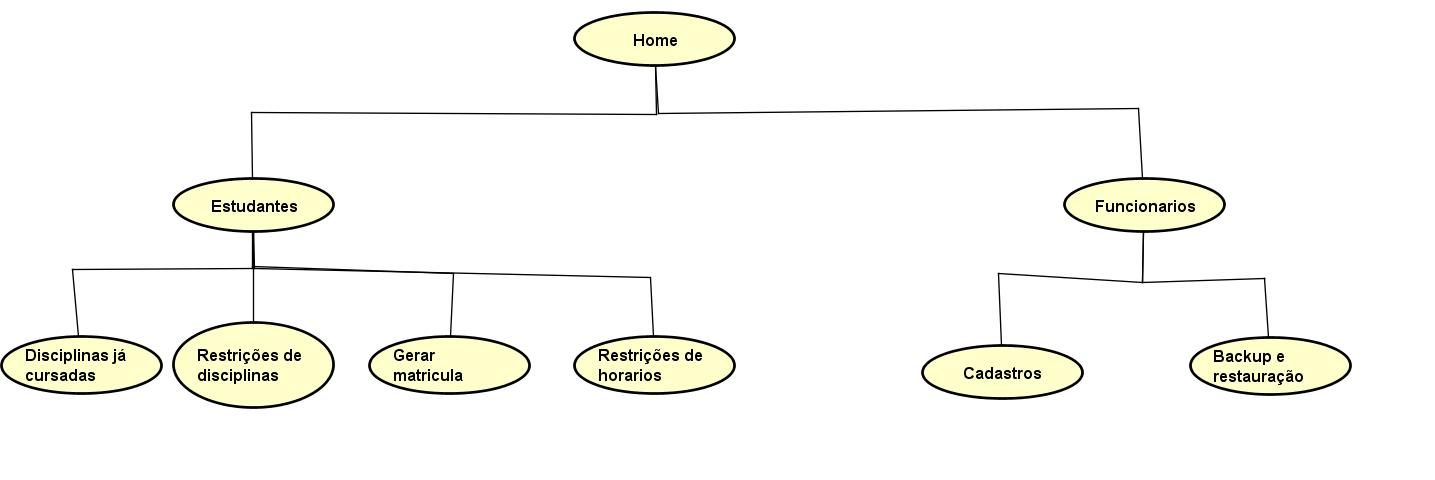
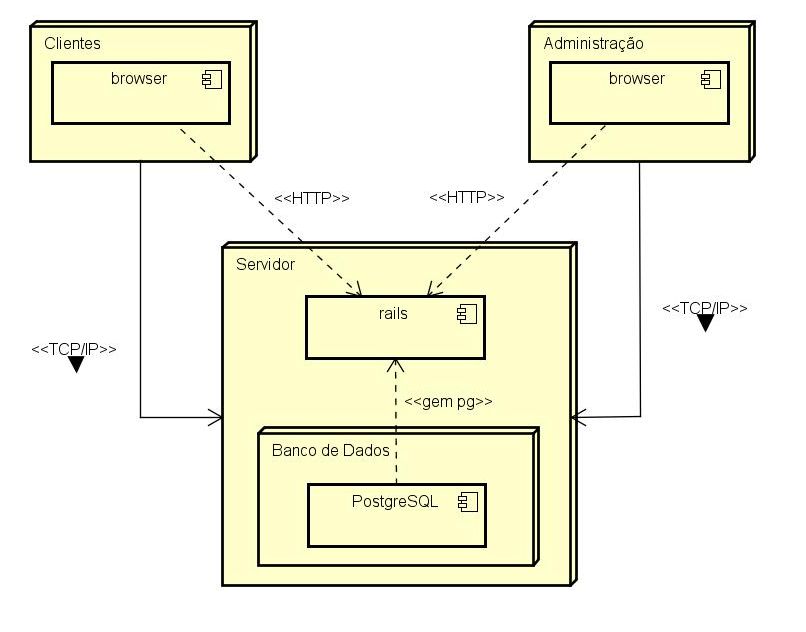
**Design**

