

MetaData

Calendar Project 보고서

학번 : 2018008068

이름 : 김재형

Summary

The most Challenging Part

- Calendar UI 를 라이브러리로 가져와서 해당 라이브러리의 코드를 뜯어보면서 코드를 수정하는 것
- User Availability 기능을 구현하기 위해 SQL 쿼리문을 짜는 것
- Remind Message 알림을 위한 기능 구현
- Java Swing FrameWork 에 대한 Learning Curve
- 이벤트를 새롭게 생성한후에 기존 Swing 화면을 새로고침하는법(시간부족으로 해결 못함 π)

Interesting design decision that I made

- 달력옆의 초, 분, 시간 단위 UI 구현
- 이외에는 Calendar 과제 명세부분 구현하였습니다.

Major Issues that I ran into while designing, implementing, testing Program
Known problems with my implementation, such as missing functionality or bugs.

- Swing Framework 의 UI Component 학습 및 UI Component 에 대한 Layout 결정
- Table Schema 에 대한 디자인 결정
- Java Code 를 너무 오랜만에 만져보았기에 Java Code 자체에 대해서 익숙해지는데 시간이 조금더 걸렸던 것 같습니다.

Specification

실행환경

Vscode -> WSL remote // Window 11 // Ubuntu 20.04

프로그램 실행 방법

1. new_calendar/src/User/UserLogin.java 실행

라이브러리 참조

1. PostgreSQL driver 참조
2. Lib 폴더안의 jar 파일 참조

명령어

/usr/bin/env

/usr/lib/jvm/java-8-openjdk-amd64/bin/java

-cp /tmp/cp_6rjrwcfms8tvljxfi3rlsdjam.jar

User.UserLogin

라이브러리 참조 :

```
"java.project.referencedLibraries": [  
  "lib/**/*.jar",  
  "/usr/lib/jvm/java-1.8.0-openjdk-  
amd64/lib/postgresql-42.6.0.jar"  
],
```

만일 화면이 정상적으로 표시가 안될시,

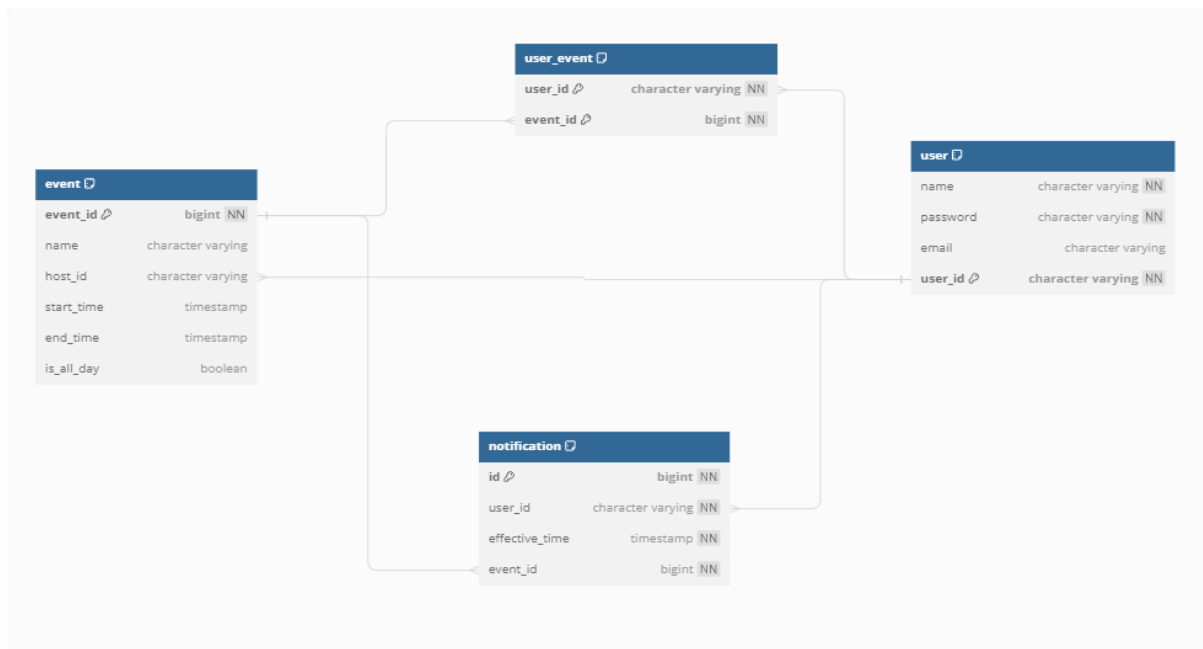
1. Export DISPLAY :=0 명령어 입력후 다시 실행하면 정상적으로 작동합니다.

ClassPath 참조 :

캘린더 구현부분

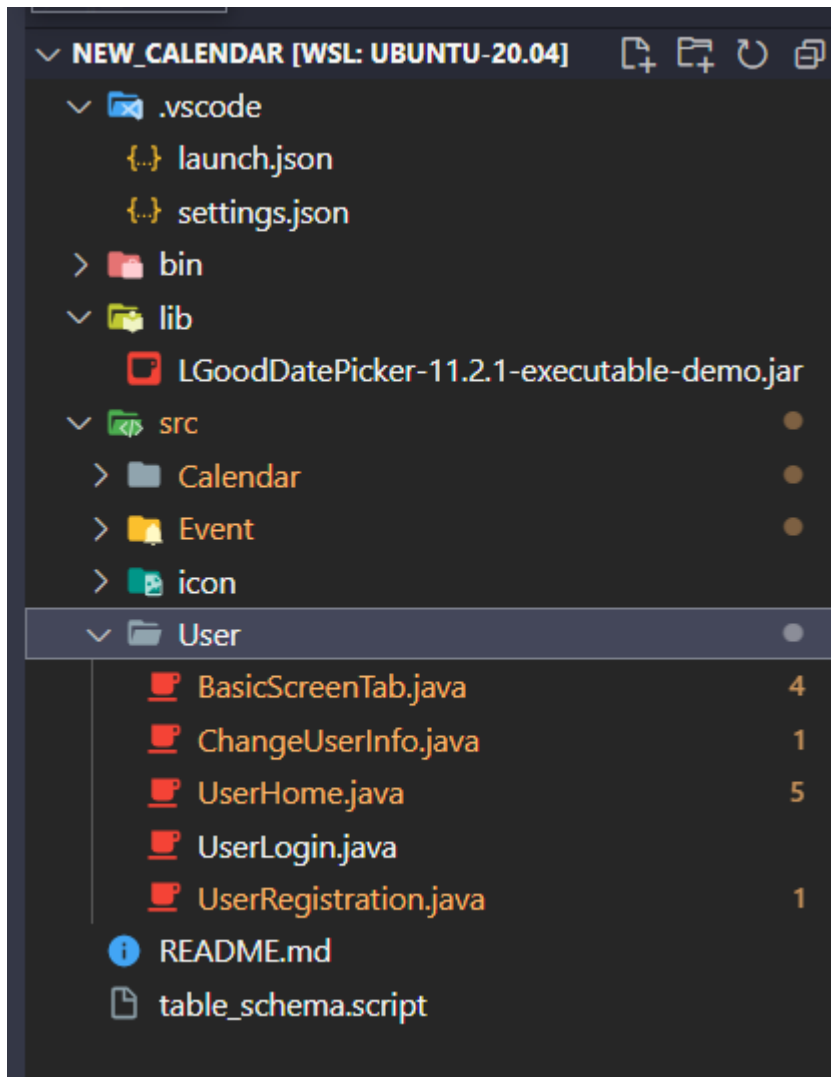
- Create a User Account
- Update a user account
- Authenticate a User
- Calendar View
- Event Reminder Setting
- Check Availability of family member
- Create an event
- Invite family members to the event
- View an event

테이블 스키마 구조



Design And Implementation

코드의 전체적인 부분에 대해서 설명해드리도록 하겠습니다.



.vscode 폴더 : 디버깅 옵션 설정

Settings.json 에 postgresql 드라이버 및 Lib 폴더 참조를 선언하였습니다.

```
{  
  "java.project.sourcePaths": [  
    "src"  
  ],  
  "java.project.outputPath": "bin",  
  "java.project.referencedLibraries": [  
    "lib/**/*.jar",  
  ]  
}
```

```

    "/usr/lib/jvm/java-1.8.0-openjdk-
amd64/lib/postgresql-42.6.0.jar"
  ],
  "java.project.classpath": ":/usr/lib/jvm/java-1.8.0-
openjdk-amd64/lib/postgresql-42.6.0.jar",
  "java.project.encoding":""
}

```

Lib 폴더 : 외부 라이브러리 파일 삽입

The screenshot shows a web form titled "Create Event" with the following fields:

- Event Name**: example_event
- Start Time**: December 8, 2023, 5:00am
- End Time**: December 8, 2023, [empty]
- Is All Day?**: [checkbox]

Handwritten annotations in red:

- "시간 (5분 단위)" (Time in 5-minute units) next to the Start Time field.
- "날짜" (Date) next to the End Time field.
- A red circle around the calendar picker for the End Time field, with an arrow pointing to it and the word "picker" written in red.
- A box on the right containing the text "invite 가능 목록" (Inviteable list) with a red arrow pointing to it.

Calendar 의 start_time 및 endtime 을 input 으로 받을때 시간을 조금 더 편리하게 입력받기위해서 외부 라이브러리를 다운받아서 사용하였습니다.

Src 폴더 : 실제 코드 구현부분

Calendar : calendar 의 UI 부분과 관련한 폴더

Event : Event 의 생성 과 관련 한 폴더

User : User 의 생성과 관련한 폴더

여기에 시작 프로그램인 **UserLogin.java** 가 있습니다.

Calendar UI 구현 원리

UI 사용의 경우 외부 라이브러리 : <https://github.com/DJ-Raven/java-swing-calendar>

를 사용하였으나 해당 코드를 수정하여

UI 에 event 를 표시하는 부분은 새롭게 코드로 작성하였습니다.

각 Calendar 의 button 은 달력의 날짜에 해당하며 ,

User 의 event 를 보며 각 user 의 event 가

달력 날짜에 해당하는 부분인지 확인을 하며

Background 에서매 시간마다 초, 그리고 Event Notification 을 검사합니다.

