




Ứng dụng đăng ký user

- Nếu người dùng không hiểu trường nào đó trong form, có thể click chuột vào form và bấm Ctrl+H để hiện bảng trợ giúp
- Nếu người dùng muốn đọc điều khoản, click chuột vào nhãn Accept Term... Để hiện bảng điều khoản
- Nếu người dùng nhập thiếu một trường username, birthday cần có thông báo nhắc nhở nhập dữ liệu
- Khi người dùng check accept, nút Submit được enable, người dùng có thể nhấn Submit hoặc gõ Enter để gửi đi thông tin đăng ký
- Nếu người dùng bấm Reset, đưa các trường trong form về trạng thái ban đầu


 Registration Form

User Name

Gender:

☒  Male ☐  Female

Birthday



☐ Accept Term of Use (Click for detail)

Help (Ctrl+h)

Reset

Submit

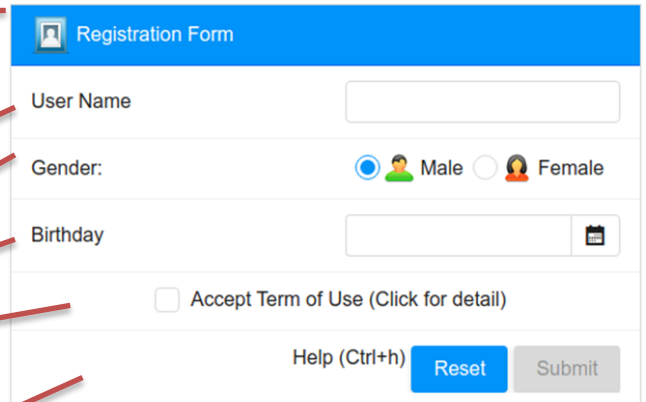
Thiết kế layout:

registration.zul

Khối Div bao ngoài grid để căn giữa cho grid và thiết lập khoảng cách lề trên. Div có độ rộng mặc định bằng chiều ngang màn hình

Vflex = "1" để tự điều chỉnh grid theo chiều dọc cho vừa nội dung chứa bên trong grid

```
<zk>
<div align="center" style="margin-top:50px;">
  <grid width="40%" vflex="1">
    <auxhead>
      <auxheader colspan="2"/>
    </auxhead>
    <columns visible="false">
      <column />
      <column />
    </columns>
    <rows>
      <row> </row>
      <row> </row>
      <row> </row>
      <row spans="2" align="center">
        <hlayout>
          </hlayout>
      </row>
      <row spans="2" align="right">
        <hlayout>
          </hlayout>
      </row>
    </rows>
  </grid>
</div>
</zk>
```



The image shows a registration form titled "Registration Form". It contains the following fields and elements:

- User Name:** A text input field.
- Gender:** Radio buttons for "Male" and "Female".
- Birthday:** A date picker field.
- Accept Term of Use:** A checkbox with the text "Accept Term of Use (Click for detail)".
- Buttons:** "Help (Ctrl+h)", "Reset", and "Submit".

Cấu trúc project:

- RegistrationAppZkDemo
 - src
 - vnua.fita.controller
 - RegistrationComposer.java
 - JRE System Library [jdk1.8.0_241]
 - Apache Tomcat v9.0 [Apache Tomcat v9.0]
 - Web App Libraries
 - build
 - WebContent
 - images
 - female.png
 - male.png
 - user_icon.png
 - META-INF
 - WEB-INF
 - help.zul
 - registration.zul
 - termofuse.zul

registration.zul

Trần Trung Hiếu - Fita - vnua

<zkl>

```
<style>.z-button i { font-size: 24px; }</style>
<div align="center" style="margin-top: 50px;" apply="vnua.fita.controller.RegistrationComposer">
  <grid id="formGrid" ctrlKeys="^h" width="40%" vflex="1">
    <auxhead>
      <auxheader colspan="2" label="Registration Form"
        style="font-size: 16px" image="/images/user_icon.png"/>
    </auxhead>
    <columns visible="false">
      <column />
      <column />
    </columns>
    <rows>
```

ctrlKeys: Khai báo khả năng nhận sự kiện phím Ctrl+H khi con trỏ đang focus vào vùng Grid

