

Orientação a Objetos - 09/2022 - Turma 02

Trabalho Prático - Entrega 3

21/09/2022

Participantes:

João Pedro Silva Rodrigues - 21/1031074

Raphael Mendes da Silva - 21/1039690

Lucas Freire Lopes - 19/0112093

Fábio Alessandro Torres Santos - 20/0037170

Introdução

Durante a fase do projeto começamos a nos preocupar com a arquitetura da aplicação, o que significa que damos real valor à tecnologia, diferente da fase de análise onde ainda estamos esboçando o problema que será resolvido. A arquitetura de uma aplicação é o modelo que define as suas estruturas. Essas estruturas englobam componentes de *software*, propriedades dos componentes, os relacionamentos entre os componentes e por fim todos os elementos que fazem parte da estrutura básica padrão e como eles interagem entre si.

Por sua vez, o *Model-View-Controller* (MVC) é um padrão de arquitetura de aplicações que divide a aplicação em três camadas: a visão (*View*), o modelo (*Model*) e o controlador (*Controller*). Este padrão foi desenvolvido por Trygve Reenskaug em 1979 com o objetivo de ser utilizado como arquitetura para aplicativos desktop, porém o padrão acabou se popularizando para uso em sistemas web.

- **Padrão MVC**

O principal propósito do padrão MVC é separar a lógica de negócios (*Model*) da interface do usuário (*View*) e do fluxo da aplicação (*Controller*), que permite usarmos uma mesma lógica de negócio que possa ser acessada através de várias interfaces diferentes como, por exemplo, *desktop*, *web* e *mobile*.

Outras vantagens são a possibilidade de reescrever partes da GUI e/ou *Controller* sem alterar o *Model*, facilidade na manutenção e adição de recursos, e fácil reaproveitamento e manutenção de código, etc.

- **Modelo (*Model*)**

A camada *Model* é a camada que gerencia e controla a forma em que os dados se comportam por meio das funções, lógicas e regras estabelecidas. Ela é a camada que sabe o que o aplicativo quer fazer e é a principal estrutura computacional da arquitetura de dados, pois é ela que modela o problema a ser resolvido, recebendo as informações do *Controller*, validando se ela está correta ou não, e então enviando a resposta mais adequada.

- **Visão (*View*)**

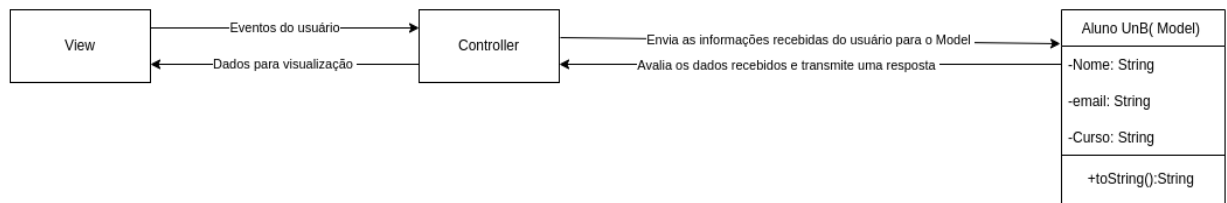
A camada *View* é a camada responsável por apresentar as informações de forma visual ao usuário, usando recursos como mensagens, botões e telas. Essa camada não sabe sobre nenhum dos processos do código, pois sua função é apenas receber e transmitir informação entre o usuário e a aplicação, sendo a única parte do código que pode se comunicar diretamente com o usuário, mas ela pode se comunicar de volta com o *Model* e o *Controller* para reportar o seu estado.

- **Controlador (*Controller*)**

A camada *Controller* é a camada que gerencia às duas outras camadas, interpretando as roteando as requisições do usuário enviadas pela camada *View*, e então transmitindo as respostas fornecidas pela camada *Model* para efetuar a alteração apropriada.

- **Diagrama de Classes UML**

Observe a seguir um diagrama UML de classes que exemplifica a funcionalidade do modelo MVC:



Conclusão

Neste relatório vimos o que é arquitetura de *software* e o padrão de arquitetura de *software* MVC (*Model-View-Controller*), analisamos os seus elementos e como eles interagem entre si, e também vimos as vantagens de adotar um padrão como o MVC que se caracteriza pela facilidade na obtenção de múltiplas visões dos mesmos dados. No entanto, definir a arquitetura de um *software* é mais que escolher seu padrão arquitetural, pois também devemos escolher a tecnologia, a linguagem a integrar, interfaces com o usuário e assim por diante. Portanto, devemos sempre reunir o máximo de pessoas experientes possíveis, conhecimentos e análise do projeto para que possamos enfim escolher a melhor arquitetura possível para o *software* que está sendo desenvolvido.

Bibliografia

- <http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/arqu/mvc/mvc.htm>
- <https://www.portalgsti.com.br/2017/08/padrao-mvc-arquitetura-model-view-controller.html>
- <https://www.lewagon.com/pt-BR/blog/o-que-e-padrao-mvc>
- <https://www.devmedia.com.br/introducao-ao-padrao-mvc/29308>