

Lance Livre

ENGENHARIA DE SOFTWARE

Etapas 2

Jean Argoud, Luis Filipe Rodrigues e Vitor Pedrollo

Lancelivre: Resumo do negócio

A plataforma Lance Livre consiste em uma ferramenta para divulgação e procura de serviços como freelancer em diversas áreas. A plataforma se trata de uma aplicação web, que tem como objetivo encontrar trabalhadores autônomos e facilitar a comunicação entre as partes para chegarem a um acordo sobre a prestação de serviços.

Desenvolvimento

- **Arquitetura escolhida:** MVC

Para esta implementação, também poderia ser interessante utilizar a Arquitetura de Repositório, já que ela oferece a vantagem de trocar para um sistema réplica em caso de falhas. No entanto, o grupo de desenvolvedores considerou que para esta aplicação não seria um grande problema ficar fora do ar por um certo período de tempo, então, por conta disso e da familiaridade do grupo, optamos por MVC.

- **Linguagem de programação:** PHP

Além de HTML e CSS para o frontend.

É possível que no futuro precisemos usar JavaScript, pois o PHP não executa assíncronamente.

- **Framework utilizado (até o momento):** Yii 2

Framework baseado em MVC e bastante utilizado no mercado de trabalho para desenvolver websites, ou seja existe bastante documentação online para ajudar a desenvolver nele.

Desenvolvimento

- **Banco de dados:** Postgres

Banco de dados baseado em tabelas relacionais similar a outros como “MySQL”.

- **Servidor web:** Apache

Ferramenta utilizada para o projeto ser acessado pelo navegador em forma de site.

- **Repositório:** Github

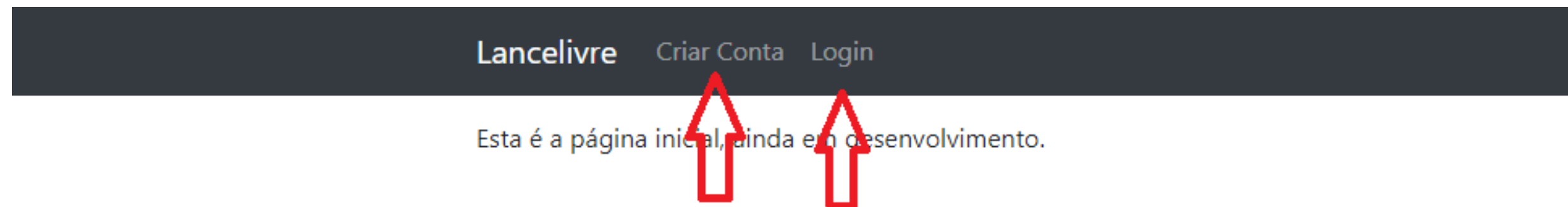
Repositório com todo o código desenvolvido até então.
Encontra-se aberto para qualquer um baixar.

Implementação

Foram implementados até agora 2 casos de uso:

- **Cadastrar uma conta (criar conta):**
- **Iniciar sessão (login):**

Ambas podem ser encontradas em uma barra de navegação:



Caso de uso: Cadastrar uma conta

[Lancelivre](#) [Criar Conta](#) [Login](#)

Nome de Usuário

Nome cannot be blank.

Senha

Senha cannot be blank.

Email

Email cannot be blank.

Criar

Tanto os campos “Nome do usuário”, “Senha” e “Email” São obrigatórios para poder criar a conta.

Caso de uso: Iniciar sessão

Lancelivre [Criar Conta](#) [Login](#)

Login

Please fill out the following fields to login:

Email



Email cannot be blank.

Senha



Senha cannot be blank.

☒ Remember Me

Login

A conta dos usuários é salva pela chave primária do “email”,
ou seja cada email só pode estar vinculado a uma conta.

