[Thực hành] Ứng dụng kiểm tra email hợp lệ sử dụng Gradle

**Mục tiêu**

Luyện tập sử dụng Controller trong Spring MVC.

**Điều kiện**

Biết cách sử dụng Controller trong Spring MVC.

**Mô tả**

Trong phần này, chúng ta sẽ xây dựng một ứng dụng cho phép kiểm tra tính hợp lệ của một email.

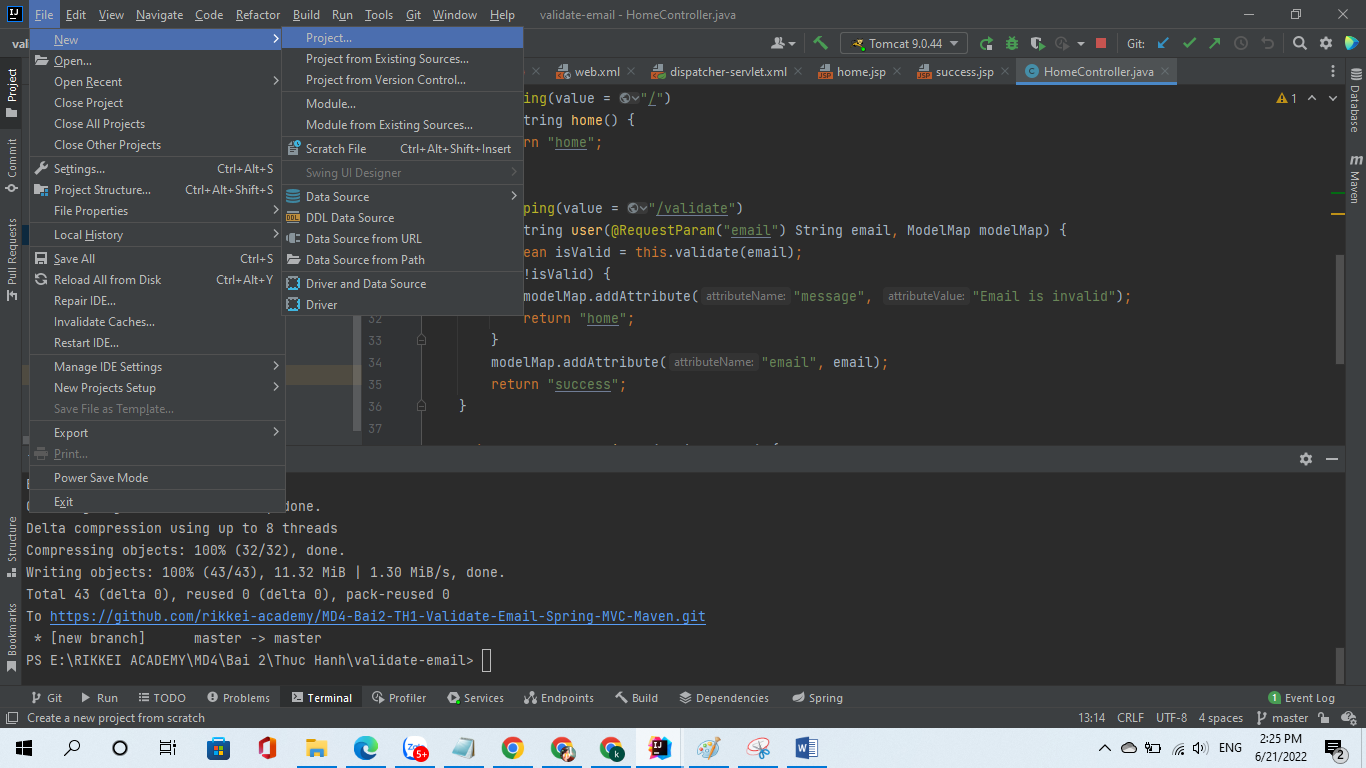
Ứng dụng cho phép người dùng nhập vào một email, nhấn nút kiểm tra và sẽ hiển thị thông báo kết quả.