Khai báo lớp quản lý và xử lý sự kiện khối div

Cần nhóm các thẻ radio trong radiogroup để chỉ có thể chọn được một trong hai lựa chọn

```
      <row id="nameRow">
        <!-- constraint="no empty" -->
        User Name
        <textbox id="nameBox" hflex="1"
          constraint="no empty: Nhập username"/>
      </row>
      <row>
        Gender:
        <radiogroup id="genderRadio">
          <radio label="Male" value="male"
            image="/images/male.png" checked="true"/>
          <radio label="Female" value="female"
            image="/images/female.png"/>
        </radiogroup>
      </row>
      <row>
        Birthday
        <datebox id="birthdayBox" hflex="1"
          constraint="no empty: Nhập ngày sinh , no today, no future"/>
      </row>
      <row spans="2" align="center">
        <hlayout>
          <checkbox id="acceptTermBox"/>
          <label
            value=" Accept Term of Use (Click for detail)"
            popup="termDetail, position=end_after"/>
        </hlayout>
      </row>
      <row spans="2" align="right">
        <hlayout>
          <label value="Help (Ctrl+h)"/>
          <button id="resetButton" label="Reset"/>
          <button id="submitButton" label="Submit"
            disabled="true"/>
        </hlayout>
      </row>
    </rows>
  </grid>
```

Ràng buộc trường ngày sinh không được: (rỗng, ngày hiện tại, tương lai)

Khai báo popup tương ứng hiển thị khi click

Mặc định nút Submit ở chế độ disable

```
  </div>
  <popup id="termDetail">
    <include src="termofuse.zul"/>
  </popup>
  <popup id="helpPopup">
    <include src="help.zul"/>
  </popup>
```