DIAGRAMA DE PACOTES



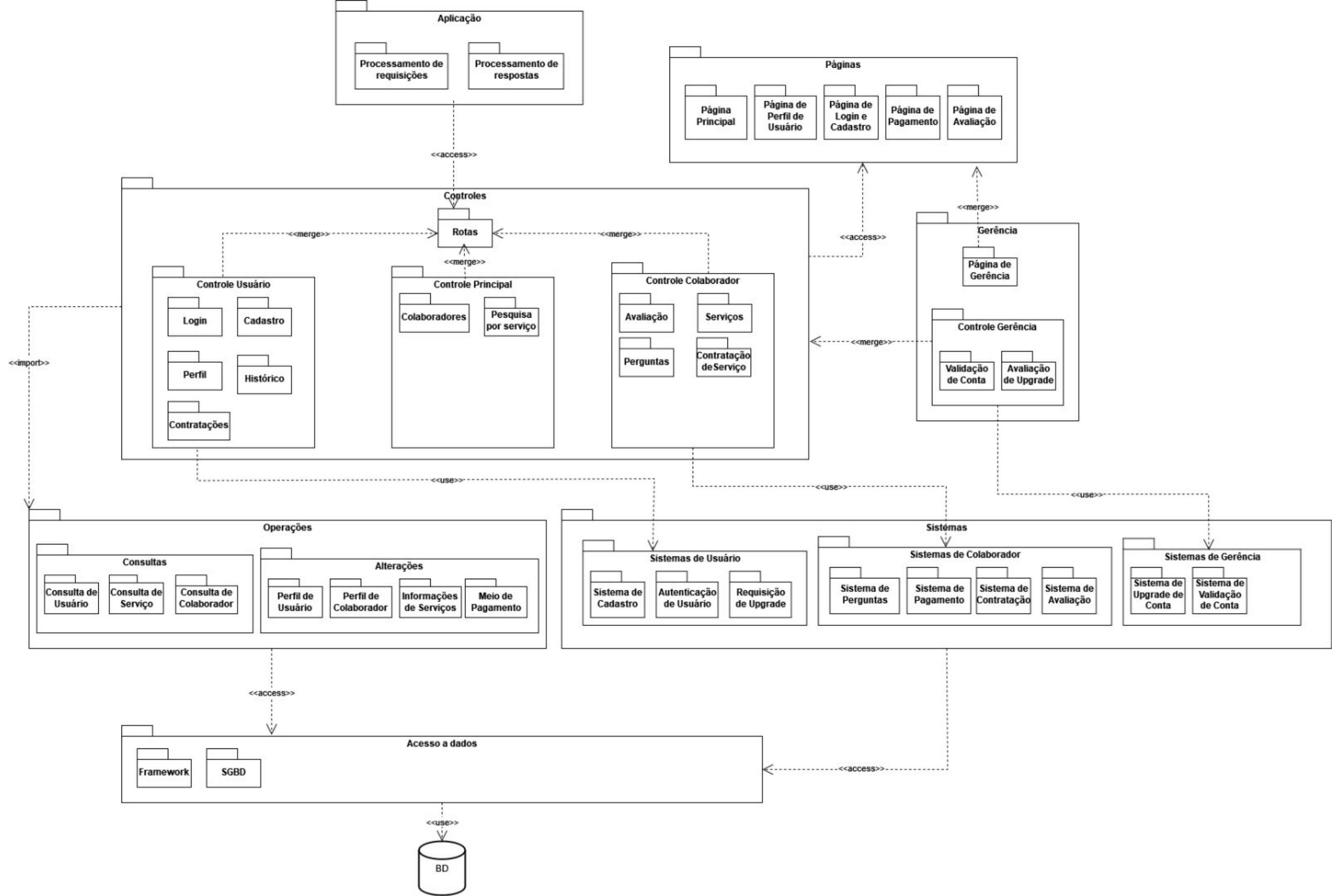


Diagrama de Pacotes

Identificação do pacote: Aplicação (1)

Função: Pacote onde fica o ponto de entrada da aplicação. Faz o processamento de requisições HTTP e das respostas.

Subpacotes: 1.1 e 1.2

Identificação do pacote: Processamento de requisições (1.1)

Função: Realiza o processamento de requisições, determinando as rotas que a aplicação deve seguir para obter os recursos que o usuário solicita.

Identificação do pacote: Processamento de respostas (1.2)

Função: Realiza o processamento da resposta, enviando o recurso solicitado ao cliente.

Identificação do pacote: Controles (2)

Função: Pacote que contém os controles da aplicação. Os controles são os endpoints especificados pelas rotas e realizam certo processamento nos dados, além de chamar funções de preenchimento de páginas web e enviá-las à aplicação.

Subpacotes: 2.1, 2.2 e 2.3

Identificação do pacote: Controle Usuário (2.1)

Função: Controles referentes ao usuário, com funções de Login, Cadastro, Perfil, Histórico e Contratações.

Identificação do pacote: Controle Principal (2.2)

Função: Controle principal da loja, com funções de mostrar Colaboradores e pesquisa de serviços.

Diagrama de Pacotes

Identificação do pacote: Controle Usuário (2.1)

Função: Controles referentes ao usuário, com funções de Login, Cadastro, Perfil, Histórico e Contratações.

Identificação do pacote: Controle Principal (2.2)

Função: Controle principal da loja, com funções de mostrar Colaboradores e pesquisa de serviços.

Identificação do pacote: Controle Colaborador (2.3)

Função: Controles referentes ao Colaborador, com funções de Avaliação, Serviços, Perguntas e Contratação de Serviço.

Identificação do pacote: Páginas (3)

Função: Contém as páginas da aplicação, funções de carregamento e alteração das mesmas com conteúdo dinâmico.

Identificação do pacote: Operações (4)

Função: Contém operações utilizadas pelos controles da aplicação, como consultas à base de dados e alterações de dados.

Subpacotes: 4.1 e 4.2

Identificação do pacote: Consultas (4.1)

Função: Consultas que podem ser feitas à base de dados, como Consulta de Usuário, Consulta de Serviço e Consulta por Colaborador.

Diagrama de Pacotes

Identificação do pacote: Alterações (4.2)

Função: Alterações de dados como Perfil de Usuário, Perfil de Colaborador, Informações de Serviços e Meios de Pagamento.

Identificação do pacote: Sistemas (5)

Função: Contém uma série de subsistemas utilizados pela aplicação, como sistemas de Usuário, Colaborador e Gerência.

Subpacotes: 5.1, 5.2 e 5.3

Identificação do pacote: Sistemas de Usuário (5.1)

Função: Contém os subsistemas referentes ao Usuário: Sistema de Cadastro, Autenticação de Usuário e Requisição de Upgrade.

Subpacotes: 5.1.1, 5.1.2 e 5.1.3

Identificação do pacote: Sistemas de Colaborador (5.2)

Função: Contém os subsistemas referentes ao Colaborador: Sistema de Perguntas, Sistema de Pagamento, Sistema de Contratação e Sistema de Avaliação.

Identificação do pacote: Sistemas de Gerência (5.3)

Função: Contém os subsistemas referentes à Gerência, como Sistema de Upgrade de Conta e Sistema de Validação de Conta.

Identificação do pacote: Sistemas de Gerência (5.3)

Função: Contém os subsistemas referentes à Gerência, como Sistema de Upgrade de Conta e Sistema de Validação de Conta.

Diagrama de Pacotes

Identificação do pacote: Acesso a dados (7)

Função: Contém os pacotes de subsistemas de acesso à base de dados, como um sistema gerenciador de banco de dados e um framework contendo as funcionalidades de acesso