Observações Importantes:

Observações Importantes:

**Durante todo o processo, é importante realizar reuniões de acompanhamento com a equipe para discutir progresso, bloqueios e ajustes necessários.**

**Realizar testes com dados reais em ambiente controlado é uma prática essencial para garantir que um sistema ou produto se comportará como esperado em um cenário real, sem afetar os usuários finais ou a operação de produção. Seguindo um planejamento cuidadoso e realizando testes abrangentes, você pode minimizar riscos, identificar problemas antes do lançamento e otimizar o desempenho do sistema para um lançamento bem-sucedido. (Será feito durante todo o processo do desenvolvimento do sistema)**

***CRONOGRAMA SISTEMA TEA (EDITORA FOCCUS)***

Mês 1: Planejamento e Levantamento de Requisitos

Semana 1-2: (22.10.2024 - 05-11.2024)

- Reunião inicial com com os participantes direto da empresa (Foccus) para definir - - - objetivos do projeto.

- Levantamento e análise de requisitos: funcionalidade, performance, segurança, etc.

- Definição do escopo do projeto.

**Processo de Levantamento e Análise de Requisitos:**

1. **Identificação das Partes Interessadas**: Definir quem são os stakeholders (usuários finais, gerentes, desenvolvedores, etc.).
2. **Coleta de Requisitos**: Usar técnicas como entrevistas, questionários, análise de documentos, etc.
3. **Documentação de Requisitos**: Registrar e organizar os requisitos de maneira clara e compreensível.
4. **Análise de Requisitos**: Avaliar a viabilidade dos requisitos, resolver conflitos entre requisitos e priorizar os mais críticos.
5. **Validação de Requisitos**: Garantir que os requisitos atendem às necessidades dos stakeholders e que são completos e consistente

Semana 3-4: (06.11.2024 - 20.11-2024)

- Definição do layout das planilhas aluno e professor

- Recebimento da planilhas excel ALUNOS ,PROFESSOR para iniciarmos o teste de migração para o banco de dados do sistema

- Estudo de viabilidade técnica.

O **Estudo de Viabilidade Técnica** é uma etapa crucial em qualquer projeto de desenvolvimento de sistemas, software ou infraestruturas tecnológicas. Ele avalia se a solução proposta é tecnicamente viável, levando em consideração os recursos, a tecnologia, as habilidades da equipe e os custos envolvidos. O estudo de viabilidade técnica ajuda a reduzir os riscos e a garantir que a equipe tenha a capacidade necessária para entregar o projeto de forma eficaz, dentro dos requisitos e prazos.

- Escolha da arquitetura e estrutura do sistema (MVC). (regras de pastas dentro de uma IDE)

- Planejamento de infraestrutura (servidores, banco de dados, etc).

- Definição de tecnologias (PHP, Frameworks, Banco de Dados,)

**PHP**: Linguagem de programação para desenvolvimento web.

**Frameworks PHP**: Conjuntos de bibliotecas que facilitam o desenvolvimento de aplicações web. Ex: Laravel, Symfony, CodeIgniter.

**Banco de Dados**: Ferramenta para armazenar dados, como MySQL (relacional)

Mês 2: Design e Arquitetura

Semana 1-2: (21.11.2024 a 05.12.2024)

- Inicio de migração das planilhas aluno, professor para testes .

- Inserção dos dados de migração

Fora de um cronograma normal de um projeto , solicitação dos envolvidos para uma solução futura de ajuda nas inserções de alunos e escolas

- Teste da migração de alunos e escolas junto com os participantes do projeto

- Definição da arquitetura do sistema (estrutura de pastas, camadas de aplicação, etc).

- Escolha de frameworks (Laravel ).

**A definição da arquitetura do sistema e a escolha de frameworks como o Laravel são passos críticos no desenvolvimento de um sistema robusto e escalável. A arquitetura em camadas, junto com a estrutura de pastas organizada, ajuda a manter o código limpo e modular. Por outro lado, o Laravel oferece um conjunto de ferramentas poderosas para o desenvolvimento rápido, mas é essencial que a equipe tenha as habilidades certas para aproveitar seu potencial e garantir uma aplicação de alta qualidade.**

**Se o sistema exigir uma solução rápida e escalável, Laravel, com sua abordagem simples e extensa gama de recursos, é uma excelente escolha, especialmente quando combinado com uma arquitetura sólida que pode crescer conforme o projeto avança.**

Semana 3-4: (06.12.2024 a 20..12.2024)

**-** Design das telas (protótipos ) 🡪 - Design das telas (protótipos ) (TELAS PRINCIPAIS). Protótipos de telas são representações visuais de como um produto digital vai aparecer para os usuários. Eles são usados para ilustrar o layout e a interação da interface, permitindo que a equipe de desenvolvimento, design e stakeholders possam visualizar o produto antes de ser construído.