확인 코드

```
42 .....showMonthYear();
43 .....new Thread(new Runnable() {
44 .....    @Override
45 .....    public void run() {
46 .....        while (true) {
47 .....            try {
48 .....                Thread.sleep(1000);
49 > .....            } catch (InterruptedException e) {...
52 .....                Date date = new Date();
53 .....                Timestamp currentTimeStamp = new Timestamp(date.getTime());
54 .....
55 .....                SimpleDateFormat tf = new SimpleDateFormat("h:mm:ss aa");
56 .....                SimpleDateFormat df = new SimpleDateFormat("EEEE, dd/MM/yyyy");
57 .....                String time = tf.format(date);
58 .....
59 .....                Integer nowMinute = Integer.parseInt(time.split(":")[1]);
60 .....                Integer nowSecond = Integer.parseInt(time.split(":")[2].split(" ")[0]);
61 .....                // System.out.println(nowMinute + " " + nowSecond);
62 > .....                if(nowMinute % 5 == 0 && nowSecond == 0){...
148 .....
149 .....                lbTime.setText(time.split(" ")[0]);
150 .....                lbType.setText(time.split(" ")[1]);
151 .....                lbDate.setText(df.format(date));
152 .....            }
153 .....        }
154 .....    }).start();
155 }
```

Background 에서 실행하는 코드는 다음과 같습니다.

매 5 분마다 확인하여 event notification 을 하는지 확인합니다.

Calendar 앱 메인 화면 구현 원리

```
void createTabbedPane(String UserId, String UserSession) {  
    JTabbedPane tPane = new JTabbedPane();  
    add(tPane);  
  
    UserHome mainPanel = new UserHome(UserId, UserSession);  
    tPane.addTab("Calendar", mainPanel);  
  
    CheckAvailability checkScreen = new CheckAvailability(UserId, UserSession);  
    tPane.addTab("Check Availability", checkScreen);  
  
    EventMainScreen eventPanel = new EventMainScreen(UserId, UserSession);  
    tPane.addTab("Events", eventPanel);  
  
    // JPanel reportPanel = new JPanel();  
    // reportPanel.add(reportLabel);  
    // tPane.addTab("Check Availability", reportPanel);  
}
```

1 2 3

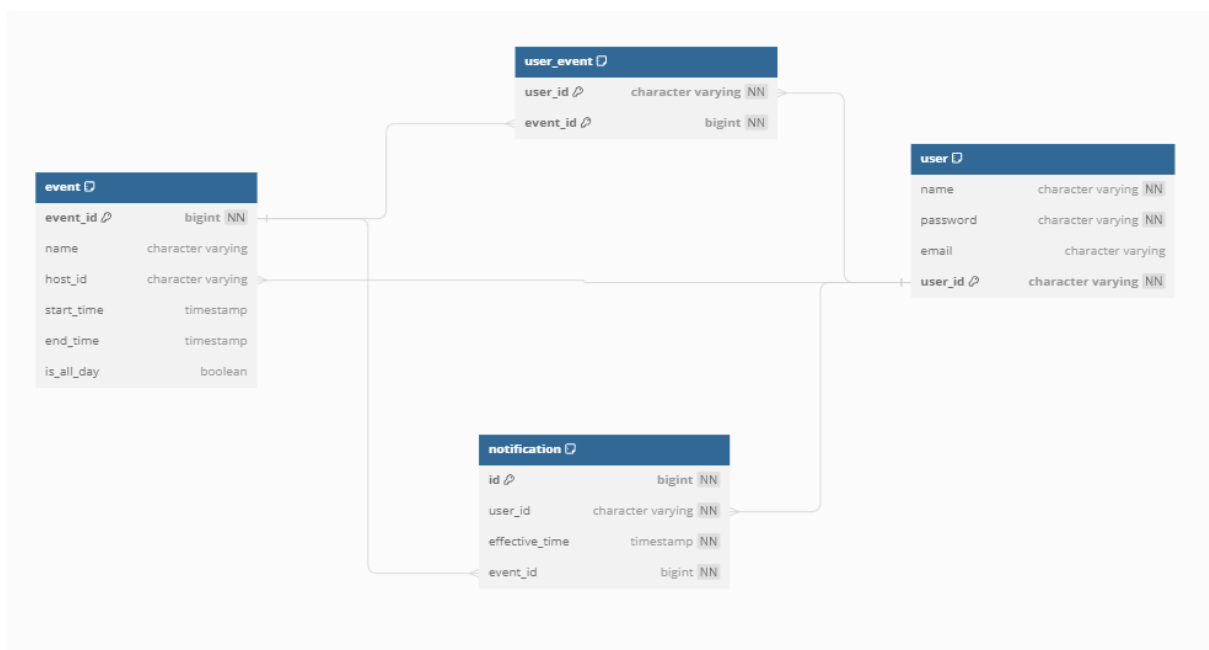
SRC 의 User/BasicScreenTab 폴더를 확인시

다음과 같이 Login 성공시 기본화면을 볼수있으며

MainPanel 에서는 Calendar 의 기본화면

Checkavailability Panel 에서는 각 User 의 event 시간내 이용가능 여부

EventMain Screen 에서는 Event 의 Create 을 생성하는 것을 확인할수 있습니다.



테이블 설계 디자인원리

User – Event : N;N 관계이므로

User-event 를 연결시키는 User_Event 테이블을 새롭게 설계하였습니다.

또한 notification 테이블 설계를 위해

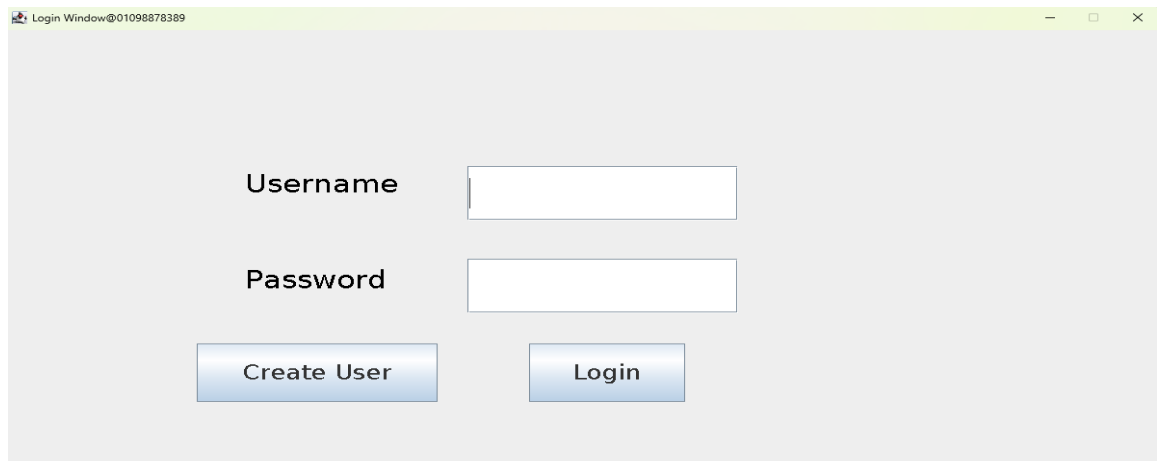
User : notification 은 1:N 관계를 만족하도록 설계하였으며 ,

Event: notification 은 1:N 관계를 만족하도록 설계하였습니다.

Testing

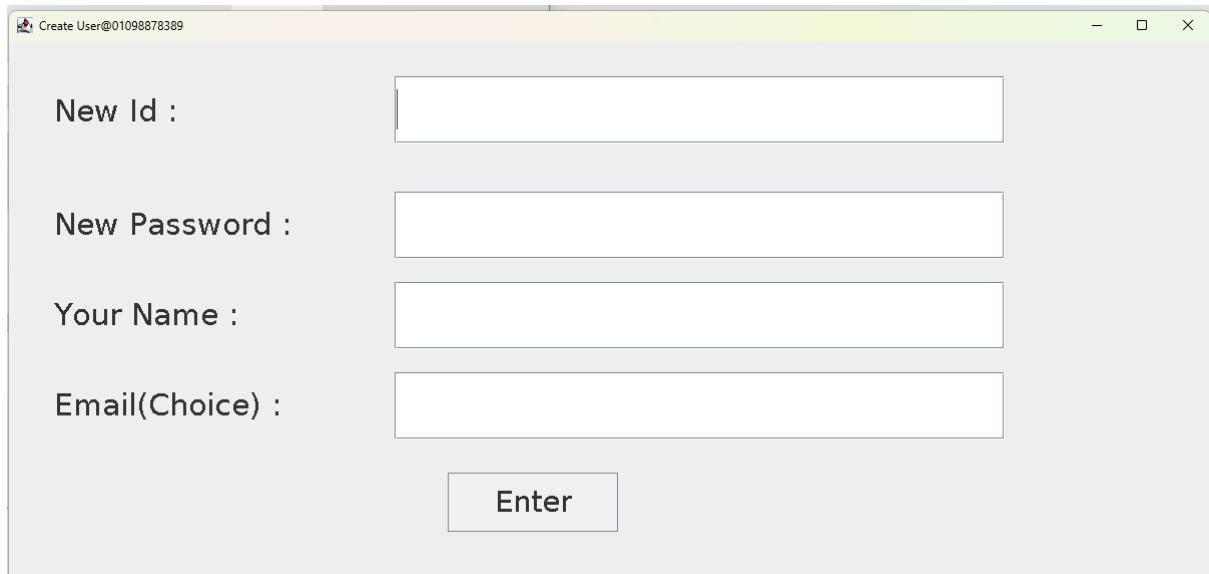
1. 로그인 초기화면

User 폴더의 UserLogin.java 파일을 실행할시 다음과 같은 화면이 나오게 됩니다.



