SIGAR



Documento de Arquitetura

There is guidance within this template that appears in a style named InfoBlue. This style has a hidden font attribute that allows you to toggle whether it is visible or hidden in this template. Use the Microsoft® Word® menu **Tools > Options > View > Hidden Text** check box to toggle this setting. There is also an option for printing: **Tools > Options > Print**.

# Propósito

Este documento oferece uma visão geral arquitetural abrangente do **Sistema de Gerenciamento de Aulas de Reforço**, usando diversas visões arquiteturais para representar diferentes aspectos do sistema, identificando qual caso de uso foi utilizado à implementação da arquitetura e como serão atendidas as regras do negócio. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

# Objetivos e Filosofia da Arquitetura

A arquitetura do software é a estrutura, ou as estruturas, do sistema à ser implementado, a sua composição é feito por componentes do software, propriedades que são visíveis destes componentes e como os componentes se relacionam.

A arquitetura é parte importante do projeto do sistema, SIGAR, pois ela facilita a comunicação da equipe de desenvolvimento com o de construção, além de antecipação dos aspectos de decisões da infraestrutura do sistema, além de permitir uma transferência de abstração de entendimento de ambas as equipes.

# Suposições e Dependência

A arquitetura do SIGAR depende de alguns componentes:

* Conhecimento em programação orientada a objetos
* Conhecimento em programação web.

# Requisitos Significantes para Arquitetura

Para implementação da arquitetura, foi escolhido o requisito de Manter cadastro de aluno. Este requisito é onde está a implementação de inserção de um novo aluno, modificação de aluno existente e exclusão segura de um aluno inserido dentro da base de dados.

# Decisões, Limitações e Justificativas

* Sistema WEB

A decisão de utilizar uma linguagem web é a facilidade de visualização, porém uma limitação é o conhecimento da equipe sobre as linguagens a ser usada. Apesar de essa limitação ser um entrave durante o desenvolvimento, porém a utilização desta linguagem facilitaria a implementação de alguns componentes que foram listados durante a fase de elicitação de requisitos.

# Mecanismos da Arquitetura

A arquitetura do sistema é composta de alguns mecanismos, que no modelo adotado são as camadas. As camadas são conjunto de entidades ou classes que são comuns (mesmo propósito), as camadas são:

## Model

A camada *Model* é aonde será implementada a lógica de negócio que contem as classes de entidades em arquivo PHP (*Hypertext Preprocessor*).

## View

A camada *View* é a camada responsável por toda a visualização do sistema, nela aonde se encontra as páginas HTML (com recursos do CSS – *Cascading Style Sheets*), juntamente com recursos da linguagem *javascript*.

## Control

A camada de *Control* é nela que será criado objetos das classes que estão implementadas na *Model* juntamente com o tratamento de exceção que o sistema deverá fazer a fim de garantir a segurança de dados e a boa usabilidade, dentro desta camada é onde se encontra os arquivos *PHP*

## DAO

Adicionalmente, colocou-se a camada DAO que é um padrão de projeto que permite criar as classes de dados independentemente da fonte de dados ser um BD relacional, um arquivo texto, um arquivo XML, etc.

Para isso, ele encapsula os mecanismos de acesso a dados e cria uma interface de cliente genérica para fazer o acesso aos dados permitindo que os mecanismos de acesso a dados sejam alterados independentemente do código que utiliza os dados.

# Principais Abstrações

A arquitetura parte do estilo MVC, e para bom entendimento é necessário um pequeno conhecimento em algumas tecnologias que serão implementadas dentro da arquitetura, dentre quais:

* MVC (*Mode-View-Control*): estilo arquitetural escolhido a ser implementado.
* POO (Programação Orientada a Objetos): estilo de programação escolhido para implementação do sistema.
* PHP (*Hypertext Preprocessor*): linguagem de programação escolhida para implementar a lógica do sistema.
* *JavaScript*: linguagem de programação que faz as validações.
* *HTML (Hypertext Markup Language*): linguagem implementada para interface gráfica.
* *CSS (Cascading Style Sheet)*: implementação da interface gráfica

