

PROVA FINAL

- Entrega Sem prazo
- Pontos 40
- Perguntas 10
- Limite de tempo 60 Minutos
- Tentativas permitidas 2

Instruções

INSTRUÇÕES DA PROVA FINAL

- A prova tem a duração de 60 minutos.
- Ao clicar em PROVA FINAL, no menu “Testes” você iniciará a prova. Ao acessar a página com as questões, o tempo começa a ser contado.
- A prova é composta de 10 (dez) questões objetivas, sendo 04 (quatro) pontos cada.
- Ao final do teste não se esqueça de enviá-lo, clicando no botão “ENVIAR TESTE”. Só utilize esse botão quando tiver finalizado a avaliação.
- Se necessário, durante a prova, entre em contato pelo link “Atendimento”.
- Você terá 02 (duas) tentativas e, caso necessite de uma nova, será preciso solicitar a prova extra, que requer pagamento de taxa adicional.

Atenção: Todas as provas iniciadas e que não houverem sido submetidas serão automaticamente encerradas pelo sistema transcorridos os 60 minutos de duração.

Boa prova!

Fazer o teste novamente

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	Tentativa 1	9 minutos	40 de 40

Pontuação desta tentativa: 40 de 40

Enviado 23 abr em 12:34

Esta tentativa levou 9 minutos.



Pergunta 1

4 / 4 pts

Sabemos que o JavaScript é uma linguagem interpretada pelo browser, sendo a principal responsável por adicionar interatividade nas aplicações web.

Sobre a linguagem JavaScript e suas características, assinale a alternativa **incorreta**:



O JavaScript é uma linguagem fracamente tipada, sendo possível declarar uma variável como string e associar um valor inteiro posteriormente.



O JavaScript é uma linguagem interpretada pelo Browser Engine, sendo possível também executá-la no lado do servidor através da plataforma Node.js.

Correto!



O JavaScript possui um mecanismo interno de checagem de tipos eficiente, tornando visível possíveis bugs de tipagem logo em tempo de desenvolvimento e build.



A linguagem pode, sem restrições, ser utilizada para desenvolvimento de aplicações complexas e de grande porte.

O JavaScript, por ser uma linguagem fracamente tipada, não possui nativamente um bom mecanismo de checagem de tipos, o que torna impossível, de forma nativa, essa possibilidade.



Pergunta 2

4 / 4 pts

Cada browser (navegador) possui a sua própria engine de interpretação de códigos no front-end, o que pode fazer com que browsers diferentes apresentem resultados visuais e de interação diferentes para o usuário final.

Sobre os Browser Engines, é **correto** afirmar:



São softwares separados do Navegador, que podem ser instalados como extensão do browser afim de facilitar o desenvolvimento.



São responsáveis também por compilar os códigos de front-end no browser.



São responsáveis por armazenar todo o sistema de rotas de uma aplicação, similar ao que acontece no web.config do IIS.

Correto!



São responsáveis por interpretar e renderizar os códigos das linguagens HTML, CSS e JavaScript no cliente (navegador).

A engine de interpretação é responsável por interpretar os códigos escritos no browser, o que torna possível a escrita de aplicações front-end.



Pergunta 3

4 / 4 pts

Sobre os frameworks Angular, React e VueJs, é **correto** afirmar:



São frameworks de front-end que possuem arquiteturas idênticas.



São frameworks de desenvolvimento web que focam majoritariamente na criação de aplicações para grandes corporações.

Correto!

- ☒ São frameworks de front-end que resolvem os mesmos problemas de formas diferentes.



São frameworks de server-side rendering, cujo principal objetivo é a renderização de aplicações com foco em SEO.

Os frameworks citados possuem arquiteturas similares, mas com algumas diferenças entre eles. Todos conseguem resolver os mesmos problemas utilizando abordagens e linguagens diferentes.



Pergunta 4

4 / 4 pts

Ao estilizar aplicações Web com o CSS, o BEM (Block-Element-Modifier) define um padrão de nomenclatura para classes do CSS.

Com base nas regras do padrão BEM, marque a alternativa que descreve corretamente as partes do nome da classe CSS: **.site-search__input-field--large**.

- ☐ A classe refere-se ao bloco input-field.
- ☐ A classe refere-se ao modifier input-field.
- ☐ A classe refere-se ao elemento site-search.

Correto!

- ☒ A classe refere-se ao bloco site-search.

A nomenclatura BEM é construída da seguinte forma: block—element__modifier.



Pergunta 5

4 / 4 pts

A arquitetura flux é proposta que visa gerir estados da aplicação de forma eficiente e robusta. Alguns frameworks como Redux e NgRx implementam essa arquitetura.