2. User Account 생성화면

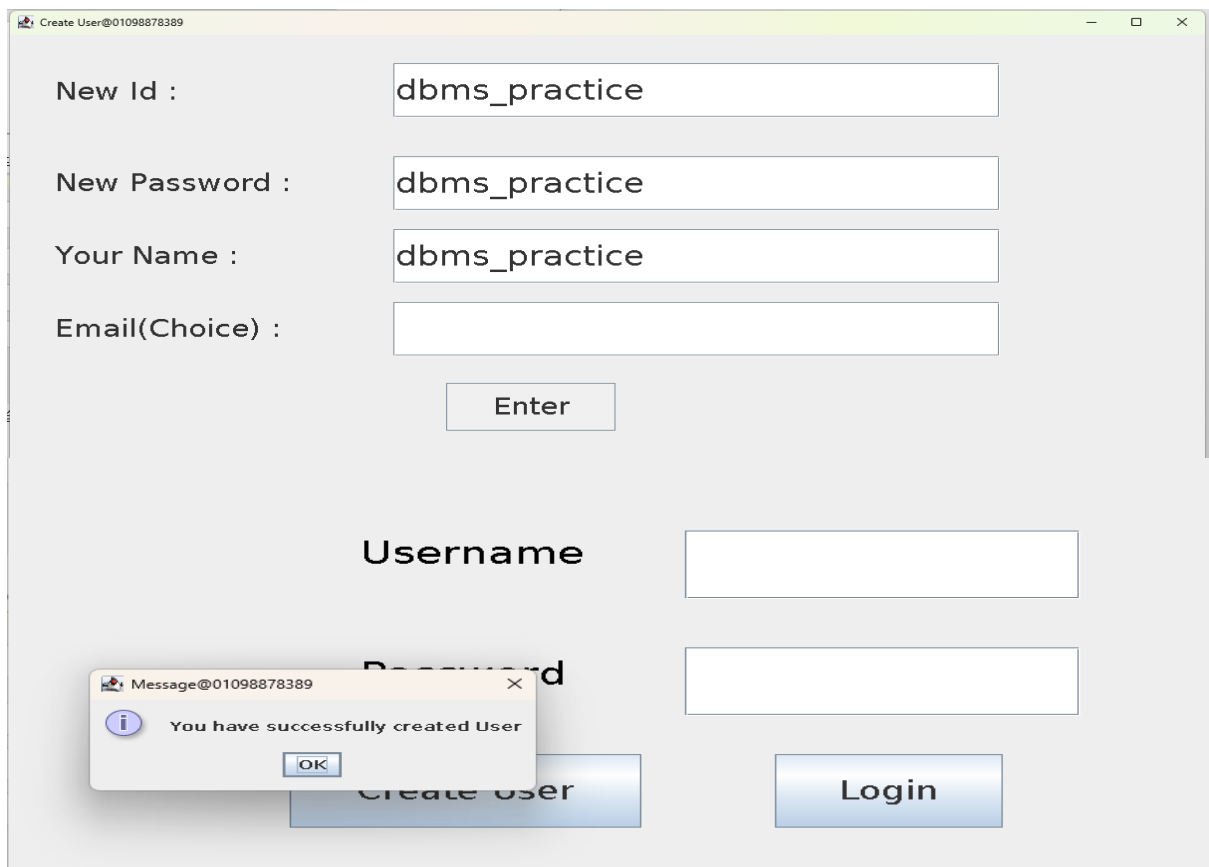
Create User Button 클릭시 다음과 같은 화면이 표시되게 됩니다.



The screenshot shows a window titled "Create User@01098878389". It contains four input fields labeled "New Id :", "New Password :", "Your Name :", and "Email(Choice) :". Below these fields is a button labeled "Enter".

여기에서 예시로 User 하나를 생성해보도록 하겠습니다.

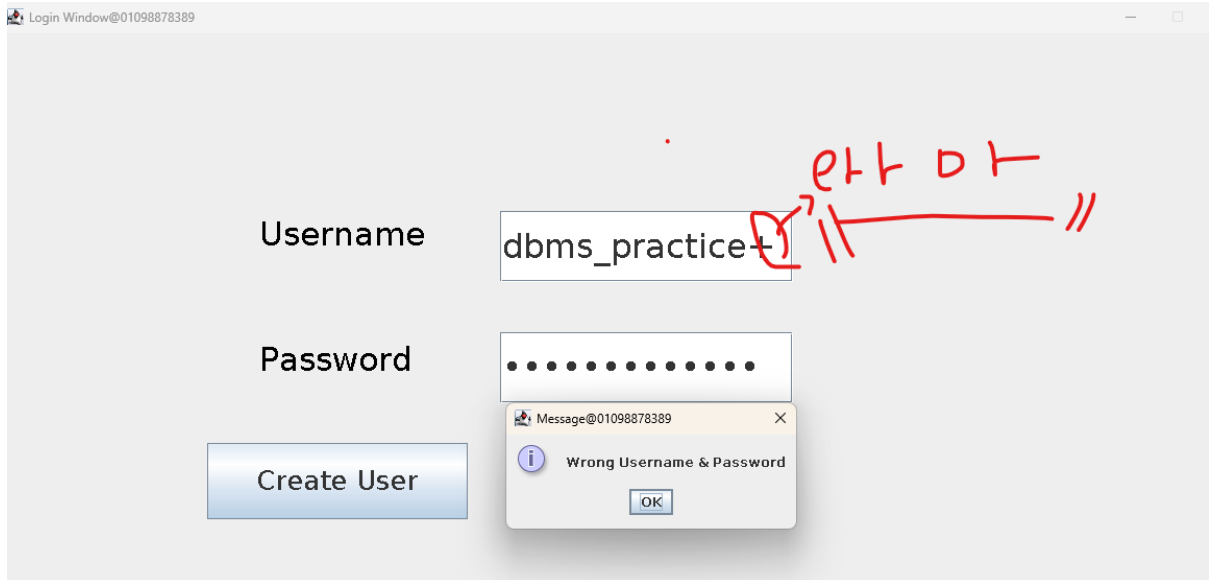
다음과 같이 User 입력후 Enter 버튼 클릭시 User 생성이 완료됩니다.



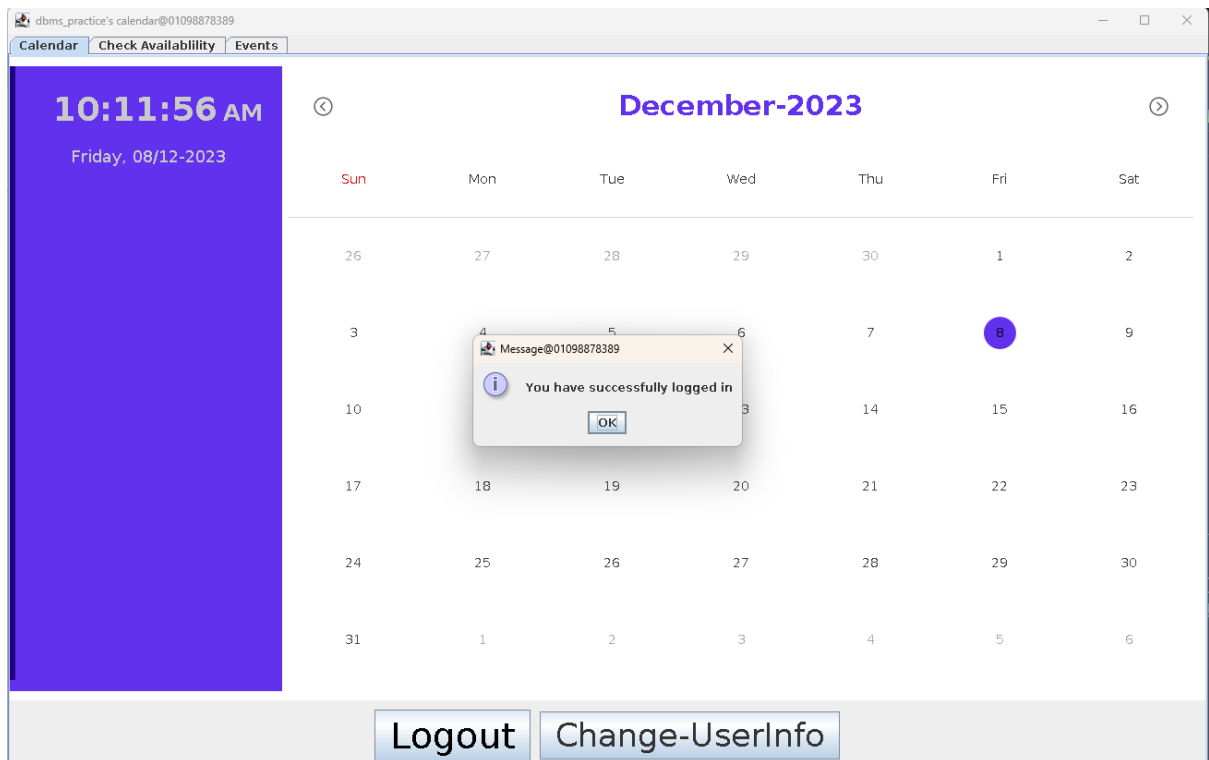
The screenshot shows the same "Create User" window, but now the input fields are filled with the text "dbms_practice". The "Email(Choice) :" field is empty. Below the "Enter" button, there are additional fields for "Username" and "Password", and a "Login" button. A small message box is overlaid on the bottom left, displaying the text "You have successfully created User" with an "OK" button.

이후 입력한 정보를 토대로 User 정보를 입력할시 정상적으로 로그인되는 것 확인할수 있습니다.