# Camadas ou *frameworks* da arquitetura

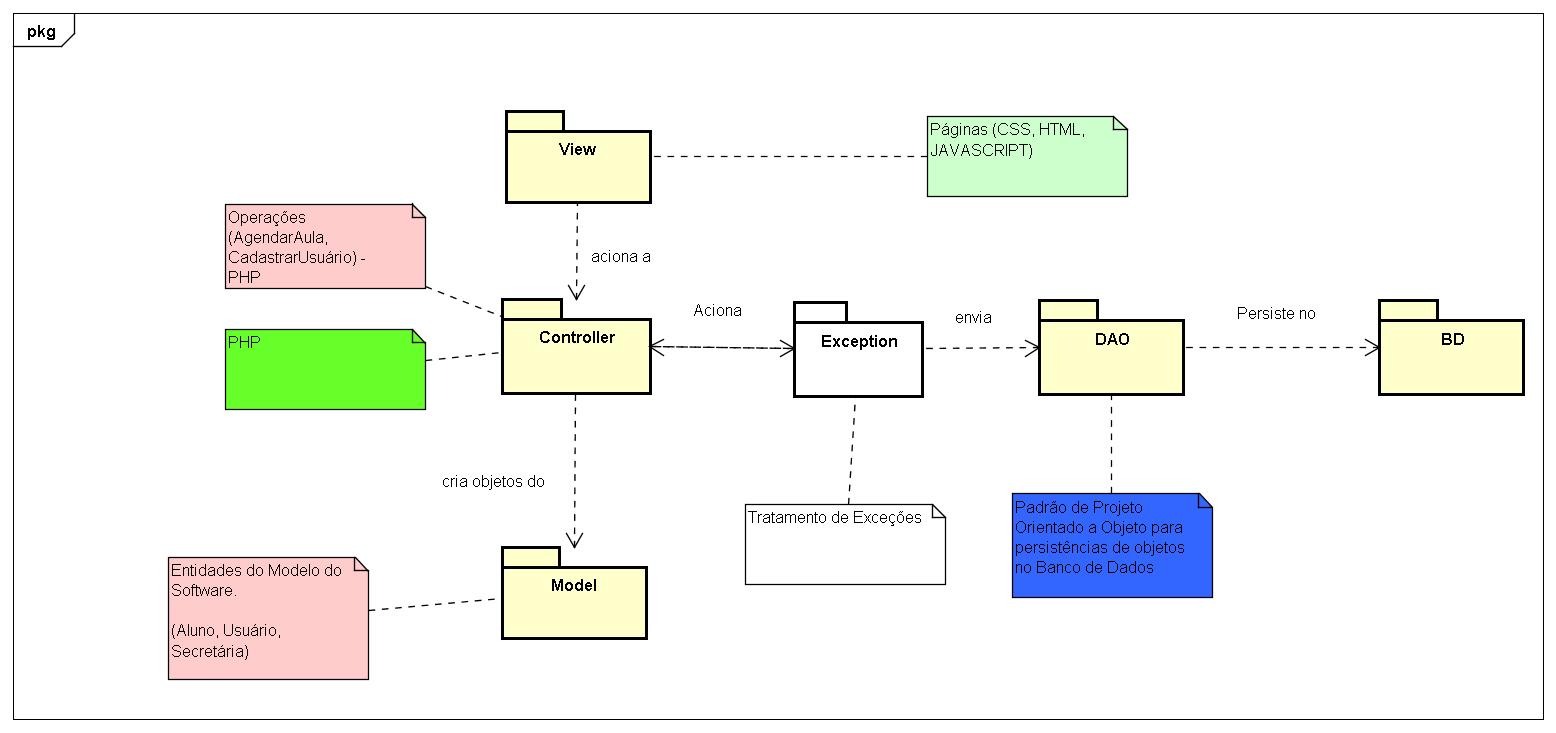
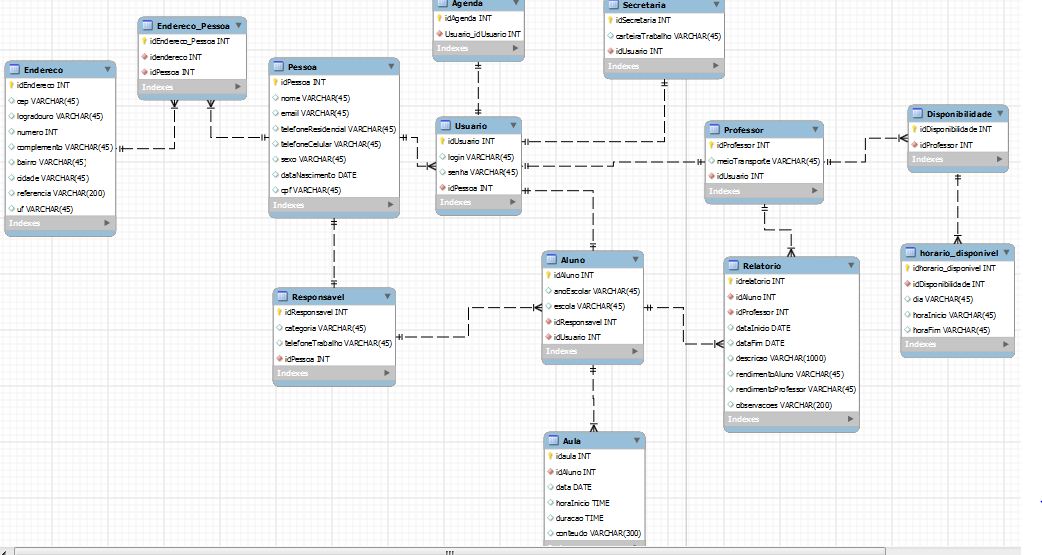
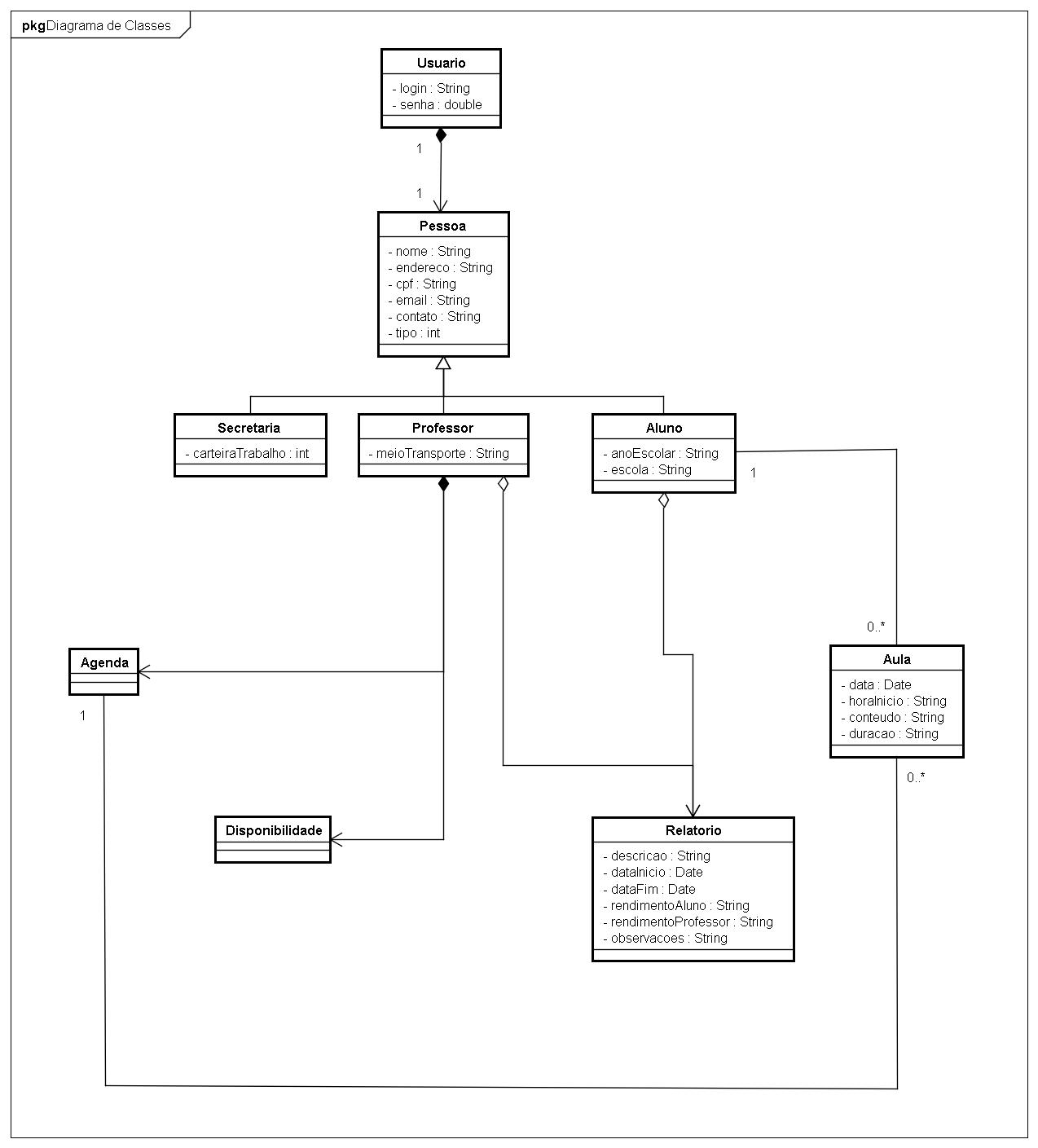


Figura 1- Visão das Camadas

As camadas que deverão ser implementadas pela arquitetura são as camadas dentro do MVC (*Model-View-Control*), aonde deverá ter um tratamento de erro dentro do controle a fim de garantir um software seguro para o usuário utilizar. Adicionalmente a DAO para persistência de dados.

# Visões Arquiteturais

* **Lógica:** A visão lógica parte de 3 diagramas para ser entendido. Um diagrama de modelagem de banco de dados e além de um modelo de entidade e relacionamento entre as entidades do banco, além do diagrama de classe.
  + **MER**
  + **DER**
  + **DIAGRAMA DE CLASSES**

****

* **Casos de Uso:**
  + Módulo de Cadastro