- Reunião com a equipe Foccus para apresentações de definições de rotinas do sistema ou seja sequência de cadastros

- Testes contínuos são fundamentais,

- Continuação do esboço do banco de dados e definição do esquema. 🡪A definição do esquema de banco de dados é um processo contínuo, e deve ser feita com cuidado para garantir a integridade dos dados, a eficiência das consultas e a escalabilidade do sistema. O esboço do banco de dados e sua implementação devem refletir os requisitos do projeto e serem ajustados conforme novas necessidades surgem durante o desenvolvimento.

Além disso, a **documentação do esquema do banco de dados** (tabelas, campos, índices, relações) é essencial para garantir que toda a equipe de desenvolvimento tenha uma compreensão clara de como os dados estão organizados.

Mês 3: Configuração do Ambiente e Início do Desenvolvimento

Semana 1-2: :(21-12-2024 a 04-01-2025)

- Preparação do ambiente de desenvolvimento (servidores, IDEs, ferramentas)🡪 A **preparação do ambiente de desenvolvimento** é uma das etapas cruciais para garantir que toda a equipe de desenvolvimento tenha as ferramentas necessárias para trabalhar de forma eficiente, colaborativa e organizada. Essa preparação envolve a configuração de servidores, IDEs (Ambientes de Desenvolvimento Integrado), ferramentas de versionamento, banco de dados, entre outras, dependendo das necessidades do projeto.

- Configuração do controle de versão (Git/GitHub/GitLab). Para os envolvidos no desenvolvimento do sistema (Software e hardware )

- Criação de templates iniciais de código (estrutura básica do projeto).

Essa estrutura básica e os templates iniciais de código fornecem um ponto de partida para um projeto Laravel. Eles configuram a estrutura de diretórios, modelos e controladores essenciais, além de garantir que você tenha uma base bem organizada para começar a adicionar funcionalidades. Com o tempo, você pode expandir e modificar a estrutura à medida que o projeto cresce, mas ter um bom começo ajuda a manter a organização e a manutenibilidade do código.

Esses templates são apenas o ponto de partida para o seu projeto. O objetivo é configurar uma estrutura limpa, escalável e bem organizada para que todos os desenvolvedores trabalhem de forma colaborativa, seguindo boas práticas de desenvolvimento.

Semana 3-4 (05 -01-2025 a 19.01.2025)

-- Continuação da Criação de templates iniciais de código (estrutura básica do projeto).

- Desenvolvimento da estrutura inicial do banco de dados.

- Criação das funcionalidades básicas (login, autenticação, cadastro, etc).

- Implementação de testes iniciais (unitários, integração).

**Validação de Dados de Entrada (dados inseridos nos fórmularios)**

**Segurança**: Além de criptografar a senha, proteja suas rotas com autenticação

**Banco de Dados**: Em ambientes de desenvolvimento, use bancos de dados de teste e garanta que eles sejam limpos após os testes.

Essa estrutura básica de funcionalidades de login, autenticação e cadastro com testes ajudará você a começar com uma base sólida.

Mês 4: Desenvolvimento de Funcionalidades Principais

Semana 1-2: : (20-01-2025 a 03.02.2025)

- Implementação das principais funcionalidades do sistema (ex: CRUDs, funcionalidades chave do projeto).

- Desenvolvimento da lógica de negócios.

A **lógica de negócios** é crucial para garantir que as regras e comportamentos essenciais do sistema sejam aplicados corretamente. Ela pode envolver validação de dados, cálculos de preços, controle de estoque, entre outras funcionalidades. No exemplo acima, implementamos diversas lógicas que podem ser expandidas e integradas para atender às necessidades do seu sistema.

A lógica de negócios deve ser bem estruturada e desacoplada das outras partes do sistema (como a camada de apresentação e a camada de persistência de dados), para facilitar a manutenção, testes e escalabilidade.

Semana 1-2: (04-02-2025 a 18.02.2025)

Criação de telas principais (front-end e back-end).

Testes de funcionalidade.

Os testes são cruciais para garantir que cada parte do sistema funcione corretamente. Existem diversos tipos de testes que você pode realizar:

**Testes de unidade**  Testam funções isoladas.

**Testes de integração**: Testam a interação entre diferentes partes do sistema (por exemplo, a comunicação entre o front-end e back-end).

**Testes end-to-end**: Testam o sistema como um todo, incluindo o comportamento do usuário

Reuniões semanais para acompanhamento de progresso.

Mês 5: Desenvolvimento de Funcionalidades Secundárias

Semana 1-2 :(19-02-2025 a 05.03.2025)

- Desenvolvimento de funcionalidades secundárias ou adicionais.

- Implementação de recursos de segurança (autenticação avançada, controle de acesso, etc).

Além de autenticação e controle de acesso, é fundamental adotar boas práticas de segurança para proteger sua aplicação.