**Estrutura de Design**

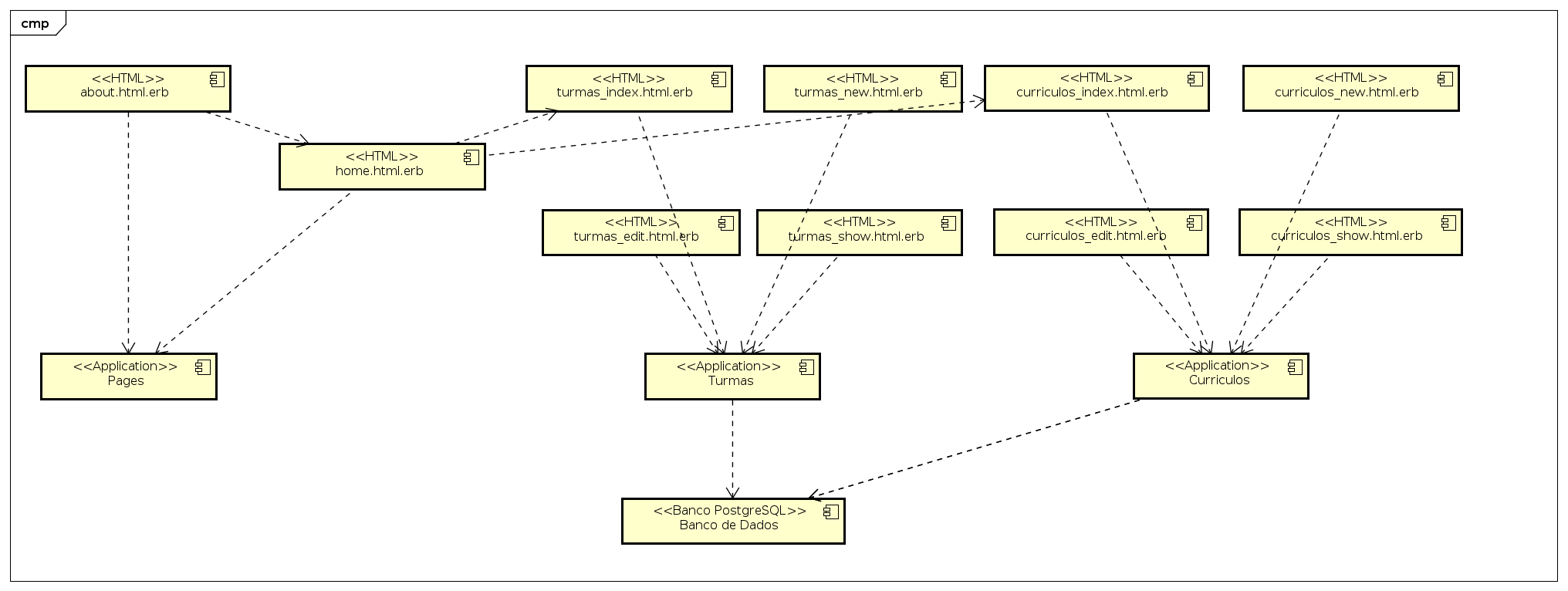


**Subsistemas**

**Físico**

****

**Componentes**

****

**Padrão**

**MVC**

**Overview**

MVC é um padrão de arquitetura de software ([design pattern](https://pt.wikipedia.org/wiki/Design_pattern)) que separa a representação da informação da interação do usuário com ele. As ideias centrais por trás do MVC são a [reusabilidade de código](https://pt.wikipedia.org/wiki/Reutiliza%C3%A7%C3%A3o_de_c%C3%B3digo) e [separação de conceitos](https://pt.wikipedia.org/wiki/Separa%C3%A7%C3%A3o_de_conceitos).

**Estrutura**

O modelo (model) consiste nos dados da aplicação, regras de negócios, lógica e funções. Uma visão (view) pode ser qualquer saída de representação dos dados, como uma tabela ou um diagrama. É possível ter várias visões do mesmo dado, como um gráfico de barras para gerenciamento e uma visão tabular para contadores. O controlador (controller) faz a mediação da entrada, convertendo-a em comandos para o modelo ou visão.

**Comportamento**



**Requirement realizations**

**[Realization1]**

**View of participants**

[Describe the participating design elements from a static perspective, giving details such as behavior, relationships, and attributes relevant to this realization.]

**Basic scenario**

[For the main flow, describe how instances of the design elements collaborate to realize the requirements. When using UML, this can be done with collaboration diagrams (sequence or communication).]

**Additional scenarios**

[For other scenarios that must be described to convey an appropriate amount of information about how the requirement behavior will be realized, describe how instances of the design elements collaborate to realize the requirement. When using UML, you can do this with collaboration diagrams (sequence or communication).]