<zkl>

Khai báo các popup để kích hoạt khi có sự kiện gọi tới

Help.zul và termofuse.zul được cung cấp file kèm theo

```
package vnua.fita.controller;

import org.zkoss.zk.ui.select.SelectorComposer;
import org.zkoss.zk.ui.select.annotation.Listen;
import org.zkoss.zk.ui.select.annotation.Wire;
import org.zkoss.zul.Button;
import org.zkoss.zul.Checkbox;
import org.zkoss.zul.Datebox;
import org.zkoss.zul.Div;
import org.zkoss.zul.Messagebox;
import org.zkoss.zul.Popup;
import org.zkoss.zul.Radiogroup;
import org.zkoss.zul.Row;
import org.zkoss.zul.Textbox;

// Lớp RegistrationComposer muốn ZK hỗ trợ quản lý trang ZUL
// cần khai báo thừa kế từ SelectorComposer<Component>
// Div là thành phần khai báo apply ở trên trang ZUL
@SuppressWarnings("serial")
public class RegistrationComposer extends SelectorComposer<Div> {

    // KHAI BÁO CÁC THÀNH PHẦN TRÊN TRANG CẦN TƯƠNG TÁC
    @Wire("#submitButton")
    private Button submitButton;
    @Wire("#nameBox")
    private Textbox nameBox;
    @Wire("#genderRadio")
    private Radiogroup genderRadio;
    @Wire("#birthdayBox")
    private Datebox birthdayBox;
    @Wire("#acceptTermBox")
    private Checkbox acceptTermCheckbox;
    @Wire("#nameRow")
    private Row nameRow;
    @Wire("#helpPopup")
    private Popup helpPopup;

    // LẮNG NGHE SỰ KIỆN TRÊN CHECKBOX
    // Tra cứu tên sự kiện trên tài liệu reference của ZK về CheckBox
    @Listen("onCheck = #acceptTermBox")
    public void changeSubmitStatus(){
        // Nếu checkbox được check
        if (acceptTermCheckbox.isChecked()){
            // Enable nút Submit
            submitButton.setDisabled(false);
            // Đặt icon sclass cho nút Submit
            // Đây là thư viện icon được tích hợp sẵn
            submitButton.setIconSclass("z-icon-check");
        }else{
            submitButton.setDisabled(true);
            submitButton.setIconSclass("");
        }
    }
}
```

```
@Listen("onClick = #resetButton")
public void reset(){
    //set raw value để tránh validate cấu hình trên registration.zul
    nameBox.setRawValue(null);
    genderRadio.setSelectedIndex(0);
    birthdayBox.setRawValue(null);
    acceptTermCheckbox.setChecked(false);
    submitButton.setDisabled(true);
}

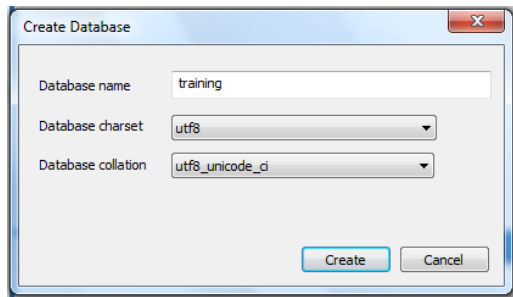
@Listen("onClick = #submitButton")
public void submit(){
    // Tam thời chưa có xử lý database
    // Lấy username và hiển thị hộp thoại báo thành công
    MessageBox.show("Congratulation! " + nameBox.getValue() + ". Your registration is
success.");
}

// Bắt sự kiện phím Enter trên Grid, thực hiện submit
@Listen("onOK = #formGrid")
public void onOK(){
    submit();
}

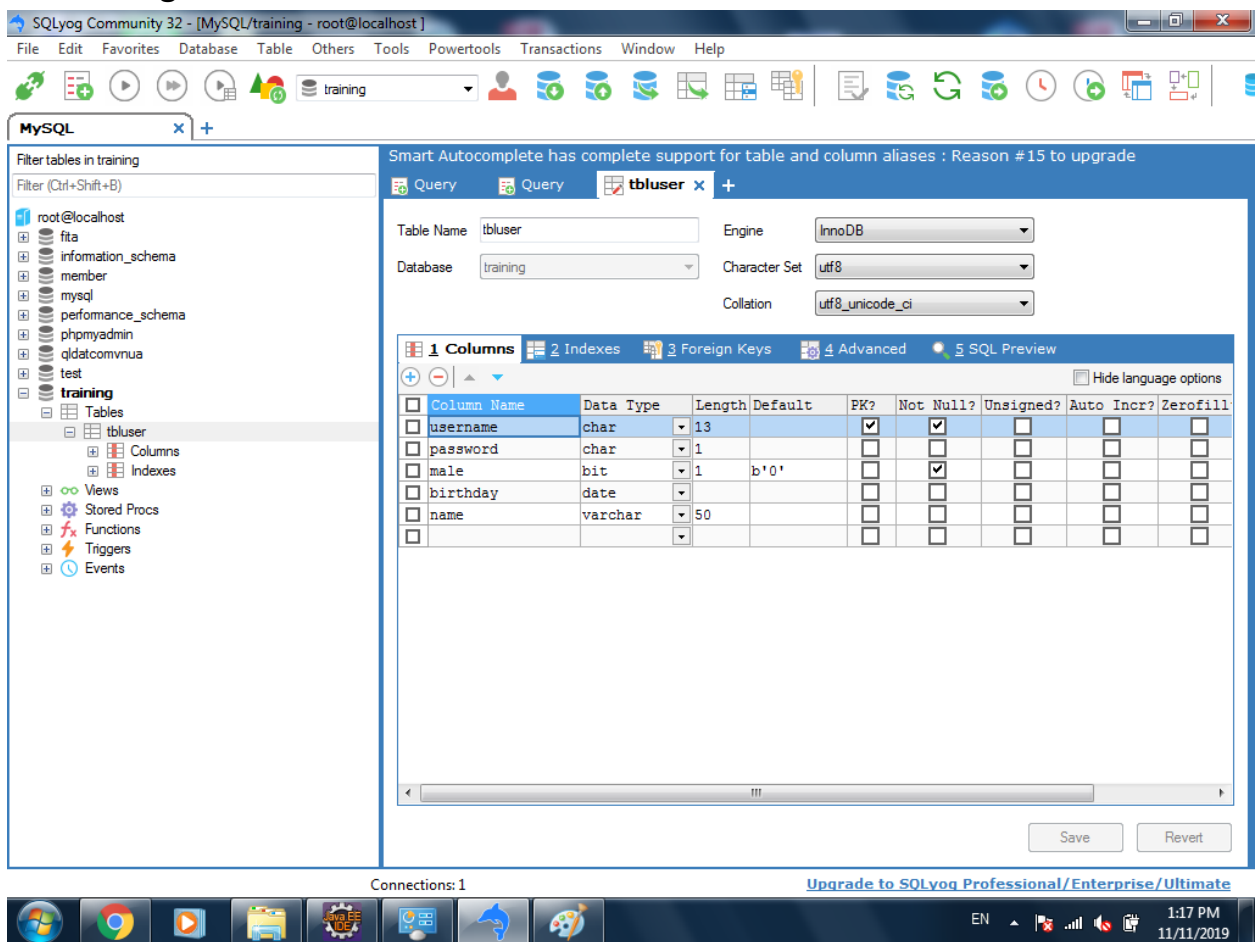
// Bắt sự kiện phím Ctrl trên Grid
@Listen("onCtrlKey = #formGrid")
public void toggleHelpRow(){
    // Hiển thị help popup tại vị trí trường username
    // end_before: xuất hiện ở cuối phía trên
    helpPopup.open(nameRow, "end_before");
}
}
```