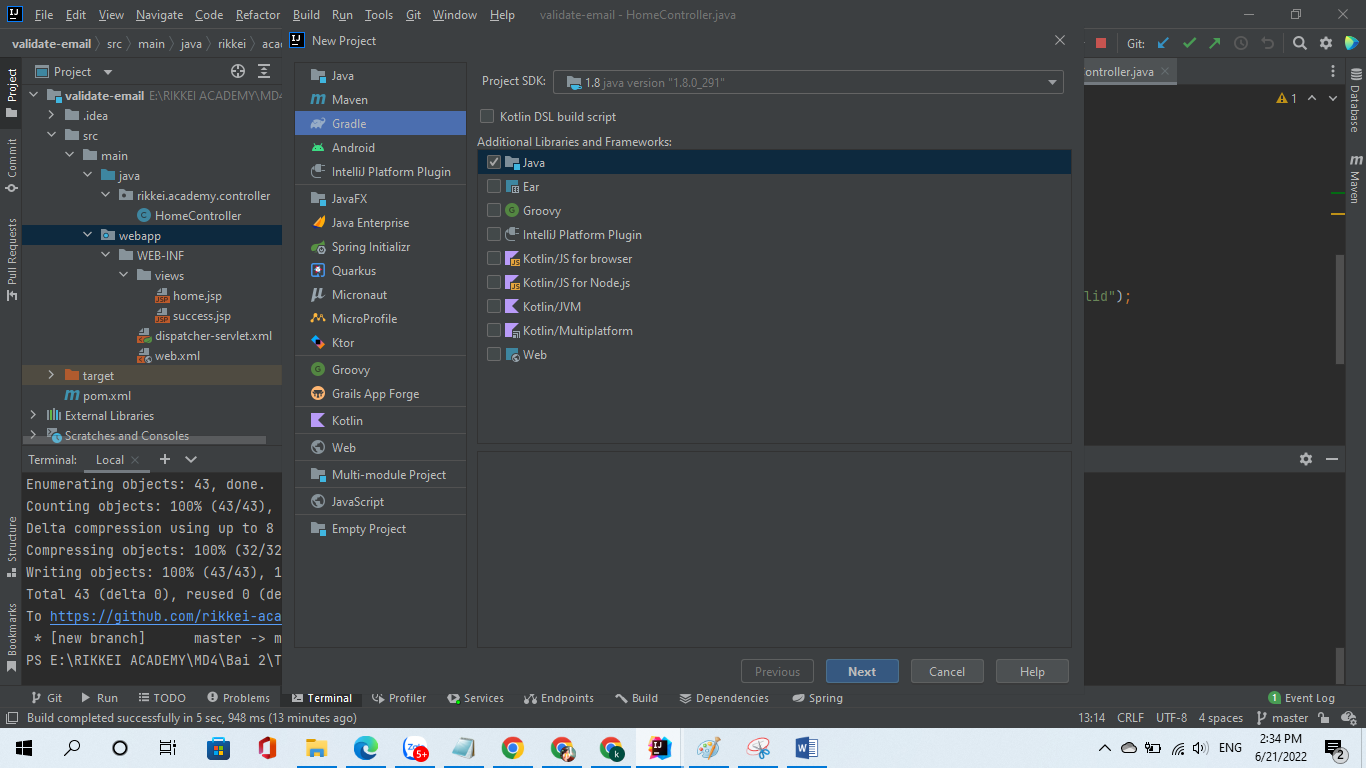
Một email hợp lệ phải đạt những yêu cầu sau:

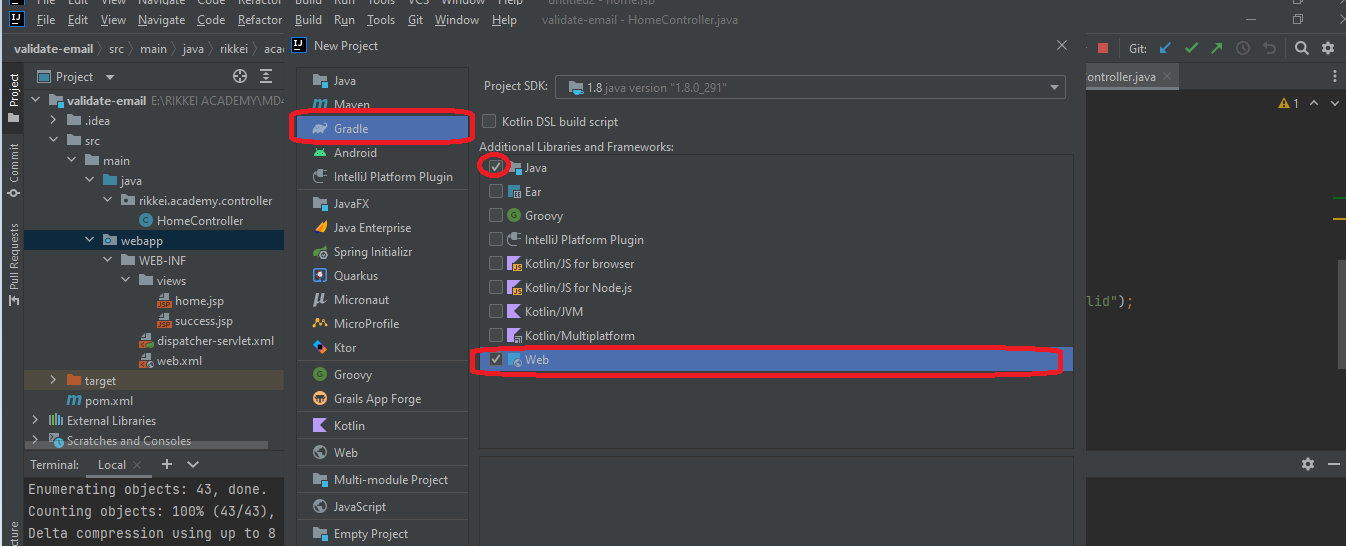
* Bắt đầu bằng một ký tự
* Không chứa các ký tự đặc biệt
* Có một ký tự @ ngăn cách tên email và domain
* Domain phải hợp lệ

