

DEVinHouse

Módulo 1 - Projeto Avaliativo 2

SUMÁRIO

1 MOTIVAÇÃO	1
2 REQUISITOS DA APLICAÇÃO	1
3 EXEMPLO DE APLICAÇÃO	5
4 CRITÉRIOS DE AVALIAÇÃO	10
5 ENTREGA	13
6 PLANO DE PROJETO	13

1 MOTIVAÇÃO

A **LABMedicine LTDA**, empresa líder no segmento tecnológico para gestão hospitalar, está com um projeto novo intitulado **LABMedical**, um software audacioso para gestão de inventário médico. O seu perfil chamou a atenção dos gestores, para criar a aplicação Front-End do software, que deverá ser construída utilizando o *framework* **Angular**.

2 REQUISITOS DA APLICAÇÃO

A aplicação que deverá ser realizada **individualmente** deve contemplar os seguintes requisitos:

1. Ser construída utilizando o *framework* Angular:
 - a. Utilizar roteamento para gerenciamento de páginas.
 - b. Gerenciar o estado global quando necessário.
 - c. Utilizar animações de loading e transições entre páginas.
 - d. Utilizar consumo da API [ViaCEP](#) para cadastro de endereço.
2. Utilizar favicon, título de página e demais assets.
3. Utilizar o localStorage ou JSON Server para guardar as informações cadastradas.
4. Utilizar o GitHub como versionador de código:
 - a. Utilização do padrão baseado em GitFlow com b main, features e releases.
 - b. Utilização de commits curtos e concisos.

5. Utilizar Trello para organização das tarefas a serem realizadas:
 - a. Criar um board com estrutura kanban (backlog, todo, doing, blocked, review e done) para organização das tarefas.
 - b. Criar cards com tarefas a serem realizadas e arrastar cada um conforme a necessidade.
6. Possuir as seguintes páginas e funcionalidades:
 - a. **Página de Login:** Contendo um formulário com campos de email e senha, botões para criar uma conta, entrar na aplicação, e um link para resetar a senha.
 - i. O botão de criar conta deve abrir uma página ou modal com formulário para criar um usuário, sendo exigido e-mail válido e senha com mais de 8 caracteres. Para a senha utilize dois campos e valide se a mesma é equivalente. Salve o usuário no localStorage (ou utilize JSON Server).
 - ii. A funcionalidade de "esqueceu sua senha" não precisa ser implementada. Caso não seja implementada, utilize um *pop-up*, *toast* ou *alert* para avisar que esta funcionalidade está em construção.
 - iii. O botão de entrar deve direcionar o usuário para a página de Início, caso o e-mail e senha sejam válidos e estejam cadastrados.
 - b. **Menu Lateral:** Com exceção da página de login, deverá estar presente em todas as outras páginas, apresentando todos os botões de navegação.
 - i. O botão de sair deve 'deslogar' e direcionar o usuário para a página de login.
 - ii. Deverá existir um botão para esconder e mostrar o menu lateral.
 - c. **Toolbar:** Com exceção da página de login, deverá estar presente em todas as outras páginas, apresentando o título da página, nome do usuário (médico) e ícone para representar a foto do usuário.
 - i. Opcionalmente poderá ser adicionado outras funcionalidades como *dropdown* e botão de sair, por exemplo.
 - d. **Página de Início (Estatísticas e Informações):** Deve conter as estatísticas do sistema e busca de pacientes.
 - i. As estatísticas devem ser ao menos a quantidade de pacientes cadastrados, quantidade de consultas cadastradas e a quantidade de exames. Você está livre para adicionar novas estatísticas caso ache necessário.
 - ii. Deverá existir um campo para buscar pacientes pelo nome, telefone ou e-mail.
 - iii. A listagem de pacientes deve mostrar um *card* para cada um com ao menos um ícone, nome completo, idade, contato (telefone ou e-mail), plano de saúde e botão para ver mais informações.
 - iv. Ao clicar no *card* (ou em "ver mais"), a página de cadastro de paciente deve ser aberta, com os botões de edição, deleção e salvamento ativados. Nesta página, o usuário (médico) poderá editar as informações ou deletar um paciente caso o mesmo não possua consultas e exames vinculados.