Thêm xử lý database:

1. Cài đặt mysql community, sqlyog community
(Xem hướng dẫn trên slide)
2. Thêm apache tomcat vào eclipse
(Xem hướng dẫn trên slide)
3. Tạo database



4. Thêm bảng user



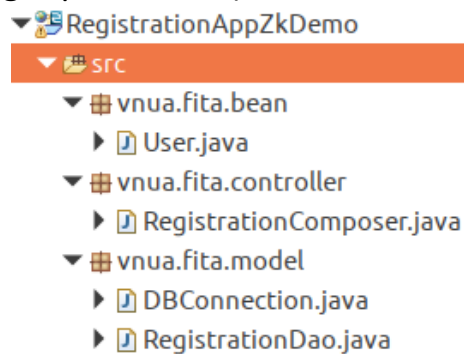
5. Thêm jdbc driver vào project

Link download jar file (tương ứng phiên bản mysql):

<https://mvnrepository.com/artifact/mysql/mysql-connector-java/8.0.19>

Thêm vào project: Copy jar file vào thư mục **web-inf/lib**

6. Thêm các lớp (file cung cấp kèm theo)



- a. User
- b. DBConnection
- c. RegistrationDao

7. Sửa code lớp RegistrationComposer

a) Thêm khai báo thuộc tính DAO dùng để truy cập database

```
private RegistrationDao registrationDao = new RegistrationDao(  
    "jdbc:mysql://localhost:3306/training??useUnicode=true&&characterEncoding=UTF-8",  
    "root", "123456");
```

b) Sửa code phương thức submit()

```
@Listen("onClick = #submitButton")  
public void submit(){  
  
    // Insert database  
    User user = new User();  
  
    user.setName(nameBox.getValue());  
    boolean male = genderRadio.getSelectedIndex() == 0 ? true : false;  
    user.setMale(male);  
    user.setBirthday(birthdayBox.getValue());  
    boolean result = registrationDao.add(user);  
    if(result) {  
        MessageBox.show("Congratulation! " + nameBox.getValue() + ". Your  
registration is success.");  
        reset();  
    } else {  
        MessageBox.show("Your registration is error!");  
    }  
}
```