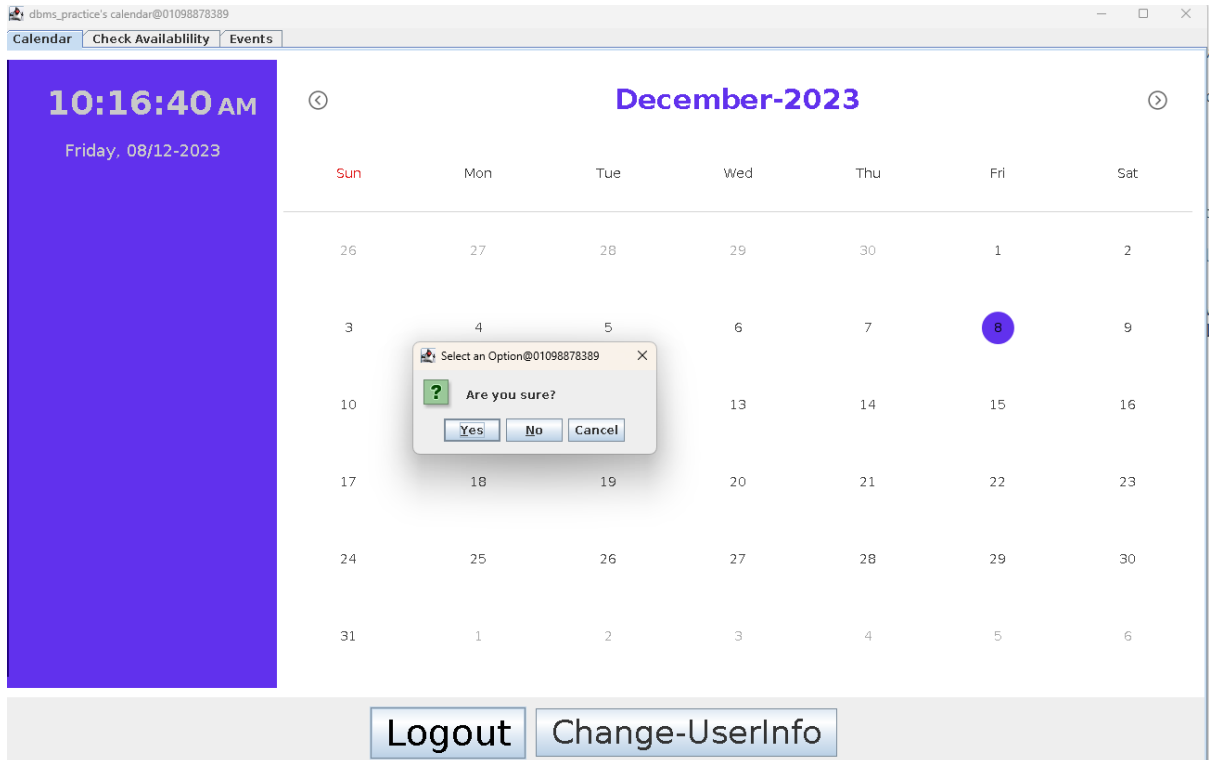
만일 로그인을 ID, 비밀번호와 다르게 할시 에러메세지가 뜨는 것을 확인할수있습니다.



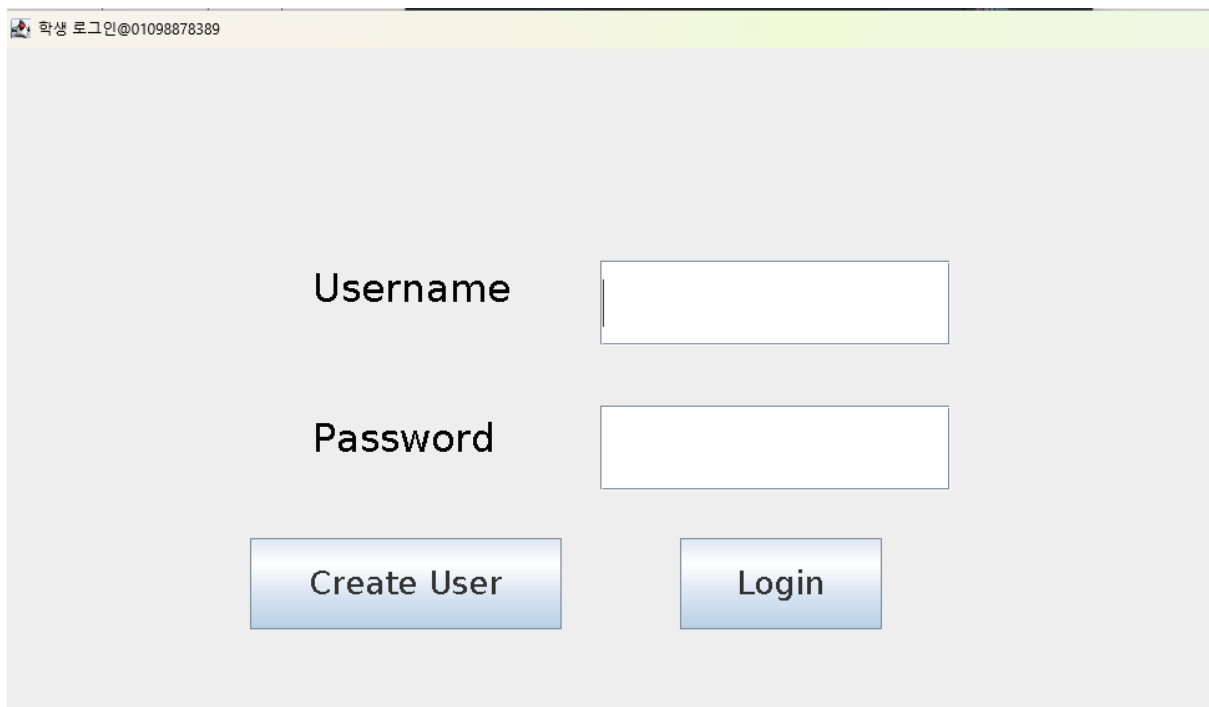
하지만 정상적으로 자신이 생성한 User ID 와 비밀번호를 입력할시 로그인도 정상적으로 되면서 Calendar View 가 나온 것을 확인할수 있습니다.



정상적으로 현재시간을 표시하는 달력이 표시되는 것을 확인할수 있으며, Calendar Tab, Check Availability Tab, Delete Tab 이 있는 것을 확인할수 있습니다. 만일 여기서 User LogOut 버튼을 클릭한다면



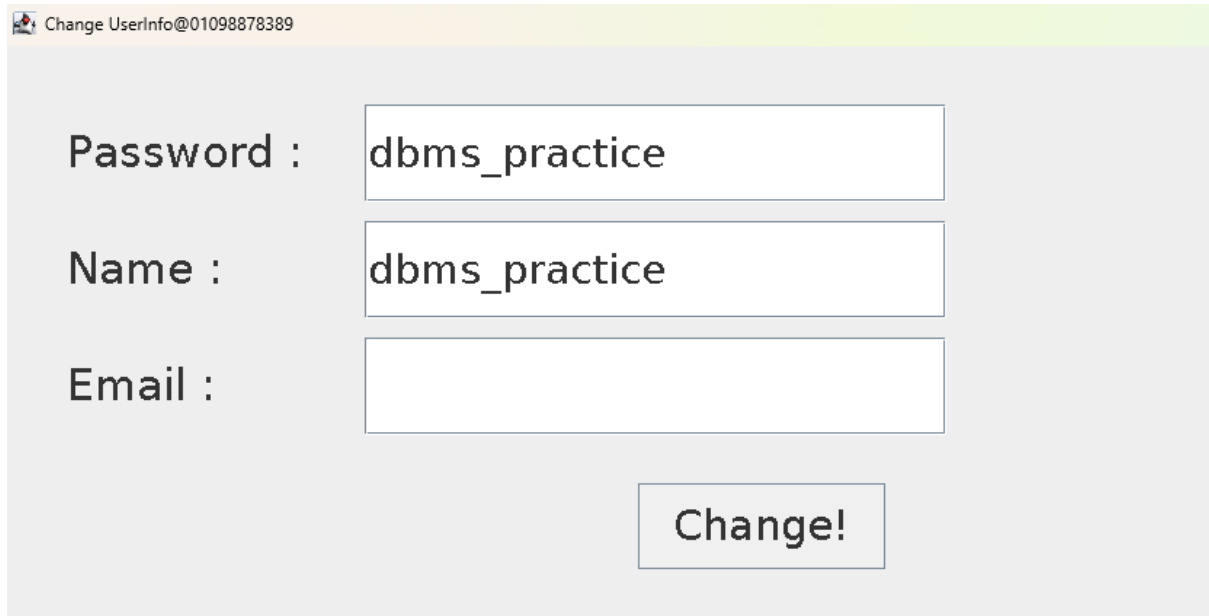
다음과 같은 팝업메세지를 확인할수 있으며 Yes 클릭시 실제 LogOut 되는것을 확인할수 있다.



로그아웃 버튼 클릭후 다시 로그인 첫 화면창으로 돌아간 것

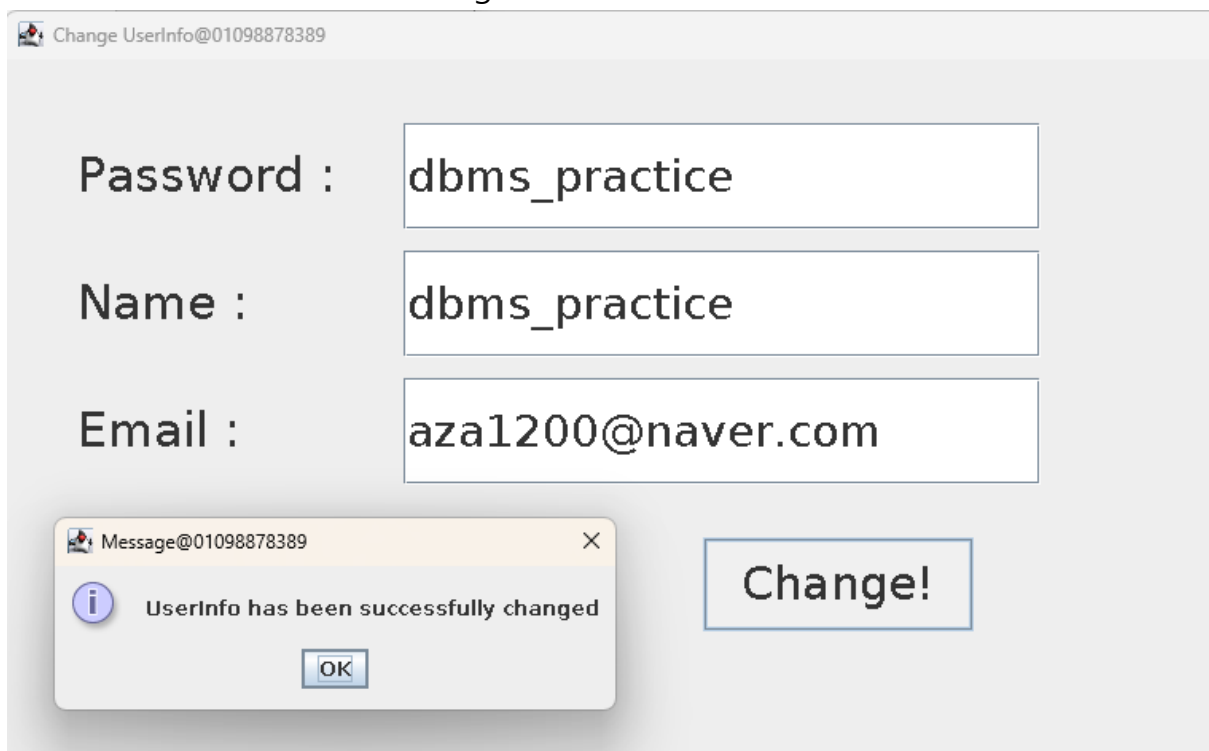
3. User Update

다시 본론으로 돌아와서 User 부분으로 정상적으로 Login 한후에 Change-UserInfo 버튼을 클릭할시 ID 를 제외한 password, name, email 부분은 처음에 입력했던대로 뜨는 것을 확인할수있다.