- e. Página de Cadastro de Paciente:** Deve conter um formulário para cadastro de paciente com botões para editar, deletar e salvar.
- i. Durante o cadastro, os botões de editar e deletar devem ficar desativados (desabilitados).
 - ii. Campos do formulário:
 - 1. Nome Completo: Obrigatório, com máximo e mínimo de 64 e 8 caracteres, respectivamente.
 - 2. Gênero: Obrigatório com *dropdown* de opções pré-definidas.
 - 3. Data de Nascimento: Obrigatório, data válida.
 - 4. CPF: Obrigatório com o formato 000.000.000-00
 - 5. RG com órgão expedidor: Obrigatório, com máximo de 20 caracteres.
 - 6. Estado Civil: Obrigatório com *dropdown* de opções pré-definidas.
 - 7. Telefone: Obrigatório com o formato (99) 9 9999-9999
 - 8. E-mail: Não obrigatório e com validação.
 - 9. Naturalidade: Obrigatório, com máximo e mínimo de 64 e 8 caracteres, respectivamente.
 - 10. Contato de Emergência: Obrigatório com o formato (99) 9 9999-9999
 - 11. Lista de Alergias: Não obrigatório.
 - 12. Lista de Cuidados Específicos: Não obrigatório.
 - 13. Convênio: Não obrigatório.
 - 14. Número do Convênio: Não obrigatório.
 - 15. Validade do Convênio: Não obrigatório.
 - 16. Endereço: Cep, Cidade, Estado, Logradouro, Número, Complemento, Bairro e Ponto de Referência.
 - iii. Deverá utilizar a API do ViaCEP para buscar os dados de endereço.
 - iv. Deverá verificar os dados informados antes de cadastrar.
 - v. Deverá criar um identificador único para cada paciente cadastrado.
 - vi. Deverá apresentar animação ao salvar.
- f. Página de Cadastro de Consulta:** Deve conter uma busca de paciente e um formulário para cadastro de consulta com botões editar, deletar e salvar.
- i. Durante o cadastro, os botões de editar e deletar devem ficar desativados (desabilitados).
 - ii. Campos do formulário:
 - 1. Motivo da consulta: Obrigatório, com máximo e mínimo de 64 e 8 caracteres, respectivamente.
 - 2. Data da Consulta: Obrigatório, buscando data atual do sistema e liberando para edição.
 - 3. Horário da Consulta: Obrigatório, buscando horário atual do sistema e liberando para edição.

- 4. Descrição do Problema: Obrigatório, com máximo e mínimo de 1024 e 16 caracteres, respectivamente.
 - 5. Medicação Receitada: Não obrigatório.
 - 6. Dosagem e Precauções: Obrigatório, com máximo e mínimo de 256 e 16 caracteres, respectivamente.
- iii. Deverá verificar os dados informados antes de cadastrar.
- iv. Deverá criar um identificador único para cada consulta cadastrada.
- v. Deverá apresentar animação ao salvar.
- g. **Página de Cadastro de Exame:** Deve conter uma busca de paciente e um formulário para cadastro de consulta com botões editar, deletar e salvar.
 - i. Durante o cadastro, os botões de editar e deletar devem ficar desativados (desabilitados).
 - ii. Campos do formulário:
 - 1. Nome do Exame: Obrigatório, com máximo e mínimo de 64 e 8 caracteres, respectivamente.
 - 2. Data do Exame: Obrigatório, buscando horário atual do sistema e liberando para edição.
 - 3. Horário do Exame: Obrigatório, buscando horário atual do sistema e liberando para edição.
 - 4. Tipo do Exame: Obrigatório, com máximo e mínimo de 32 e 4 caracteres, respectivamente.
 - 5. Laboratório: Obrigatório, com máximo e mínimo de 32 e 4 caracteres, respectivamente.
 - 6. URL do Documento: Não obrigatório.
 - 7. Resultados: Obrigatório, com máximo e mínimo de 1024 e 16 caracteres, respectivamente.
 - iii. Deverá verificar os dados informados antes de cadastrar.
 - iv. Deverá criar um identificador único para cada exame cadastrado.
 - v. Deverá apresentar animação ao salvar.
- h. **Página de Listagem de Prontuário:** Deverá possuir uma barra de pesquisa, listagem de pacientes cadastrados e botão para abrir os detalhes.
 - i. A barra de busca deverá buscar o paciente pelo nome ou identificador cadastrado.
 - ii. Ao clicar no *card* do paciente, uma nova página deve ser aberta mostrando as informações cadastradas, e o histórico de consultas e exames.
- i. **Página de Prontuário de Paciente:** Deverá possuir informações concisas e histórico de consultas e exames.
 - i. As informações a serem exibidas são: nome do paciente, convênio, contato de emergência, lista de alergias e lista de cuidados específicos.