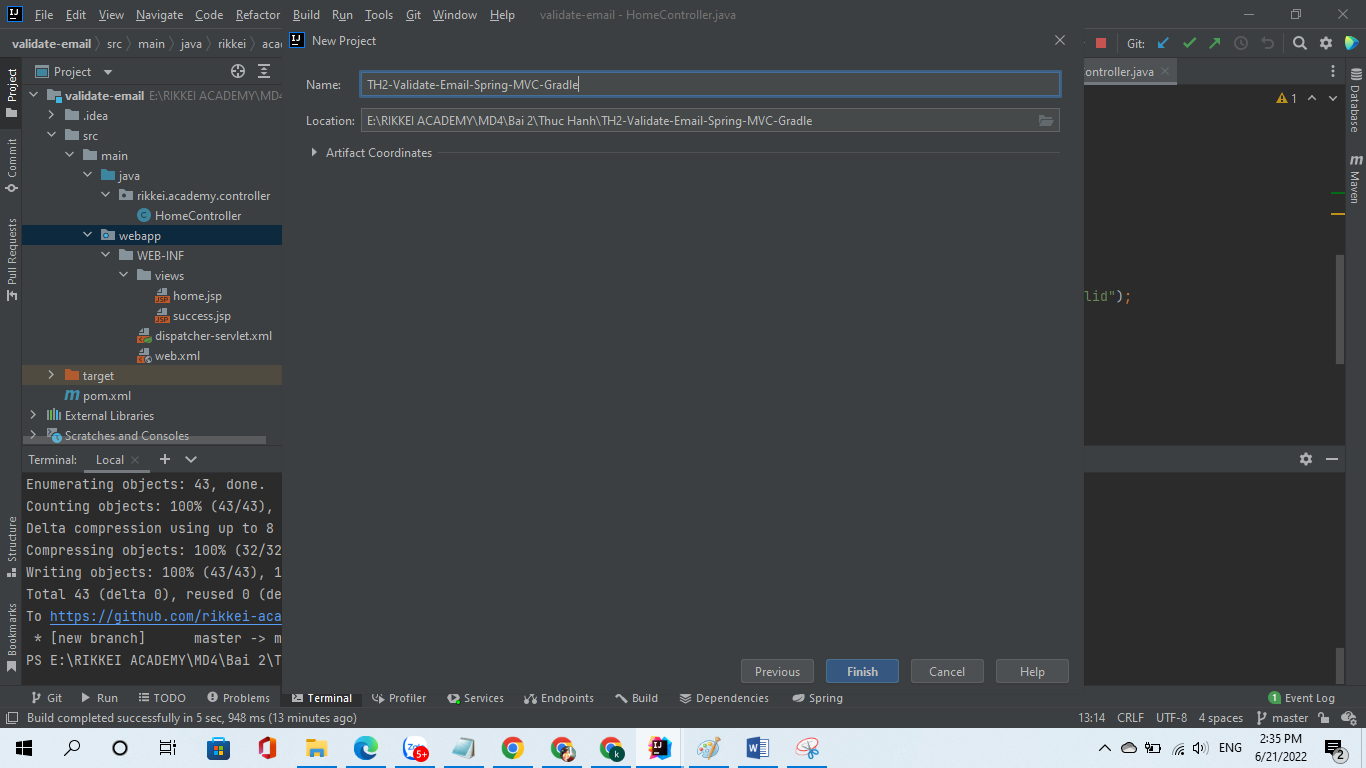
**Hướng dẫn**

* **Tạo một project: Spring MVC Gradle**

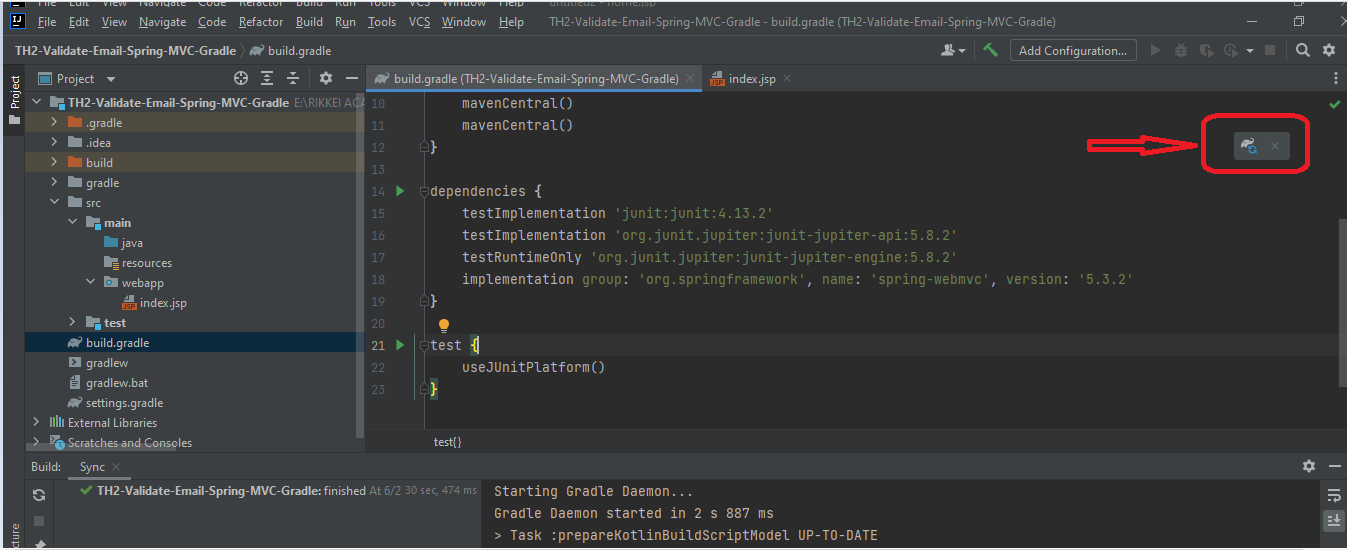








* Build thêm thư viện vào file: build gradle
* dependencies **{** testImplementation 'junit:junit:4.11'  
   testImplementation 'org.junit.jupiter:junit-jupiter-api:5.8.1'  
   testRuntimeOnly 'org.junit.jupiter:junit-jupiter-engine:5.8.1'  
   implementation group: 'org.springframework', name: 'spring-webmvc', version: '5.3.2'  
  **}**
* Build thư viện:



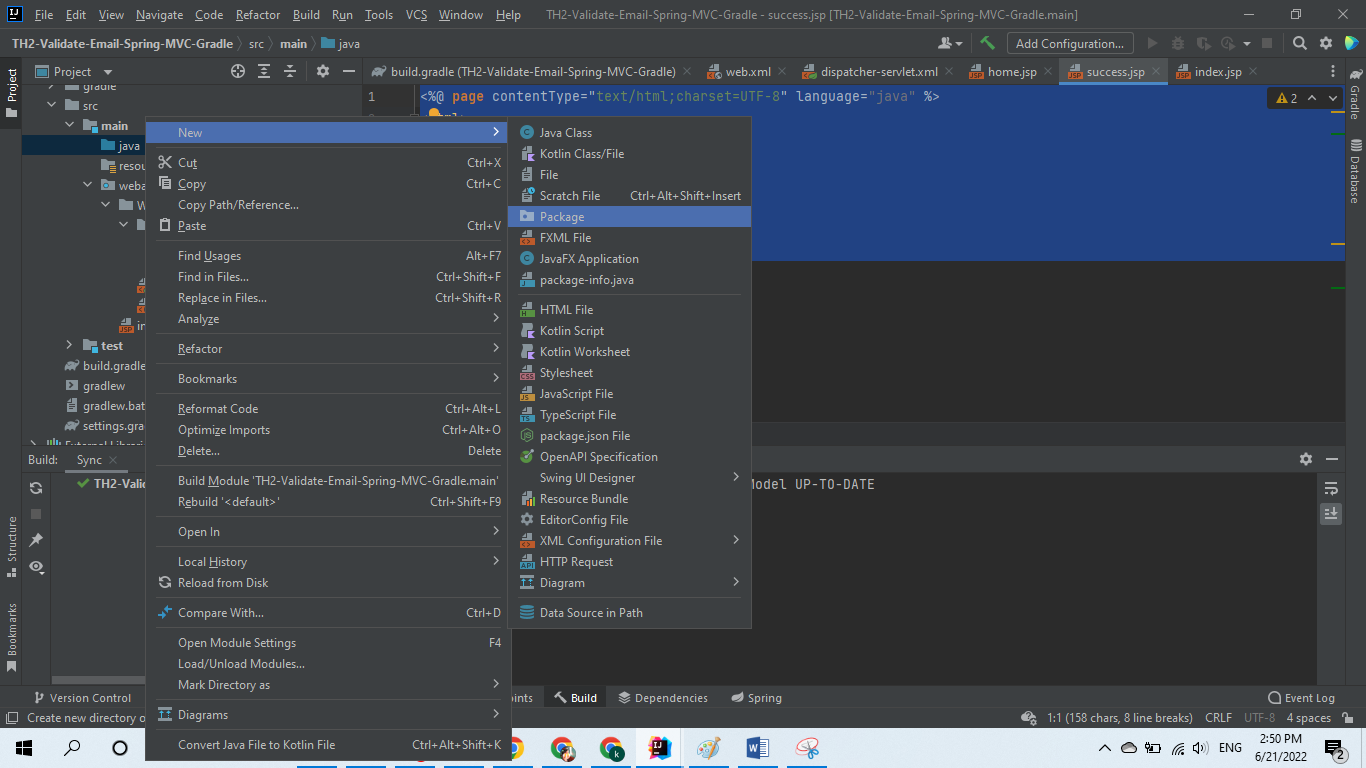
* Tạo thư mục WEB-INF trong webapp:
* Tạo file: web.xml trong WEB-INF và thêm code config vào như các bài thực hành trước:
* <?xml version="1.0" encoding="UTF-8"?>  
    