The screenshot shows a window titled "Change UserInfo@01098878389". It contains three input fields: "Password :" with the value "dbms_practice", "Name :" with the value "dbms_practice", and "Email :" which is empty. Below the fields is a button labeled "Change!".

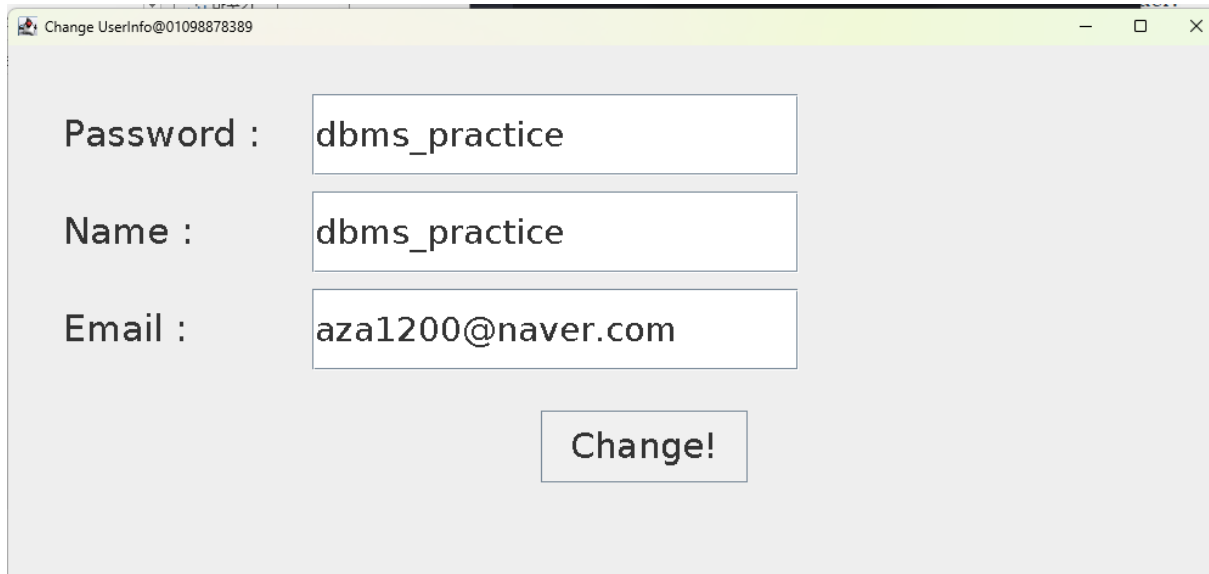
ID 는 User 를 판별하는 고유값이기에 따로 수정을 할수없도록 설정했다 여기에 email 을 입력하고 change 버튼을 클릭해보면



The screenshot shows the same "Change UserInfo@01098878389" window, but now the "Email :" field contains "aza1200@naver.com". A message dialog box titled "Message@01098878389" is overlaid on the bottom left, displaying an information icon, the text "UserInfo has been successfully changed", and an "OK" button. The "Change!" button is still visible on the right.

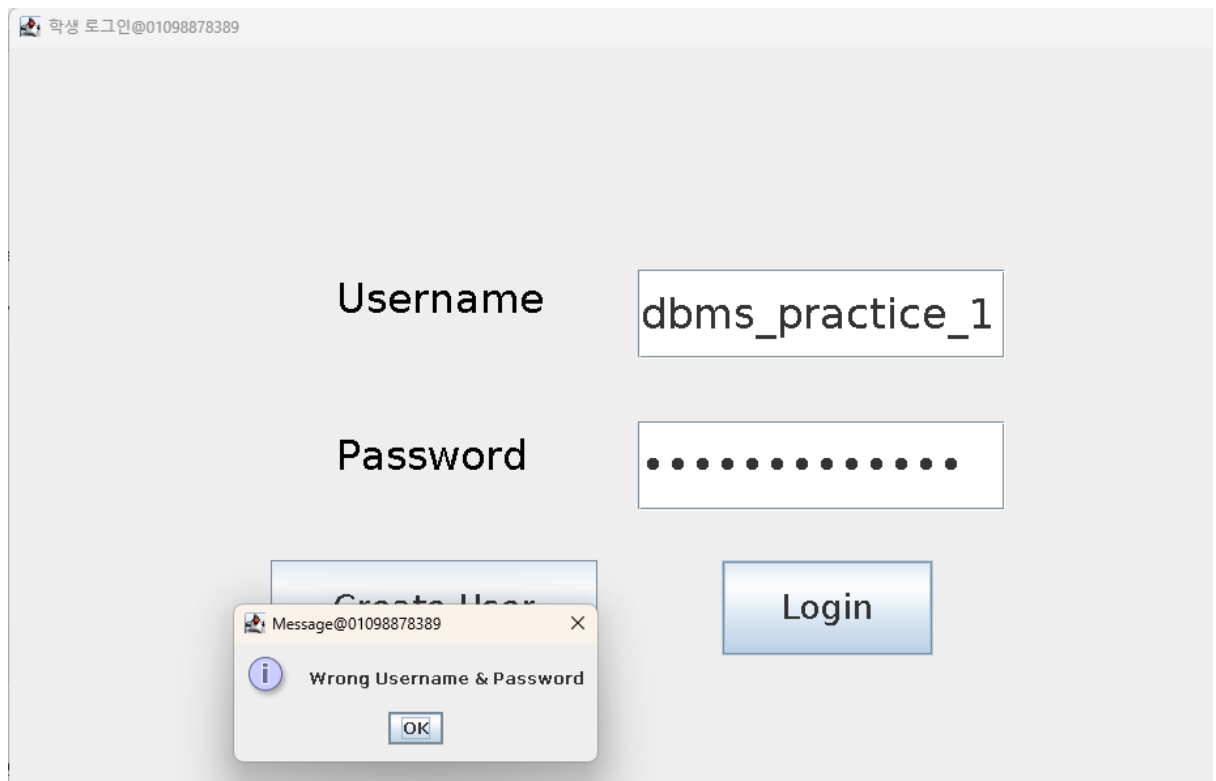
Change 가 완료되었다는 것을 확인할수 있으며 ,

이후 Change-UserInfo 버튼을 클릭하여 페이지 확인지 , 처음에 email 을 create 한부분이 새롭게 생성하는 것 확인할수있다.

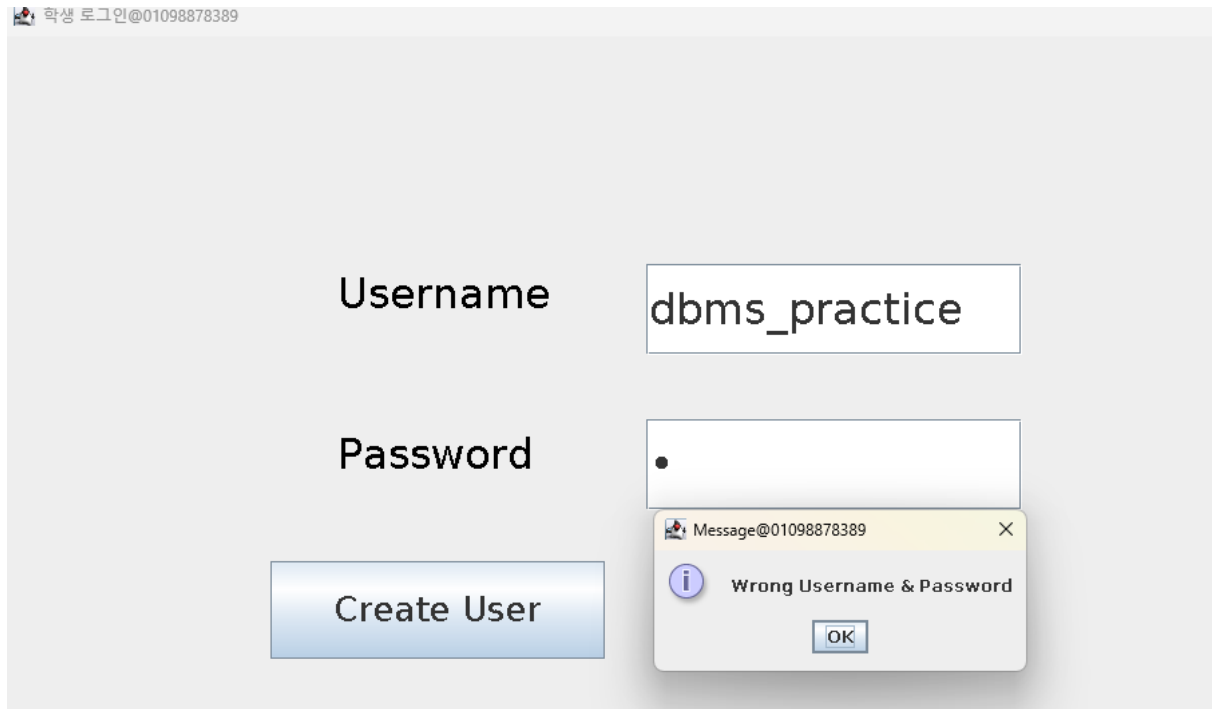


4. Authenticate User 부분

User 의 로그인, 구현 및 ID, Password 확인부분은 Login 부분에 설명하였으니 추가적인 설명은 생략하도록 하겠습니다.

