

UNIVERSIDADE DO CENTRO OESTE DE SANTA CATARINA – CAAMPUS VIDEIRA

ALUNOS: JOÃO VITOR BORGES GODINHO, JULIO ORSO, JULIANNA ORSO

PROFESSOR: Leandro Otavio Cordova Vieira.

DISCIPLINA: PROGRAMAÇÃO III

## RELATÓRIO PROJETO FINAL – FRAMEWORKS PHP (LAMINAS)

### 1. Introdução ao Laminas (Zend Framework)

O Laminas é um framework PHP robusto e modular, conhecido por sua flexibilidade e componentes reutilizáveis. Ele permite o desenvolvimento ágil de aplicativos web escaláveis, seguindo boas práticas de desenvolvimento.

### 2. Desenvolvimento

#### 2.1.1. Configuração do Banco de Dados

Instalação do MySQL: Configuramos um servidor MySQL local ou remoto para armazenar os dados da aplicação. O MySQL é amplamente suportado pelo Laminas e oferece robustez e desempenho para aplicações web.

Configuração do Adaptador de Banco de Dados: Utilizamos o adaptador ``Laminas\Db\Adapter\Adapter`` para estabelecer uma conexão com o banco de dados MySQL. Isso envolve configurar parâmetros como host, nome do banco de dados, usuário e senha no arquivo de configuração do Laminas (``module.config.php``).

#### 2.1.2. Criação de Tabelas e Esquema

Definição de Estrutura: Utilizamos o SQL para definir a estrutura das tabelas, incluindo campos, tipos de dados, chaves primárias, estrangeiras e índices necessários para otimização de consultas.

Integração com Laminas: Usamos o componente ``Laminas\Db\TableGateway\TableGateway`` para interagir com as tabelas do banco de dados de forma eficiente. Isso inclui operações como inserção, atualização, exclusão e consulta de dados.

#### 2.1.3. Migrations e Versionamento

Para facilitar a gestão do esquema do banco de dados ao longo do tempo e entre diferentes ambientes de desenvolvimento, consideramos o uso de ferramentas de migração de banco de dados.

#### 2.1.4. Segurança e Boas Práticas

Adotamos boas práticas de segurança ao lidar com o banco de dados, como:

- Prevenção contra SQL Injection: Utilizamos prepared statements e parâmetros vinculados ao executar consultas SQL para mitigar riscos de injeção de SQL.
- Gerenciamento de Credenciais: Mantemos informações sensíveis, como senhas de banco de dados, seguras e fora do controle de versão usando arquivos de configuração protegidos.

#### 2.1.5. Integração com o Laminas

Integramos o banco de dados MySQL com o Laminas através de seus componentes de banco de dados, garantindo que as operações de CRUD.

Claro, vou expandir um pouco mais sobre os principais tópicos discutidos na nossa conversa sobre desenvolvimento com o Laminas (anteriormente conhecido como Zend Framework).

### 2.2. Configuração do Projeto

Utilizamos o Composer para inicializar e gerenciar o projeto Laminas. O Composer facilita a instalação de dependências e pacotes necessários para o desenvolvimento, incluindo o próprio Laminas e suas extensões.

#### 2.2.1. Criação de Formulários

A criação de formulários no Laminas é simplificada pelo componente ``Laminas\Form``, que oferece métodos para definição de campos, validação de entrada de dados e geração de formulários HTML. Exemplos incluem formulários como ``UserLoginForm`` e ``UserRegisterForm``, onde discutimos estratégias para validação de dados e exibição de mensagens de erro.

#### 2.2.2. Autenticação e Autorização

Implementamos um sistema de autenticação usando ``AuthenticationService`` do Laminas. Utilizamos adaptadores como ``DbTable\CredentialTreatmentAdapter`` para autenticar usuários com dados armazenados em um banco de dados MySQL. Gerenciamos sessões usando ``Session\Container`` para armazenar informações de autenticação de forma segura.

#### 2.2.3. Controladores e Rotas

Criamos controladores como ``LoginController`` e ``IndexController``, que são responsáveis por gerenciar a lógica de negócios da nossa aplicação. Configuramos rotas usando o sistema de roteamento do Laminas para direcionar solicitações HTTP para os controladores e suas respectivas ações.

#### 2.2.4. Views e Templates

Utilizamos o ``Laminas\View\Renderer\PhpRenderer`` para renderizar templates e views. Criamos templates como ``register.phtml`` para exibir formulários de registro e utilizamos helpers do Laminas para gerar URLs e links dentro dos templates.

#### 2.2.5. Erros e Problemas Comuns

Durante o desenvolvimento, enfrentamos e resolvemos diversos problemas comuns, como erros de sintaxe, classes não encontradas e problemas de configuração de rotas e principalmente, compatibilidade entre as versões dos componentes.

### 3. Considerações Finais

O Laminas é um framework bastante robusto e capaz de proporcionar ferramentas mais que suficientes pra desenvolvimento do projeto. Apesar disso, muito do conteúdo disponível sobre ele na internet tem que ser adquirido diretamente da documentação oficial, o que por muitas vezes, tornou a informação de difícil acesso. A curva de aprendizagem é longa e requer tempo. O projeto ainda deve ser finalizado.