니다. 이 때, 강의계획서는 건국대학교에서 제공하는 강의계획서 웹페이지를 띄우게 됩니다.



다음과 같이 사용자가 오른쪽 클릭을 하게되면, TimeTableView 에서는 OnContextMenu() 함수를 호출하여 메뉴를 띄어주고, 다시 사용자가 OnViewLecture()를 선택하면 칸에 해당하는 강의정보를 인터넷 창으로 팝업을 띄어줍니다.

기타 편의 사항



메뉴의 '바로 가기' 메뉴를 보면, 전체강의 검색 및 수강신청 버튼이 있습니다. 프로그램을 이용하여 작성한 시간표를 토대로 바로 학교 수강신청 홈페이지로 이동하여 수강신청을 할 수도 있고, 프로그램에 새로운 강의DB를 추가하기 위하여 직접 학교 종합강의시간표에서 강의를 확인하여 추가하거나 기존 강의의 정보를 수정할 수도 있습니다.

Test

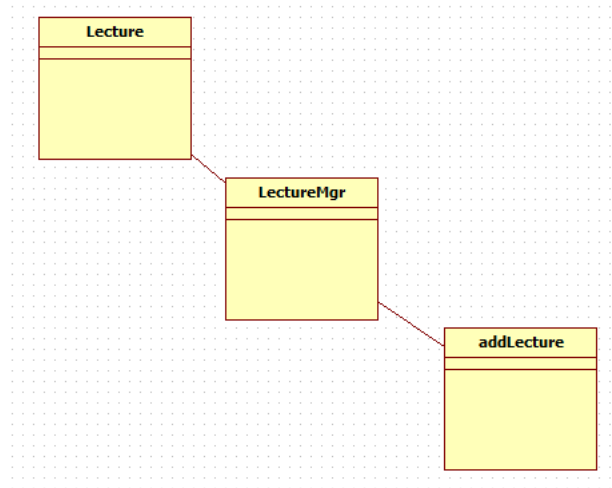
배포를 하기 전에 내부적으로 각 요구사항에 대해서 각 기능들이 제대로 작동하고 있는지 확인하기 위하여 function test를 해보았습니다. 각 test는 항목 별로 팀원 당 10 번 정도 반복하였고, test를 통해 나온 error들은 현재 버전에서는 모두 수정되었습니다.

Test	Result
과목변경	pass
현재 학기 초기화	초기화 시 한 번에 초기화 되지 않고, 2~3번에 걸쳐서 초기화 됨.
전체강의 검색	pass
수강신청	pass
강의계획서 보기	pass
강의 삭제	pass
일정 추가	끝 시간을 19시로 하면 다음 날 아침 영역까지 침범, 후에 해당 칸 일정 추가 안됨
강의 추가	현재 학기 초기화 후 같은 과목 추가 안 됨(겉으로는 과목이 지워졌지만, 내부적으로 지워지지 않은 것 같음)
저장/로드	pass

프로젝트의 기간 관계상 다양한 예외 상황에 대한 test는 해보지 못했지만, 보이는 기능들은 제대로 수행됨을 확인했고, 배포 후에 사용자들로부터 error를 보고 받아 수정할 계획입니다.

Why Design was changed??

1. 본래 design 단계에서 다음 그림과 같이 Lecture에 접근하기 위해서는 Lecture를 관리하는 ClassMgr 클래스를 거쳐야 했었습니다.



LectureMgr에서 Lecture 클래스의 벡터의 정보를 초기화하고, 원하는 Lecture을 찾는 방식으로 design을 했었습니다. 하지만 실제 구현할 때, MFC API를 이용하여 GUI를 구현하려 보니, LectureMgr과 addLecture 클래스가 서로 기능이 중복됨을 발견했습니다. addLecture에서 MFC API를 상속받아 팝업창을 띄우고, 마우스 이벤트를 처리하는 과정에서 강의 등록하는 역할을 수행하고 있습니다. 이는 design 단계에서 GUI를 생각하지 않은 채 설계한 것에서 비롯되었고, 마우스 이벤트를 처리하는 클래스에서 바로 기능을 수행하는 것이 좋다고 판단하여 addLecture 클래스에서 바로 Lecture 클래스의 벡터를 만들어 직접 처리하도록 하였습니다. 역시 마찬가지로 이유로 역시 매니저 클래스를 없애고 바로 addPrivate 클래스에서 기능을 수행하도록 하였습니다.

2. DBMgr 클래스도 삭제되었습니다. 이 역시 기존의 TimeTable 클래스에서 초기화, 출력, 추가, 삭제 등의 전반적인 기능을 관리하고 있고, 현재 화면에 출력되고 있는 time 클래스의 데이터를 저장하기 때문에 굳이 클래스를 따로 나누지 않고 Time 클래스를 관리하는 TimeTable 클래스에 통합하였습니다.

3. 프로그램에서 제공하는 DB 외에도 사용자가 직접 강의를 등록하거나, 등록된 강의 정보를 변경하는 기능을 제공하기 위하여 따로 EditLecture 클래스를 추가하였습니다. 수정된 강의는 lecture 클래스 벡터에 추가되고, 나중에 DB에 저장되어 집니다. 또한 EditPrivate는 그 활용도가 적고, 초기에 requirement에도 없기에 기능을 삭제하였습니다.

Expansibility

설계 때 '재사용' 을 염두에 두고 설계를 했습니다. 비록 어떤 디자인패턴을 이용하여 재사용을 하도록 설계가 된 것은 아니지만, 클래스 하나 당 한 기능만 구현하는 OO 원칙을 따름으로써 일부 클래스들이 다른 프로그램을 작성하는데 재사용이 가능해졌습니다.

AddPrivate와 Private 그리고 TimeTable 클래스입니다. 데이터의 입력을 받고 저장하는 AddPrivate, 데이터를 가지고 있는 Private 클래스, 그리고 이 데이터를 화면에 출력하는 TimeTable 처럼 각각의 클래스가 각각의 기능만을 담당하고 있습니다.

이를 재사용하여 달력 프로그램을 제작할 수도 있습니다. AddPrivate에서는 기념일 이름과 날짜를 입력받고, Private에서는 이 정보를 저장하고, TimeTable에서는 GUI를 살짝 바꿔서 달력을 만들고, Private에 저장된 정보를 해당 일에 출력을 해줌으로써 시간표 프로그램에서 쓰이던 클래스들이 그대로 달력 프로그램에서도 재사용 될 수 있습니다.

또한 이 외에도 Lecture 클래스, addLecture 클래스를 각각 상속 받아서, 학교 강의 시간표가 아닌 언어교육원 혹은 학원, 아니면 회사에서 근무표 등으로도 재사용이 가능합니다.