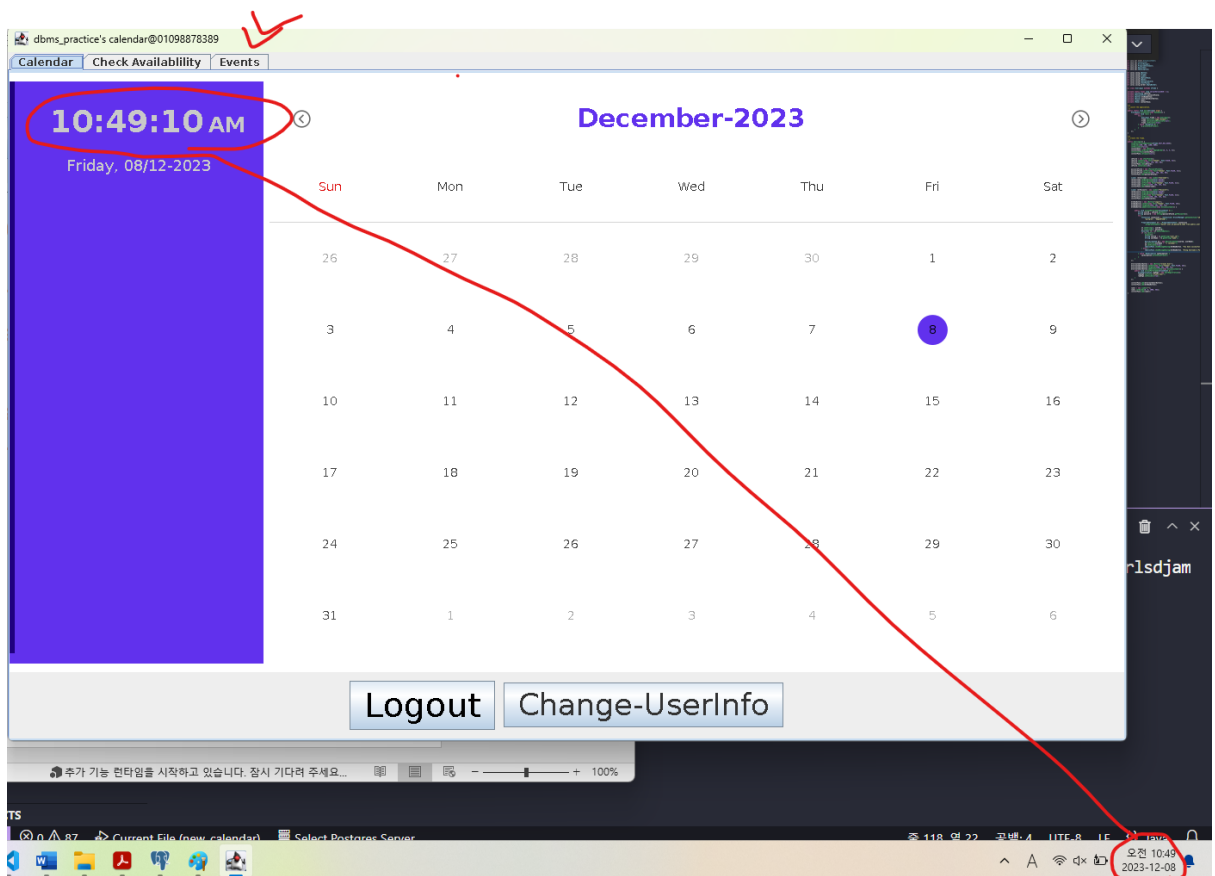


Wrong ID 를 입력할시 화면 모습



Wrong Password 입력할시 화면의 모습

5. Calendar View 구현

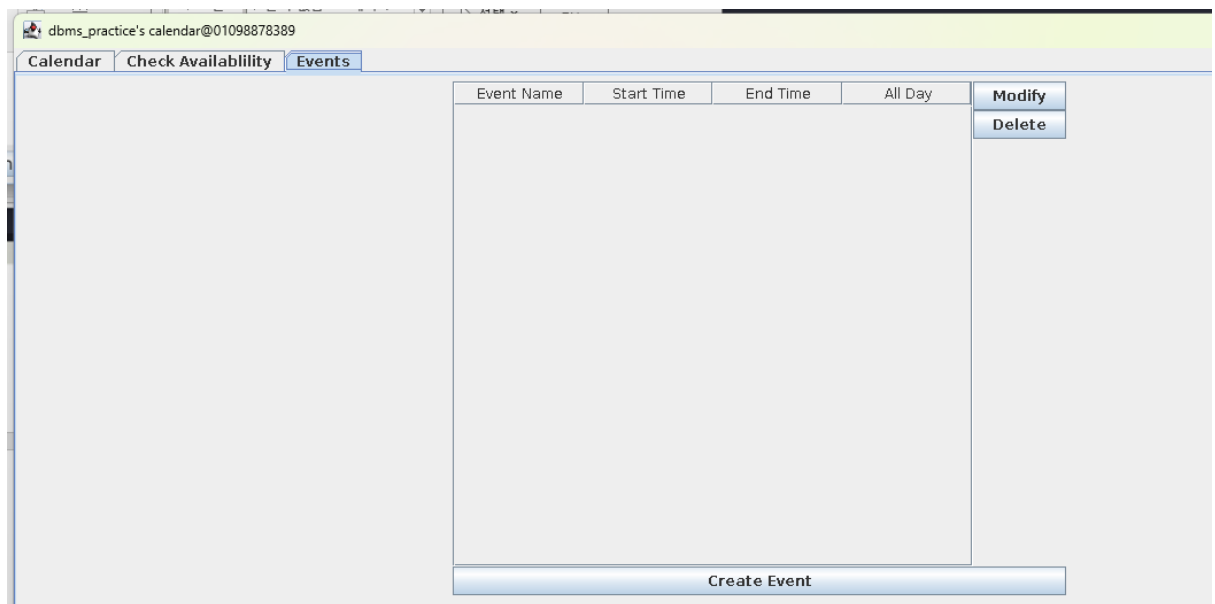


처음에 로그인시 , Calendar View 가 표시되는 데 현재 달력의 날짜와 시간 ,초가 실시간으로 표시되는 것 확인할수 있다.

달력에 현재 event 가 표시되어야하는데

Dbms_practice 는 새로 생성된 계정이므로 이벤트가 현재 없는 것을 확인할수 있습니다.

Event 가 Calendar 에 표시가 되지 않기에 새롭게 이벤트를 생성한 후 Calendar View 를 보도록 하겠습니다.



Event tab 클릭시 event tab 화면

Create event 클릭시 event 생성화면

A screenshot of the 'Create Event' form in the calendar application. The form includes the following fields and controls: 'Event Name' (text input), 'Start Time' (time picker), 'End Time' (time picker), 'Is All Day?' (checkbox), 'Invite Users' (checkbox), and a 'Create Event' button. A list of users is displayed on the right side of the form, including 'kimjaehyeong', 'jhkim2', 'jhkim_test', and 'dbms_practice'.

Event Name: example_event

Start Time: December 8, 2023 5:00am

End Time: December 8, 2023

Is All Day?

Calendar: December 2023

Users: kimjaehyeong, jhkim2, jhkim_test, dbms_practice

Handwritten notes: 시간 (5분 단위), 날짜, invite 가능 목록

Create Event

다음과같이 폼을 입력하고 총 3 종류의 이벤트를 생성할계획이다.

1. Event 의 시작시간과 끝시간이 하루만에 끝나는이벤트(12 월 8 ~12 월 8)

2. Event 의 시간시간과 끝시간이 여러날짜에 걸칠때 (12 월 10~12 월 12)

총 2 개의 event 를 생성하고 달력에 정상적으로 표시되는지 확인해볼예정이다. (일단 다른 User invite 는 추후에 하도록 한다.) (현재는 event 값이 calendar 에 잘 표시되는지 확인하기 위한 목적이므로)

(이벤트 생성 시간 DateTimePicker 를 클릭할 때 클릭이 잘안될시, 시간값을 여러 번 클릭하거나 , 혹은 다른버튼을 클릭후 날짜 혹은 시간 버튼을 클릭해야 잘 클릭됩니다!

Event Name: example_event

Start Time: December 8, 2023 5:00am

End Time: December 8, 2023

Is All Day?

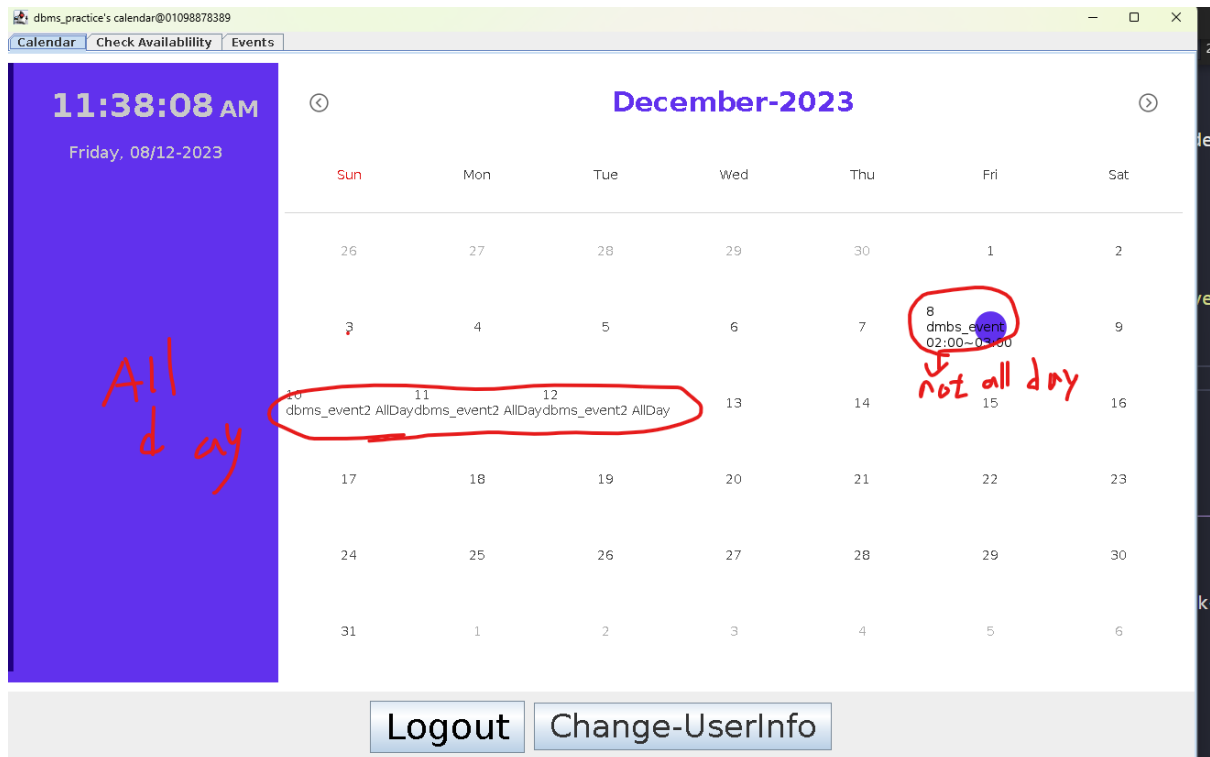
Confirmation Dialog: Really wanna Create Event? (Yes, No, Cancel)

Create Event

Create 이벤트 버튼 클릭시 다음과같이 팝업메세지 나오고

(로그아웃 후에 다시 로그인시..정상적으로 이벤트가 표시된 것 확인할수 있습니다.)