- ii. O histórico de exames e consultas deve mostrar todas as consultas e exames realizados pelo paciente, sendo mostrados em ordem cronológica.
- iii. Nas consultas, deve haver todas as informações cadastradas, dando destaque ao motivo, data e hora. Também deve haver um botão de editar, que quando clicado, abrirá a página de cadastro da consulta, com os botões de edição, deleção e salvamento ativados. Nesta página, o usuário (médico) poderá editar as informações ou deletar uma consulta.
- iv. Nos exames, deve haver todas as informações cadastradas, dando destaque ao nome, laboratório, data, hora e anexo. Também deve haver um botão de editar, que quando clicado, abrirá a página de cadastro de exame, com os botões de edição, deleção e salvamento ativados. Nesta página, o usuário (médico) poderá editar as informações ou deletar um exame.

3 EXEMPLO DE APLICAÇÃO

A aplicação deverá conter os requisitos apresentados anteriormente, sendo codificada em **html**, **css** e **javascript** através do *framework* **Angular**, também deverá ser utilizado **markdown** para o readme.md.

As imagens a seguir demonstram exemplos da aplicação que deverá ser desenvolvida. Você poderá desenvolver igual ao modelo, ou criar um layout diferente com estilo próprio.

Visualização do protótipo: [LABMedical Project](#)



<LAB365>

Não possui uma conta?

[Criar Conta](#)

Login

E-Mail

Senha

[Entrar](#)

[Esqueceu sua senha?](#)



LABMedical

Imagem 1: Página de Login



LABMedical

ESTATÍSTICAS E INFORMAÇÕES

Cesar Abascal 

Genal

[INICIO](#)

[SAIR](#)

Pacientes

[+ CADASTRAR](#)

[LISTAR PRONTUARIO](#)

Exames

[+ CADASTRAR CONSULTA](#)

[+ CADASTRAR EXAME](#)

Encolher Menu 

Estatísticas do Sistema


 70 Pacientes	 104 Consultas	 82 Exames
--	---	---

Informações Rápidas de Pacientes

[Buscar](#)

 Cesar Abascal 30 Anos (55) 9 8224-4534 Unimed Ver mais	 Andressa Lima 29 Anos (48) 9 9112-3486 Sem Plano Ver mais	 Saul Souza 7 Anos (48) 9 9112-3486 Plamev Ver mais
 Fernando Burtal	 Felipe Muck	 Luiz Lima

Imagem 2: Página de Início (Estatística e Informações)



LABMedical

Genal

INICIO

SAIR

Pacientes

+ CADASTRAR

LISTAR PRONTUÁRIO

Exames

+ CADASTRAR CONSULTA

+ CADASTRAR EXAME

Encolher Menu

CADASTRO DE PACIENTE

Cesar Abascal

Preencha os campos para cadastrar

Identificação

Editar Deletar Salvar

Nome Completo Gênero Data Nascimento

CPF RG Estado Civil

Telefone E-Mail Naturalidade

Convênio

convênio Número da Carteira Validade


Dados de Endereço

CEP Cidade Estado

Logradouro Número

complemento Bairro Ponto de Referência

Imagem 3: Página de Cadastro de Paciente



LABMedical

Genal

INICIO

SAIR

Pacientes

+ CADASTRAR

LISTAR PRONTUÁRIO

Exames

+ CADASTRAR CONSULTA

+ CADASTRAR EXAME

Encolher Menu

CADASTRO DE CONSULTA

Cesar Abascal

Encontre o paciente

Q Digite o nome do paciente Buscar

consulta de [Nome Paciente]

Editar Deletar Salvar


Motivo da Consulta Data da Consulta Hora da Consulta

Descrição do Problema

Medicação Receitada


Dosagem e Precauções

Imagem 4: Página de Cadastro de Consulta



LABMedical

CADASTRO DE EXAME

Cesar Abascal 

Encontre o paciente

Exame de [Nome Paciente]

