MÔ HÌNH MVC

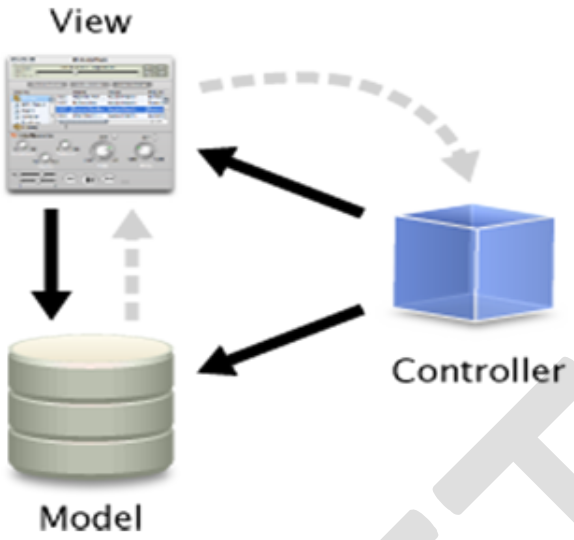
Lý thuyết

Trong quá trình phát triển phần mềm, một hệ thống thường phải được xây dựng theo một bản thiết kế. Các bản thiết kế này thường được xây dựng theo các quy định khác nhau tùy vào từng loại phần mềm. Từ một số quy định và nguyên tắc thiết kế, xây dựng hệ thống, người ta đưa ra các mô hình để phát triển, để thiết kế và xây dựng phần mềm.

Mô hình MVC là gì?

Mô hình MVC là một mô hình được sử dụng trong thiết kế phần mềm. Mô hình này giúp người phát triển hệ thống tách biệt ứng dụng của họ thành 3 thành phần là Model, View, Controler. Mỗi thành phần sẽ có một nhiệm vụ khác nhau. Cụ thể,

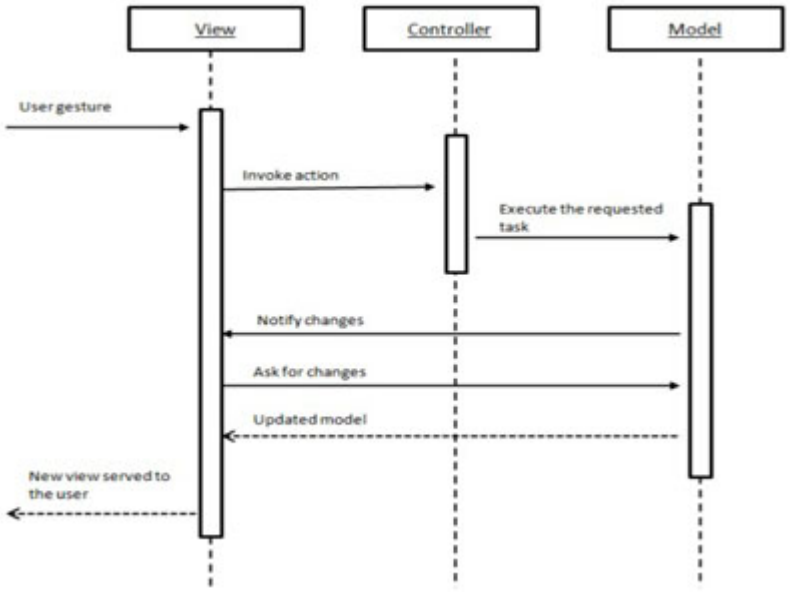
- Model là thành phần chứa tất cả các xử lý nghiệp vụ, logic, truy xuất dữ liệu. Ví dụ: Mô tả các thực thể, thuật toán, truy xuất cơ sở dữ liệu,...
- View là thành phần đảm nhận việc hiển thị thông tin, tương tác với người sử dụng. VD: các cửa sổ tương tác giữa người dùng và ứng dụng, các thành phần giao diện do người dùng tự xây dựng,...
- Controler là thành phần có nhiệm vụ điều hướng các tương tác của người dùng thành các xử lý phù hợp. Nói cách khác, Controler có nhiệm vụ kết nối giữa View và Model



Hình 1 Mô hình MVC

Mô hình MVC hoạt động như thế nào?

Mô hình MVC hoạt động một cách khá đơn giản. Người sử dụng có một tương tác với hệ thống trên View, Controler sẽ nhận các tương tác này và điều hướng chúng đến các xử lý ở Model. Model nhận yêu cầu xử lý, xử lý, sau đó trả về kết quả. Kết quả này có thể được trả thẳng lên View (hoặc trả lại qua Controler).



Hình 2 Nguyên tắc làm việc của MVC

Mô hình MVC có ưu nhược điểm như thế nào?

- Ưu điểm:

Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì..