원래는 event 삽입후 새로고침된 것을 구현하려 했으나 시간부족 이슈로 구현하지는 못했습니다 π



두가지 이벤트가 정상적으로 생성된 것 확인할수있습니다.

1. 12 월 8 일 날짜로 이벤트 생성 (당일 이벤트 : not all day)
2. 12 월 10 일~ 12 월 12 일날짜로 이벤트 생성

여러날짜에 걸쳐서 이벤트가 생성이 될시 달력에 여러날짜가 걸쳐서 시간이 표시된 것을 확인할수 있습니다.

6. Event Reminder Setting

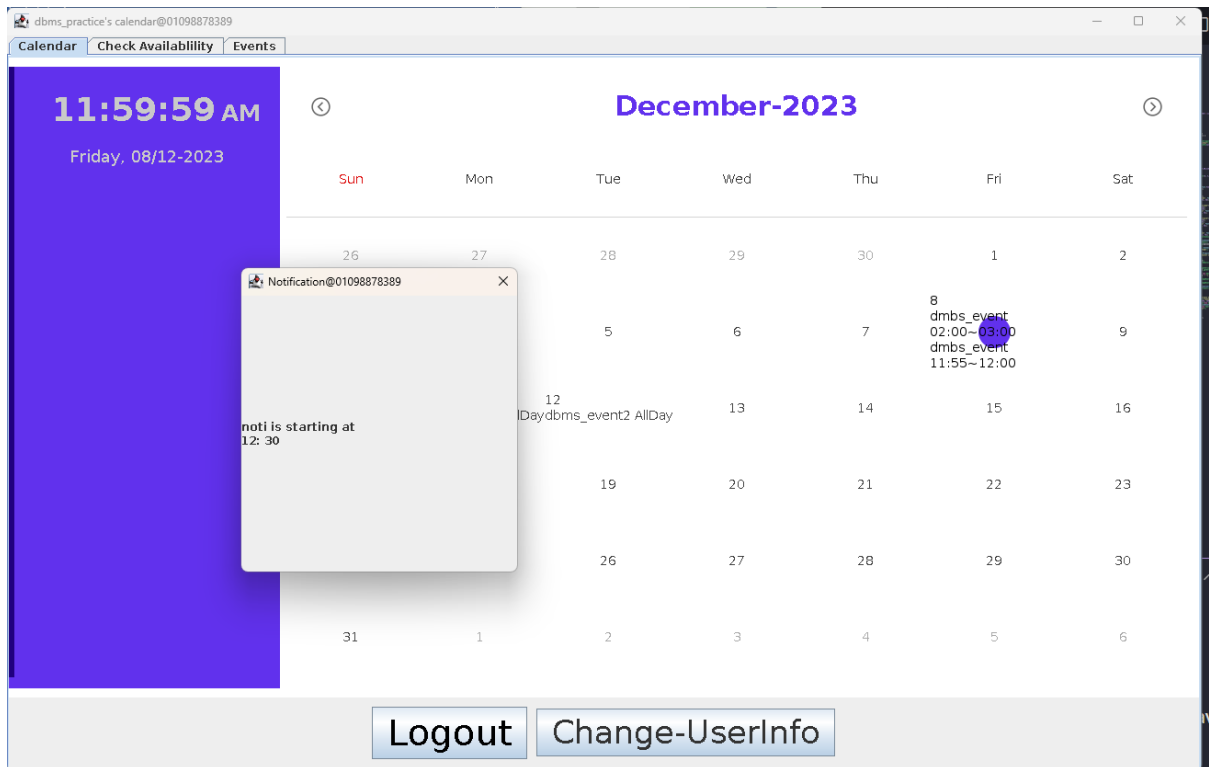
이벤트를 생성시,

자동으로 이벤트 시작시간 30 분전에

해당 User 에게 Notification 이 일어나도록 설정이 되어있습니다.

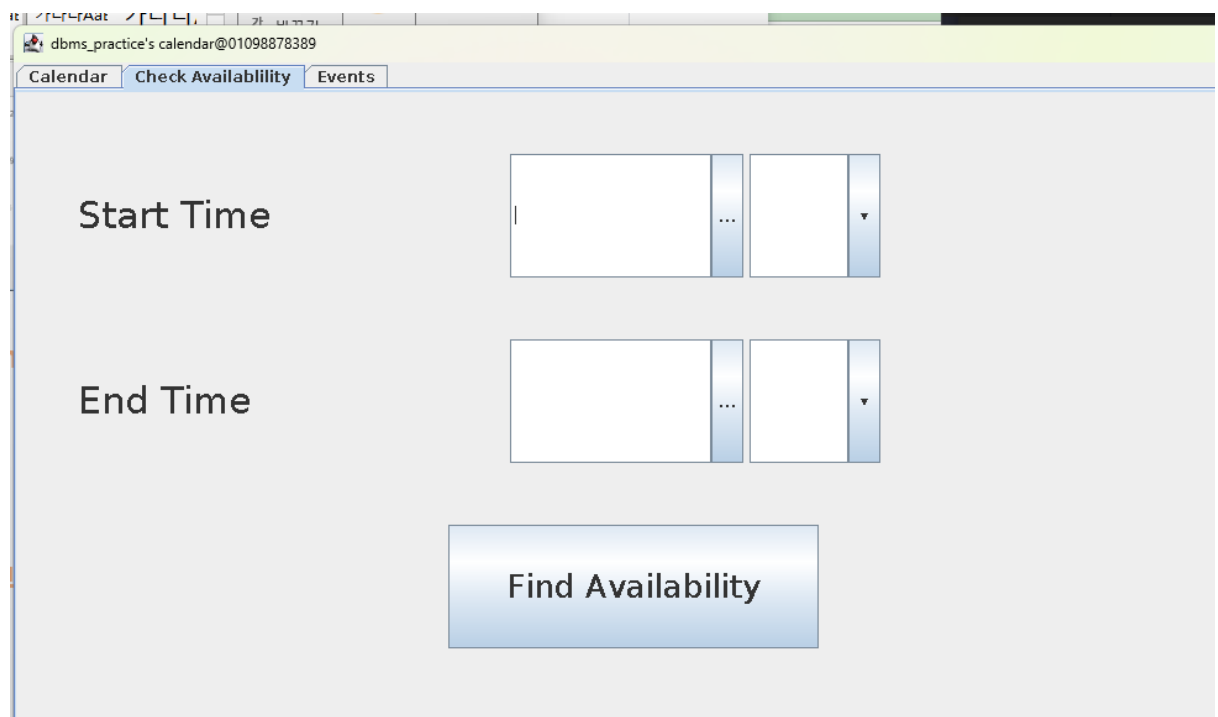
즉 이벤트가 일어나는 시간까지 Calendar 화면을 열고있을시 다음과 같이 Notification 이 일어나는 것을 확인할수 있습니다.

Event Notification 설정 실행 화면 캡처



다음과 같이 팝업 메시지가 뜨는 것을 확인할수 있습니다.

7. Check Availability of family member 부분 구현



Family Tab 에 들어갈시 다음과같이 input 시간 과 output 시간을 입력할시 해당 시간에 family 들이 이용이 가능한지 구현이 되어있습니다.

Check availability 쿼리코드 구현 및 설명

```
PreparedStatement st = (PreparedStatement) connection.prepareStatement(
    "select u.name, case when " +
    "( " +
    "select count(*) " +
    "from public.event e left join public.user_event ue " +
    "on e.event_id = ue.event_id " +
    "where " +
    "( " +
    "(start_time <= ? and ? <= end_time) " +
    "or " +
    "(start_time <= ? and ? <= end_time) " +
    "or " +
    "(? <=start_time and end_time <= ?) " +
    ") and ue.user_id = u.user_id " +
    ") > 0 then false else 'true' end as is_possible " +
    "from public.user u " +
    "where u.user_id <> ?"
);

st.setTimestamp(1, startTimestamp);
st.setTimestamp(2, startTimestamp);
st.setTimestamp(3, endTimeTimestamp);
st.setTimestamp(4, endTimeTimestamp);
st.setTimestamp(5, startTimestamp);
st.setTimestamp(6, endTimeTimestamp);
st.setString(7, UserId);
```

각 User 별로 쿼리를 돌면서

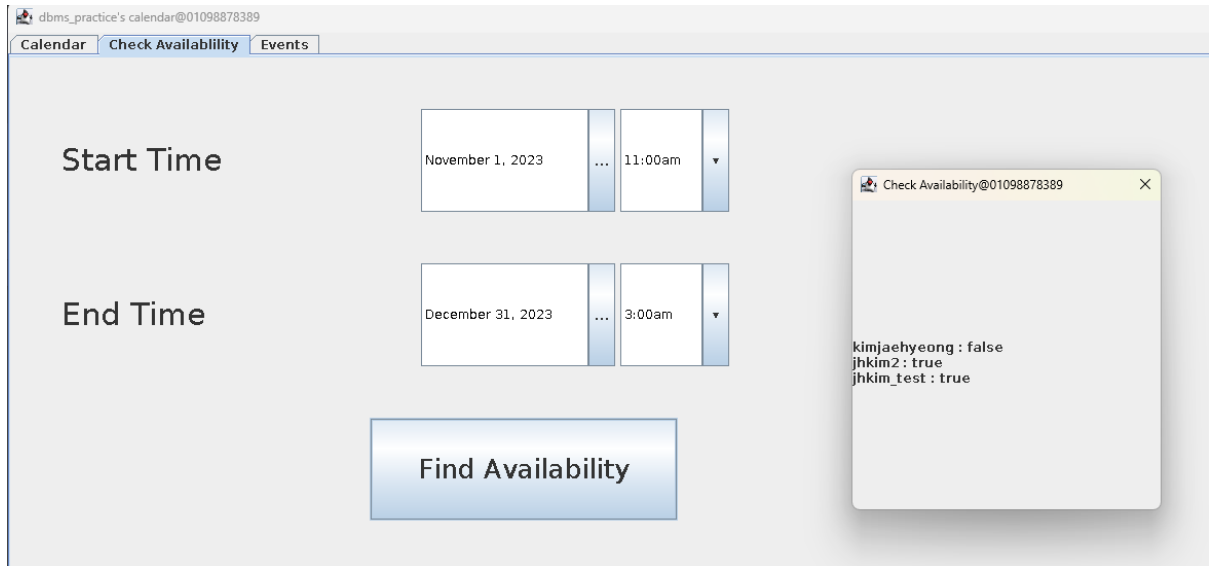
해당 User 가 가지고 있는 이벤트가 입력 시작 이벤트와 입력 end event 시간 사이에 있을시 false 아닐시 true 를 리턴하도록 구현되어있습니다.