Editar
Deletar
Salvar

Nome do Exame

Data da Exame

Hora da Exame

Tipo do Exame


Laboratório

URL do Documento do Exame

Resultado do Exame


Encolher Menu

Imagem 5: Página de Cadastro de Exame



LABMedical

LISTAGEM DE PRONTUÁRIOS

Cesar Abascal 

Utilize a barra de pesquisa para buscar

Registro	Nome do Paciente	Convênio	
0000000001	Cesar Abascal	Unimed	>
0000000002	Andressa Lima	Sem Plano	>
0000000003	Saul Souza	Plamev	>
0000000004	Fernando Puntel	Unimed	>
0000000005	Elon Musk	Bradesco Saúde	>
0000000006	Luan Lima	Sul América	>
0000000007	Bill Gates	Cauzzo	>

Encolher Menu

Imagem 6: Página de Listagem de Prontuários de Paciente

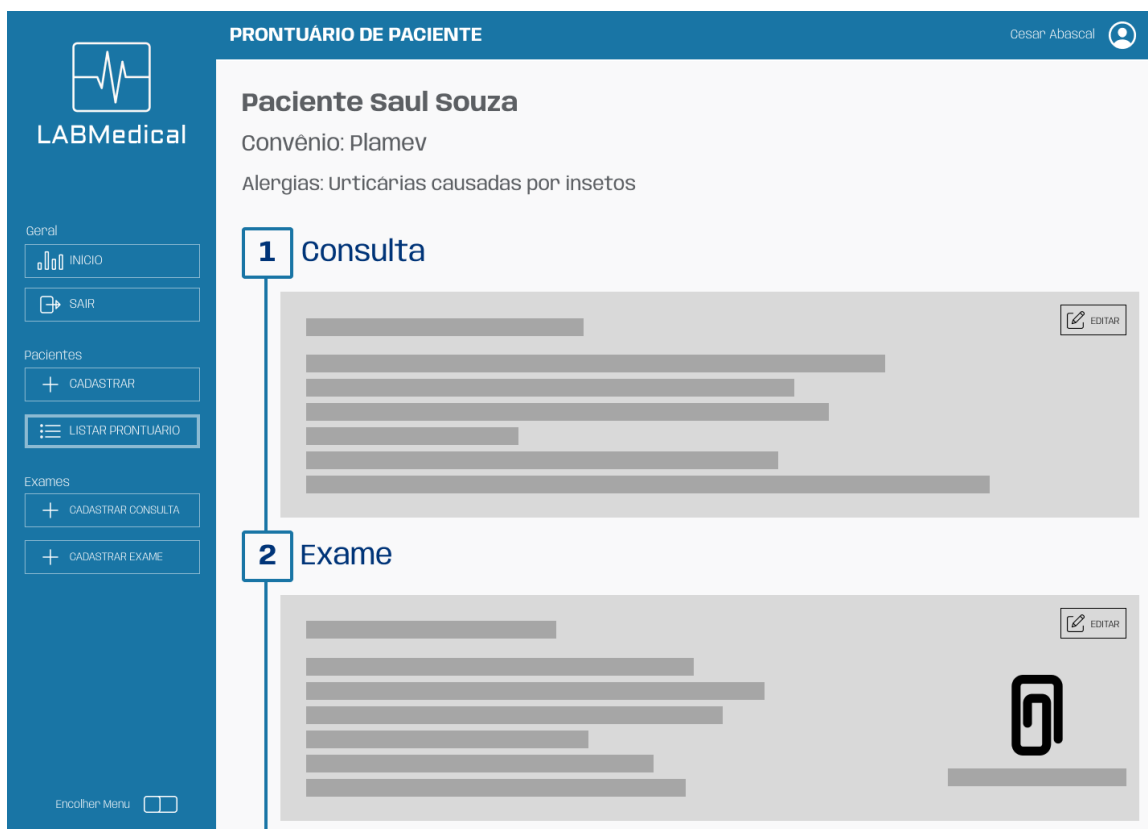


Imagem 7: Página de Prontuário de Paciente

4 CRITÉRIOS DE AVALIAÇÃO

A tabela abaixo apresenta os critérios que serão avaliados durante a correção do projeto. O mesmo possui variação de nota de 0 (zero) a 10 (dez) como nota mínima e máxima, e possui peso de **40% sobre a avaliação do módulo**.