- Nhược điểm:

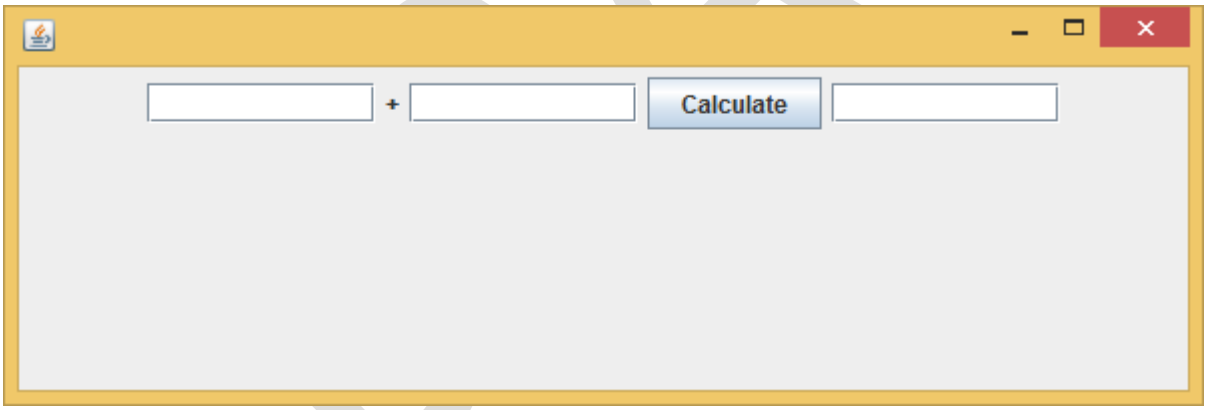
Đối với dự án nhỏ việc áp dụng mô hình MC gây cồng kềnh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các thành phần.

Thực hành

Hướng dẫn

Chúng ta xét ví dụ:

Xây dựng chương trình máy tính đơn giản để thực hiện phép tính cộng hai số. Giao diện của chương trình như sau:



Chức năng xử lý:

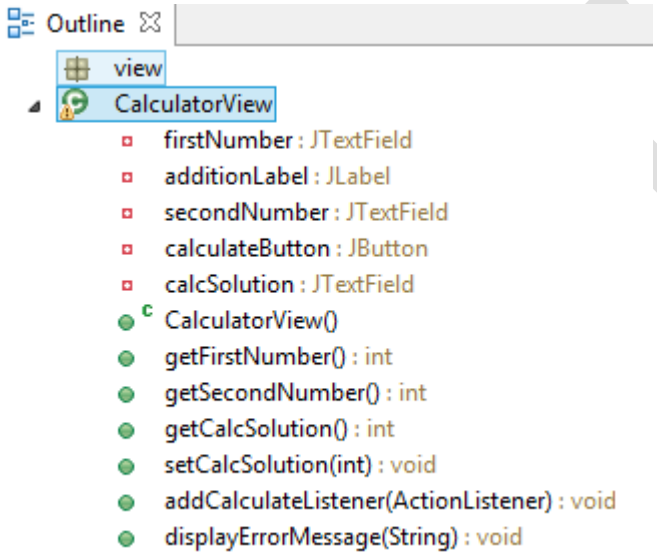
Khi người sử dụng ấn vào nút Calculate, kết quả sẽ được hiện ra ở ô kết quả.

Các bước thực hiện: Xem video: https://www.youtube.com/watch?v=dTVVa2gfht8

Bước 1: Xây dựng lớp CalculatorView với giao diện như yêu cầu.

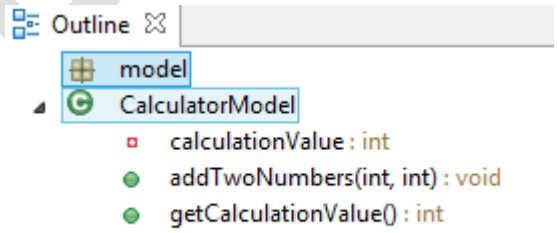
- Cần những Component nào? Layout nào?

- Cung cấp cho View một số phương thức để truy cập đến các TextBox (chứa số thứ 1, số thứ 2, kết quả), phương thức để thêm xử lý sự kiện cho nút Calculate. Sau này, Controller sẽ dùng các phương thức này để điều khiển View. Cụ thể, với textbox chứa số thứ 1 và số thứ 2, sẽ cần một phương thức để lấy giá trị nhập trong ô (getter) với textbox chứa kết quả, sẽ cần một phương thức để ghi giá trị kết quả (setter); với nút Calculate, sẽ cần một phương thức để thêm bộ xử lý sự kiện (addCalculateListener).