Event 시간이 겹치는 조건

1. Input 의 시작시간이 user 가 가지고 있는 event 의 시작시간의 끝시간 사이에 있을 때
2. Input 의 이벤트 끝시간이 user 가 가지고 있는 event 의 시작시간과 끝시간 사이에 있을 때,

3. Input 이벤트 시간의 시작시간과 끝시간 모두 user event 자체를 포함하고 있을 때 세가지 경우일 때 false 를 return 하고 나머지 경우 true 를 리턴하도록 설정하였습니다.

구현결과 캡처



8. Event Create , Invite Family , Calendar View 부분 구현

@01098878389

Event Name

Start Time ...

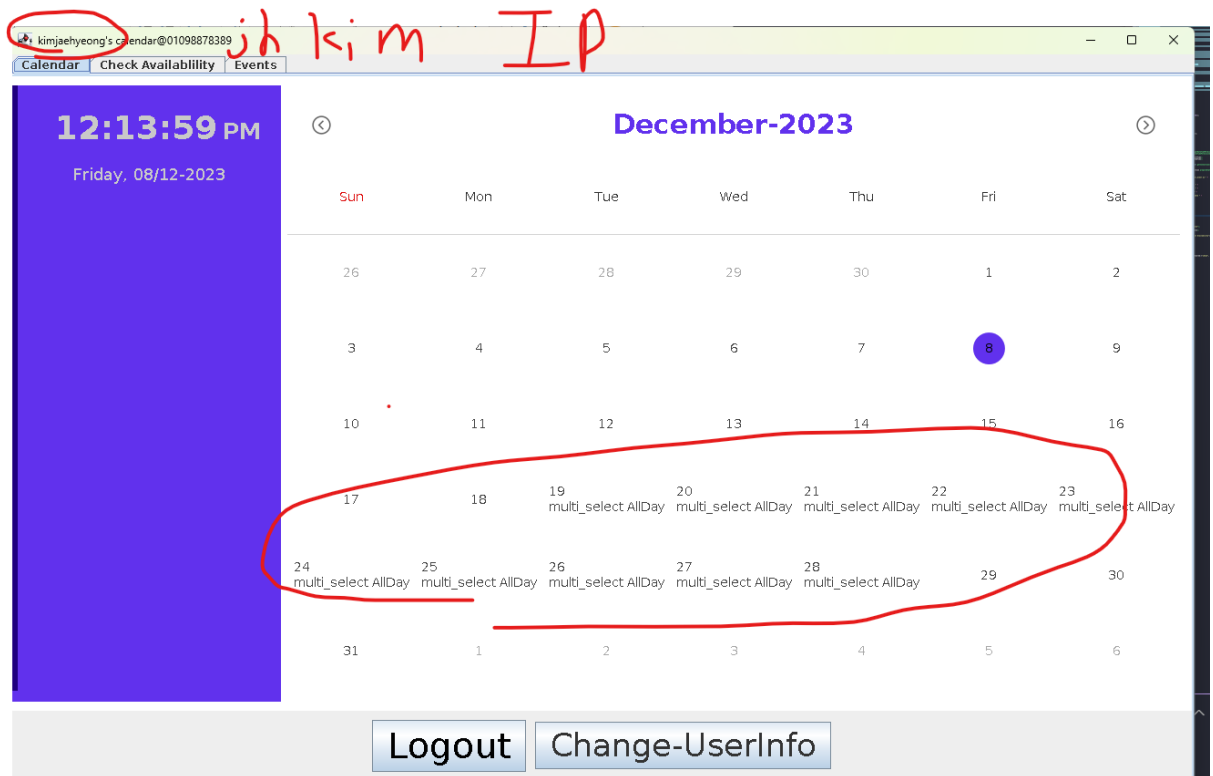
End Time ...

Is All Day? ☒ Invite Users

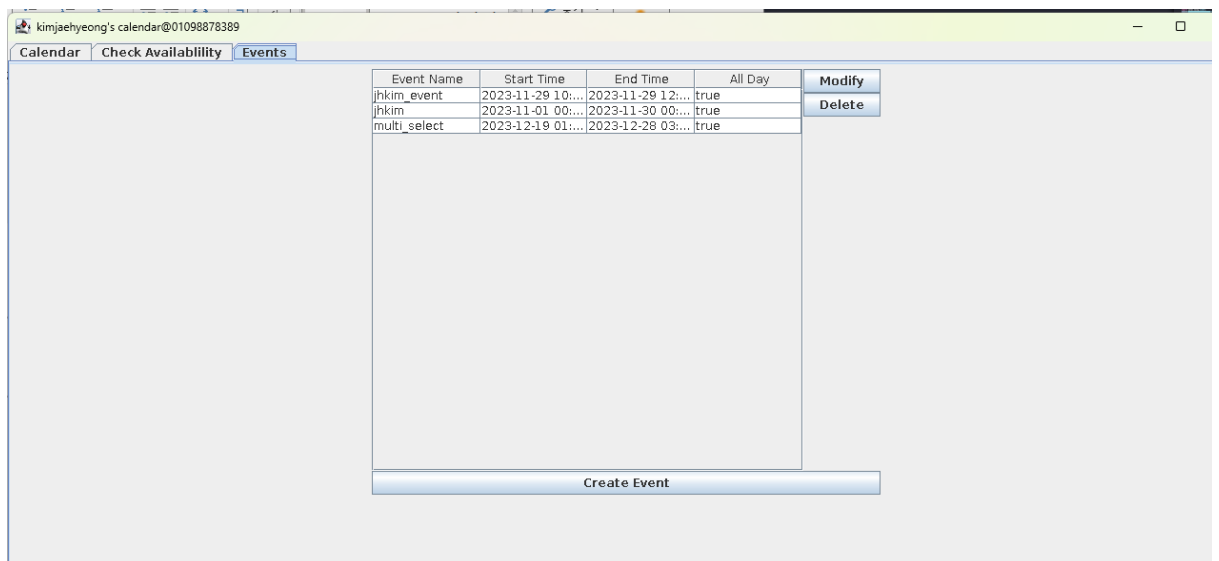
kimjaehyeong
jhkim2
jhkim_test
dbms_practice

다음 그림과 같이 User 모두를 Multiselect 하고 다른 user 에서 event 가 정상적으로 invite 되었으며 event 가 생성이 되었는지 확인하도록 하겠습니다.

Event create 부분은 Calendar event view 에서 설명하였으므로 생략하도록 하겠습니다.



다음과 같이 이벤트가 다른 User 에서 로그인하였을 시 정상적으로 생성되었으며, Invite 가 정상적으로 된 것 확인할수 있습니다.



JHKIM 이라는 User 로 로그인하였을시, 해당 user 는 event 를 가지고 있는 것 정상적으로 확인할수 있습니다.

known Problems

1. Event 의 modify, delete 부분을 구현하지 못하였습니다.
2. User 의 RSVP 부분을 구현하지 못했습니다.
3. Event Create 시 ,기존 달력의 UI 가 새로고침 되어있어야 하는데 해당 UI 부분을 새로고침하는부분을 구현하지 못하여 , 새로고침된 event list 를 보고싶을시 , Logout 후 다시 login 하여야 합니다.
4. Calendar 의 event 를 UI 에 표시하는 부분의 시간복잡도가 (n^2) 이기에 calendar 의 실행 속도가 상당히 느리다고 볼수있습니다.

각 달력 (1~30 일) 사이의 component 마다 해당 User 가 가지고 있는 모든 eventlist 를 돌면서 해당 event 가 각 component 에 해당되는지 check 하기에 비교적 시간이 오래걸립니다. 해당 문제에 대해서 정확히 해결하지 못한것같습니다.

Comments

가장 큰 문제는 시간이 부족했던 것 같습니다.

하나하나 요구사항을 구현하는데 적지않은 시간이 들었지만 12 월 15 일 이내에 과제를 끝내기에는 일부만 끝내고 마무리하기에는 아쉬웠습니다.

두번째 아쉬웠던 점은 프레임워크의 제한 이었습니다.

실제 현업에서는 java swing 이 아닌 웹 프레임워크 (Java spring,) 등을 더 많이 사용하는데 웹 어플리케이션 프레임워크를 사용하지 않는 바람에 실제 현업에서 사용하지 않는 java swing 을 사용하는 것이 조금 아쉬웠습니다.

하지만 이제껏 만들어본적없는 calendar 어플 을 만들어본 결과 java 언어에 조금 더 익숙해질수 있었으며, java ui, java 앱의 구현방식 , 외부 라이브러리 사용 및 변경 및 활용할 수 있는 방안에 대해서 고민을 할수 있었습니다.

과제 명세에 대해서

이해가 안되는 부분 역시 있었습니다.

RSVP 부분인데 family 멤버들에게 rsvp 여부를 물어보고

추후 이벤트를 등록한다는 것인데 실제 google calendar 에서는

이벤트를 만들고 멤버 초대기능을 넣어서

초대를 수락한 멤버들에 대해서만 이벤트를 추가할수있게 하였습니다.

하지만 RSVP 는 프로세스상 두번을 거치므로, 제 생각에 너무 overhead 와 번거로운 기능이라고 생각합니다. 만일 진짜로 calendar 를 상용화할 생각이라면 ,rsvp 기능보다는 user 초대기능을 과제 명세로 넣는 것이 더 바람직하지 않나 생각합니다.