Serão **desconsiderados e atribuída a nota 0 (zero)** os projetos que apresentarem plágio de soluções encontradas na internet ou de outros colegas. Lembre-se: Você está livre para utilizar outras soluções como base, mas **não é permitida** a cópia.

Uso adequado do Trello				
Nº	Critério de Avaliação	0	0,25	0,5
1	Aluno atribuiu adequadamente seu nome no card, bem como especificações da tarefa?	O aluno não criou/atribuiu o card.	Aluno criou o card, mas não colocou seu nome ou comentários especificados nos requisitos.	Aluno criou o card, atribuiu seu nome nele e colocou as especificações de implementação no mesmo.

2	O aluno movimentou adequadamente o card no board?	Aluno não movimentou o card no board.	Aluno movimentou o card de maneira inadequada no board.	O aluno movimentou adequadamente o card, movimentando 1 por vez, e respeitando as regras no Kanbanize.
---	---	---------------------------------------	---	--

Uso adequado do GitHub

Nº	Critério de Avaliação	0	0,25	0,5
3	As mensagens do commit estão claras e adequadas?	Os commits do aluno não tiveram nenhuma mensagem	Os alunos fizeram commits com comentários, mas esses estavam inadequados ou incompletos com relação ao conteúdo das alterações	As mensagens do commit estão claras e adequadas em relação ao conteúdo das alterações
4	Foi aberto um branch separado para cada etapa, funcionalidade?	Os commits foram realizados diretamente na master/main.	Foram abertas branches para cada funcionalidade, porém foram realizados commits de mesma funcionalidade em branches diferentes.	Foram abertas branches para cada funcionalidades, e os commits foram realizados de forma concisa nas mesmas.

Desenvolvimento adequado da Aplicação

Nº	Critério	0	0,20 a 0,60	0,80
5	Inovou, construindo um layout agradável, responsivo e com acessibilidade?	Não inovou e construiu um layout sem responsividade e acessibilidade.	Inovou, construindo um layout agradável, personalizado e responsivo.	Inovou, construindo um layout agradável, personalizado, responsivo e com acessibilidade.
6	Estruturou o projeto, separando os recursos de acordo com o padrão de projetos?	Não foi efetuado a separação das páginas e componentes e a criação do router e store estão sendo feitas direto no main.	Foi efetuado a separação parcial de páginas e componentes ou inicialização do router e store em arquivos separados, com poucos erros na padronização.	Foi efetuado a separação de views e componentes, criação da store e router em seus devidos diretórios, separando regras de negócios e criação, sempre que necessário, em arquivos específicos.

7	Desenvolveu a página de login com a funcionalidade de autenticação funcional e com devidos tratamentos no formulário. Além de conter todos os elementos descritos no requisitos como acesso à página de cadastro, login e esqueci a senha?	Não desenvolveu a página ou entregou sem estar funcional.	Desenvolveu a página com a autenticação funcional porém com falhas na validação do formulário, alertas ou falta de redirecionamento.	Desenvolveu a página de login com todos os elementos do requisito, autenticação funcional com redirecionamento, alertas de avisos e falhas, além criar o guard com redirecionamento para página inicial se usuário já estiver autenticado.
8	Criou o template da aplicação, contendo o Menu Lateral e Toolbar funcional de acordo com os requisitos?	Não desenvolveu o template para a aplicação.	Desenvolveu o template, porém sem o Menu Lateral ou Toolbar ou com algum dos elementos desses componentes não funcionais.	Desenvolveu o template utilizando Menu Lateral e Toolbar, com todas as funcionalidades de acordo com os requisitos e com o guard permitindo o acesso somente com autenticação.
9	Desenvolveu a página de início contendo as estatísticas, listagem de pacientes, pesquisa e acesso a um modal ou página de edição do paciente?	Não desenvolveu a página ou apenas efetuou a criação da página com dados estáticos.	Desenvolveu a página de início, com todas as estatísticas e listagem de pacientes.	Desenvolveu a página de início, com todas as estatísticas, listagem de pacientes, pesquisa e acesso a mais detalhes, edição e deleção.
10	Desenvolveu a página de cadastro de paciente de acordo com os requisitos, efetuando validação no formulário e armazenando os dados no localStorage (ou JSON Server)?	Não desenvolveu a página ou não está funcional.	Desenvolveu parcialmente a página ou com erros de validação, não salvando corretamente no localStorage ou não efetuando a busca de CEP com a API ViaCep.	Desenvolveu a página de cadastro de paciente funcional, com todas as validações, efetuando consulta na API, guardando os registros no localStorage e permitindo edição e deleção corretamente.