Sobre a arquitetura flux, é **correto** afirmar:

- ☐ A arquitetura flux é utilizada majoritariamente para gerenciamento de estados de CSS.

Correto!

- ☒ A aplicação deve conter uma única Store que concentra os estados da aplicação.



A aplicação deve conter múltiplas Stores, ao menos uma para cada módulo para que torne mais fácil a manutenção de cada módulo.

- ☐ As actions descrevem como as alterações acontecem, contendo a lógica necessária para atualizar a store.

A arquitetura flux é uma implementação para construção de interfaces de usuário. Tem o objetivo de manter o estado da aplicação como um todo (token de sessão de usuário, dark mode etc.), e por isso contém apenas uma única Store.



Pergunta 6

4 / 4 pts

As aplicações construídas utilizando a abordagem de server-side rendering têm características específicas que são inerentes ao desenvolvimento web, mesmo possuindo parte da sua

renderização sendo feita no lado do servidor.

Sobre as aplicações server-side rendered, qual alternativa melhor define um dos seus principais benefícios?

Correto!



Melhoram a aplicação no quesito SEO, uma vez que as ferramentas de busca conseguem interpretar o conteúdo que chega diretamente do servidor e interpretar o HTML semântico da aplicação.



Oneram a aplicação no quesito SEO, uma vez que as ferramentas de busca não conseguem ler as informações do servidor.



Deixam a aplicação mais lenta pois tudo o que é renderizado no servidor demora mais tempo do que se fosse renderizado no browser.



Todas as alternativas estão corretas.

As aplicações já possuem o HTML semântico ao fazer uma requisição GET na URL



Pergunta 7

4 / 4 pts

Dentre as formas de implementação de micro front-ends, temos a alternativa de implementar em tempo de build por meio de pacotes npm.

Sobre essa forma de implementação, é **correto** afirmar que:



Todas as alternativas estão corretas.



A base de código é concentrada em um único projeto, similar ao que acontece na arquitetura baseada em monorepos.



Cada micro front-end perde a sua coesão uma vez que deve concentrar a responsabilidade do gerenciamento de versão do projeto.

Correto!



A pipeline de build, test e deploy não é totalmente independente, uma vez que para lançar uma versão do micro front-end é necessário também atualizar o projeto principal.

O controle de versão nesse caso é de cada um dos pacotes, mas como ele está listado como dependência do projeto principal, para atualizar um micro front-end é necessário também atualizar a dependência no projeto principal.



Pergunta 8

4 / 4 pts

A implementação de micro front-ends por meio de iframes é possível utilizando algumas APIs do browser, uma vez que se torna necessário em alguns casos a comunicação entre micro front-ends.

Sobre essa forma de implementação, assinale a alternativa **correta**:



Utiliza um assets-manifest.json para identificar os micro front-ends, similar ao Service Discovery na abordagem de microsserviços.



Torna impossível a convivência de mais de um framework na mesma aplicação.



É necessário utilizar um framework como o Redux para estabelecer uma comunicação entre os micro front-ends.

Correto!



Propõe a criação de uma camada de event-receivers em uma aplicação principal, que são emitidos/recebidos via PostMessage API.

Na implementação via iframes, só é possível estabelecer uma comunicação entre aplicações através da API de PostMessage do próprio browser.



Pergunta 9

4 / 4 pts

As aplicações PWA possuem uma gama de APIs para facilitar o controle e desenvolvimento de aplicações web.

Sobre o Service Worker API, é **correto** afirmar que:



Trata as requisições HTTP ao servidor.



Armazena os arquivos estáticos da aplicação.



Relacionado a estrutura de dados e armazenamento no browser.

Correto!



Atua como proxy entre a aplicação e o servidor.

A principal função do service worker é interceptar as requisições da aplicação para o servidor, possibilitando respostas personalizadas com recursos off-line, por exemplo.



Pergunta 10

4 / 4 pts

A computação serverless, como qualquer coisa na computação, não é uma bala de prata que resolve todos os problemas das organizações.

Sobre os casos de uso de computação serverless, podemos afirmar que **não** é comumente utilizada em qual dos cenários abaixo?



Back-end de IoT.

Correto!



Desenvolvimento de aplicações móveis nativas.



Aplicações serverless-side rendering.



Processamento de dados em tempo real.

Para desenvolvimento de aplicações móveis nativas temos outras ferramentas, como o React Native, uma vez que é demandado um recurso de build e conversão de código.

Pontuação do teste: 40 de 40