  <web-app id="WebApp\_ID" version="2.4"  
   xmlns="http://java.sun.com/xml/ns/j2ee"  
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
   xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee  
   http://java.sun.com/xml/ns/j2ee/web-app\_2\_4.xsd">  
   <display-name>example</display-name>  
   <servlet>  
   <servlet-name>spring</servlet-name>  
   <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  
   <init-param>  
   <param-name>contextConfigLocation</param-name>  
   <param-value>/WEB-INF/dispatcher-servlet.xml</param-value>  
   </init-param>  
   <load-on-startup>1</load-on-startup>  
   </servlet>  
   <servlet-mapping>  
   <servlet-name>spring</servlet-name>  
   <url-pattern>/</url-pattern>  
   </servlet-mapping>  
  </web-app>
* Tạo file: **dispatcher-servlet.xml:** Thêm code config vào

<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:p="http://www.springframework.org/schema/p"  
 xmlns:mvc="http://www.springframework.org/schema/mvc"  
 xsi:schemaLocation="  
 http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context.xsd  
 http://www.springframework.org/schema/mvc  
 http://www.springframework.org/schema/mvc/spring-mvc.xsd">  
  
 <!-- Spring MVC Annotation -->  
 <context:annotation-config />  
 <mvc:annotation-driven/>  
  
 <!--Controller -->  
 <context:component-scan base-package="rikkei.academy.controller"/>  
 <!--View -->  
 <bean id="viewResolver"  
 p:prefix="/WEB-INF/views/" p:suffix=".jsp"  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver"/>  
</beans>

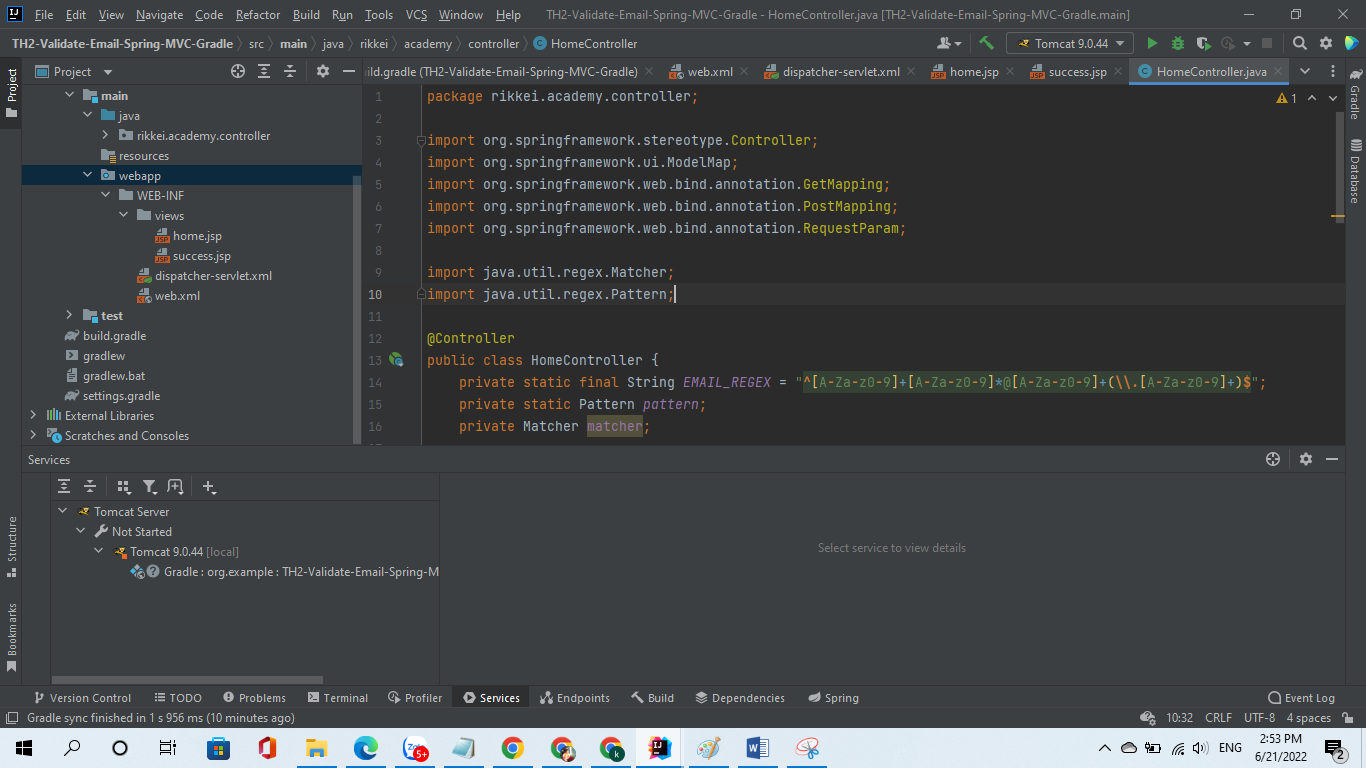
* Tạo views trong WEB-INF và thêm các file .jsp:
* Tạo file: home.jsp và thêm code vào:
* <%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
  <html>  
  <head>  
   <title>Home</title>  
  </head>  
  <body>  
  <h1>Email Validate</h1>  
  <h3 style="color:red">${message}</h3>  
  <form action="validate" method="post">  
   <input type="text" name="email"><br>  
   <input type="submit" value="Validate">  
  </form>  
  </body>  
  </html>
* Tạo file: success.jsp và thêm code vào:

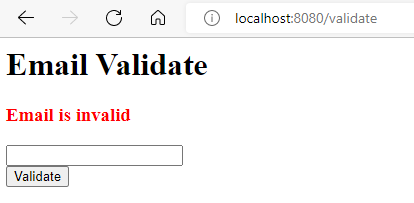
<%@ **page** contentType="text/html;charset=UTF-8" language="java" %>  
<html>  
<head>  
 <title>Success</title>  
</head>  
<body>  
<h3>Hi ${email}</h3>  
</body>  
</html>

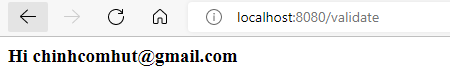
* Tạo package: rikkei.academy.controller từ java:



* Tạo Class: HomeController trong package: controller: Thêm code vào như sau
* package rikkei.academy.controller;  
    
  import org.springframework.stereotype.Controller;  
  import org.springframework.ui.ModelMap;  
  import org.springframework.web.bind.annotation.GetMapping;  
  import org.springframework.web.bind.annotation.PostMapping;  
  import org.springframework.web.bind.annotation.RequestParam;  
    
  import java.util.regex.Matcher;  
  import java.util.regex.Pattern;  
    
  @Controller  
  public class HomeController {  
   private static final String *EMAIL\_REGEX* = "^[A-Za-z0-9]+[A-Za-z0-9]\*@[A-Za-z0-9]+(\\.[A-Za-z0-9]+)$";  
   private static Pattern *pattern*;  
   private Matcher matcher;  
    
   public HomeController() {  
   *pattern* = Pattern.*compile*(*EMAIL\_REGEX*);  
   }  
    
   @GetMapping(value = "/")  
   public String home() {  
   return "home";  
   }  
    
   @PostMapping(value = "/validate")  
   public String user(@RequestParam("email") String email, ModelMap modelMap) {  
   boolean isValid = this.validate(email);  
   if (!isValid) {  
   modelMap.addAttribute("message", "Email is invalid");  
   return "home";  
   }  
   modelMap.addAttribute("email", email);  
   return "success";  
   }  
    
   private boolean validate(String regex) {  
   matcher = *pattern*.matcher(regex);  
   return matcher.matches();  
   }  
  }
* Xóa file: index.jsp và add Tomcat vào chạy phần mềm và test các trường hợp validate Email:







Mã nguồn tham khảo:

<https://github.com/rikkei-academy/MD4-Bai2-TH2-Validate-Email-SpringMVC-Gradle.git>

* HV nộp bài lên link git cá nhân.