11	Desenvolveu a página de cadastro de consulta, guardando as informações no localStorage (ou JSON Server), efetuando as devidas validações de formulário?	Não desenvolveu a página ou não está funcional.	Desenvolveu a página de cadastro, armazenando a consulta no localStorage, porém com falhas na validação.	Desenvolveu a página de cadastro com todas as validações funcionais, armazenando a consulta no localStorage utilizando a regra de negócio, além de retornar para o usuário mensagem de conclusão.
12	Desenvolveu a página de cadastro de exame, guardando as informações no localStorage (ou JSON Server), efetuando as devidas validações de formulário?	Não desenvolveu a página ou não está funcional.	Desenvolveu a página de cadastro, armazenando o exame no localStorage, porém com falhas na validação.	Desenvolveu a página de cadastro com todas as validações funcionais, armazenando o exame no localStorage utilizando a regra de negócio, além de retornar para o usuário mensagem de conclusão.
13	Desenvolveu a página de listagem de prontuário com filtro?	Não desenvolveu a página de listagem ou entregou com dados estáticos.	Desenvolveu a página de listagem, porém não buscou os dados corretamente ou não aplicou o filtro.	Desenvolveu a página de listagem com todos os requisitos funcionais.
14	Desenvolveu a página de detalhes do prontuário?	Não desenvolveu a página ou não está funcional.	Desenvolveu a página, porém a mesma não possui os botões de edição e anexos.	Desenvolveu a página totalmente funcional, seguindo todas as especificações solicitadas.

5 ENTREGA

O código desenvolvido deverá ser submetido no **GitHub**, e o link deverá ser disponibilizado junto com o link do **Trello** na tarefa **Módulo 1 - Projeto Avaliativo 2**, presente na semana 12 do AVA até o dia **29/01/2023 05/02/2023 às 23h55**.

O repositório deverá ser privado, com as seguintes pessoas adicionadas:

- Kelvis Antonio Rodrigues Borges - **kelvisdev**
- Operação DEVinHouse - **devinhouse-operacao**

Não serão aceitos projetos submetidos após a data limite da atividade, e, ou alterados depois de entregues.

Importante:

1. Não serão aceitos projetos submetidos **após a data limite da atividade**, e, ou **alterados** depois de entregues.
2. Será considerado como data final de entrega a **última atualização** no repositório do projeto no GitHub. Lembre-se de não modificar o código até receber sua nota.
3. Não esqueça de **submeter o link no AVA**. Não serão aceitos projetos em que os links não tenham sido submetidos.

6 PLANO DE PROJETO

Ao construir a aplicação proposta, o aluno estará colocando em prática os aprendizados em:

- **Kanban Board:** O que é Kanban e utilização no Trello.
- **Versionamento:** O que é versionamento de código, aplicações e utilização no Github
- **HTML e CSS:** HTML5 (elementos semânticos) e CSS (seletores, principais estilos, layouts e flexbox).
- **JavaScript:** Variáveis, Tipos de Dados, Estrutura de Controle de Fluxo (condicional e repetição), Operadores (aritméticos, lógicos e relacionais), Funções, ES6+, Escopo, Arrow Functions, Operadores Rest e Spread, Funções de Arrays, Manipulação do DOM, Utilização de Seletores, Eventos, Classes, Atributos, Encapsulamento, Herança, Polimorfismo, Módulos, Funções Assíncronas, JSON, LocalStorage, Interval e Timeout
- **Angular:** Setup Inicial (instalação e configuração), Angular CLI, Estrutura, Bootstrap, Componentes, Property Binding, Passagem de Valores, Injeção de Dependência, Eventos de Componentes, Rotas, Navegação Segura, Decorators, Diretivas (ngIf e ngFor) e ng-content.