Hình 3 Outline của lớp View

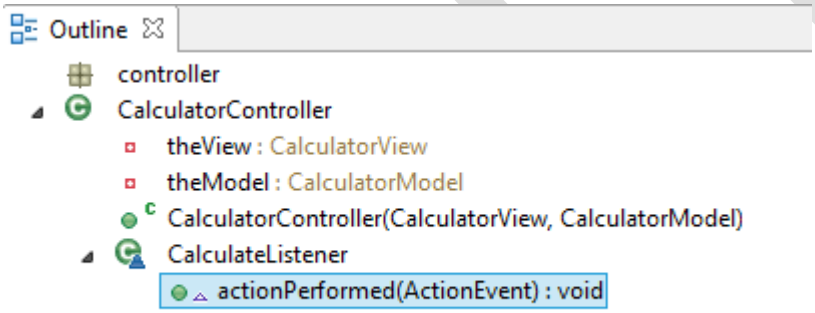
Bước 2: Xây dựng lớp CalculatorModel chứa phương thức nghiệp vụ là phương thức cộng. Phương thức này có thể lưu lại kết quả vào một thuộc tính (trường) của lớp. Thuộc tính này sẽ cần phải được cài đặt kèm theo một phương thức cho lớp Controller có thể truy cập để lấy giá trị (getter).



Hình 4 Outline của lớp Model

Bước 3: Xây dựng lớp CalculatorController điều khiển Model và View

- Lớp này sẽ có hai đối tượng thuộc lớp View và Model.
- Trong lớp này, sẽ có một lớp lồng (Nested Class) để xử lý cho sự kiện nhấn vào nút Calculate trên View. Trong quá trình xử lý sự kiện, ta sẽ thực hiện lần lượt các bước:
    - Lấy hai giá trị trong hai ô textbox của View (bằng các getter)
    - Tính toán nghiệp vụ (cộng tổng) trên hai giá trị này bằng phương thức xử lý nghiệp vụ của Model.
    - Lấy giá trị kết quả trong Model (bằng getter của Model) để hiển thị lên View (bằng setter của View).
- Khi khởi tạo lớp này (trong hàm dựng), ta sẽ tạo ra View và Model, lưu ý, khi tạo ra View, cần thêm bộ lắng nghe sự kiện cho nút Calculate trên View bằng cách sử dụng phương thức addCalculateListener đã cài đặt ở lớp View.



Hình 5 Outline của lớp Controller

Bài tập

Bài 1

Xây dựng ứng dụng SimpleCalculator theo mô hình MVC theo yêu cầu sau:
Giao diện:

Chức năng:

- Khi ấn vào dấu =, kết quả phép tính được thực hiện.
- Kiểm tra sự hợp lệ của dữ liệu nhập vào.

Bài 2

Xây dựng ứng dụng HUSNotepad theo mô hình MVC theo yêu cầu sau:
Chức năng:

- Soạn thảo tệp văn bản mới;

- Mở tệp văn bản đã soạn thảo để chỉnh sửa;

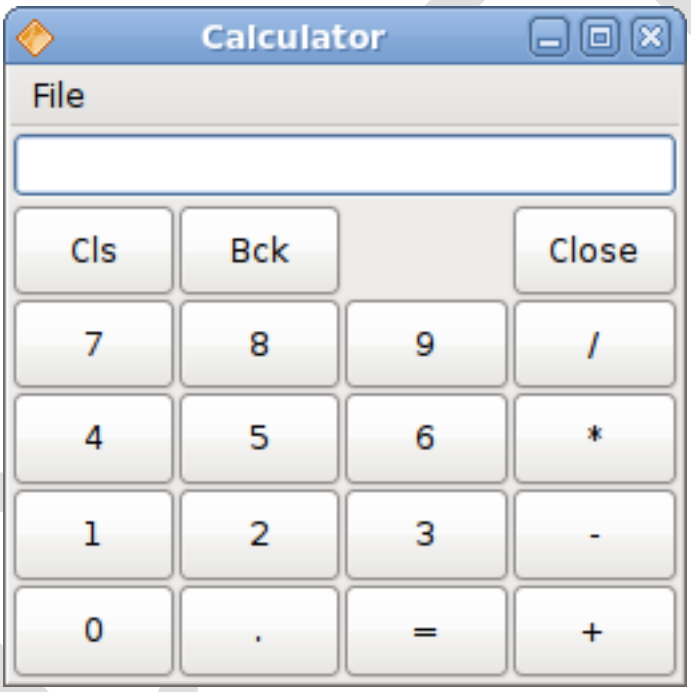
- Lưu tệp văn bản

- Lưu tệp văn bản với tên khác

- Các chức năng khác như phần mềm Notepad trên window.

Bài 3

Xây dựng ứng dụng HUSCalculator theo mô hình MVC theo yêu cầu sau:
Giao diện:



Chức năng:

- Khi ấn Cls, Xóa toàn bộ công thức,
- Khi ấn Bck, xóa số hiện tại đang gõ, không xóa công thức
- Khi ấn Close, thoát chương trình
- Các số nhập vào phải hợp lệ (vd: không được có số có 2 dấu .,…)
- Khi ấn các dấu +, -, *, /; kết quả công thức đang viết được tính luôn và sử dụng khi tính toán ở bước sau. VD: 12 + 3 * 5, Khi ấn đến dấu *, sẽ cho kết quả là 15 và khi ấn số 5 rồi ấn dấu =, kết quả là 75.