**Validação e Saneamento de Entradas**: Proteja seu sistema contra **SQL Injection**, **XSS (Cross-site Scripting)**, e **CSRF** sanitizando todas as entradas de usuário.

**Uso de HTTPS**: Sempre utilize HTTPS para garantir a criptografia das requisições e respostas.

**Armazenamento Seguro de Senhas**: Sempre use **bcrypt** ou **argon2** para criptografar senhas de usuários.

**Logout e Expiração de Sessão**: Defina um tempo de expiração de token adequado (como uma hora) e possibilite o logout para invalidar tokens.

Semana 3-4: (06-03-2025 a 20.03.2025)

- Testes de integração entre diferentes partes do sistema.

- Ajustes baseados no feedback dos testes iniciais.

- Primeira demonstração do produto para stakeholders.

A primeira demonstração de um produto para stakeholders é um momento importante, pois oferece a oportunidade de mostrar o valor do produto, destacar suas funcionalidades e obter feedback crucial para o desenvolvimento e aprimoramento contínuo. Na verdade vamos estar realizando a partir do mês 2,3,4,5,6

Mês 6: Teste ,Ajustes e correções

Semana 1-2: : (21-03-2025 a 04.04.2025)

- Realização de testes funcionais e de carga.

Essas atividades são essenciais para garantir que o software seja funcional, rápido e fácil de usar. Se você precisar de mais detalhes sobre algum desses pontos ou de ajuda com técnicas específicas, posso explicar mais a fundo.

- Identificação de gargalos de performance.

Melhorias contínuas em usabilidade e correções de bugs são essenciais para manter o sistema eficiente, agradável de usar e livre de falhas. Aplique boas práticas como validações robustas, mensagens de erro claras, e uma navegação intuitiva. Para garantir que as mudanças não afetem negativamente outras partes do sistema, mantenha uma boa cobertura de testes automatizados.

- Correções de bugs e melhorias de usabilidade.

São partes cruciais para manter um sistema estável, seguro e fácil de usar. Ao corrigir bugs, sempre se concentre em garantir que a solução não introduza novos problemas e adote boas práticas de monitoramento e testes. Quanto à usabilidade, faça mudanças iterativas, buscando sempre simplificar a experiência do usuário, oferecendo feedback claro e criando um fluxo lógico e intuitivo.

Semana 3-4: (05-04-2025 a 19.04.2025)

- Testes de segurança (injeção SQL, XSS, CSRF).

- Ajustes finais nas funcionalidades.

- Segunda demonstração para stakeholders.

A segunda demonstração para stakeholders é uma etapa crucial no ciclo de desenvolvimento, pois permite validar se a aplicação está no caminho certo e atender melhor às necessidades do cliente ou usuário final. A chave é demonstrar de forma clara as funcionalidades, obter feedback construtivo e planejar as próximas ações para garantir que o projeto atenda às expectativas e seja entregue com qualidade.

Mês 7: Refatoração e Melhoria Contínua

Semana 1-2: (20-04-2025 a 04.05.2025)

- Refatoração do código para melhorar performance e legibilidade.

A **refatoração do código** é um processo contínuo durante o desenvolvimento de software, que visa melhorar o desempenho, a legibilidade, a manutenção e a escalabilidade do código sem alterar sua funcionalidade. A refatoração deve ser feita com cuidado, especialmente quando o sistema já está em uma fase avançada de desenvolvimento, para não introduzir novos erros.

### 

- Reavaliação da arquitetura, se necessário.

A reavaliação da arquitetura de um sistema deve ser realizada quando houver sinais claros de que a arquitetura atual não atende mais aos requisitos de escalabilidade, manutenção ou performance. A abordagem deve ser cuidadosa, levando em consideração os custos e benefícios de migrações ou mudanças, e deve ser realizada de forma gradual para minimizar o impacto. Identificar e corrigir gargalos, adotar novas tecnologias (como microsserviços, containers e GraphQL), e melhorar a infraestrutura são passos importantes para garantir que o sistema permaneça robusto e eficiente à medida que cresce.

Semana 3-4: (05-05-2025 a 19.05.2025)

- Implementação de logging e monitoramento.

A implementação de logging e monitoramento é crucial para garantir que o sistema seja observável, o que significa que você pode diagnosticar problemas rapidamente, entender como ele se comporta em tempo real e identificar áreas de melhoria.

- Ajustes nos testes de unidade e integração. Se necessário

- Planejamento da preparação para a fase de pré-produção.

A fase de pré-produção é crucial porque ela prepara o terreno para a execução do projeto. Quanto mais bem planejada e organizada for essa fase, menores são as chances de problemas surgirem durante a produção. Invista tempo na definição dos detalhes e na comunicação entre todos os envolvidos, para que a produção aconteça da forma mais fluida possível.

Mês 8: Preparação para Pré-Produção

Semana 1-2: (20-05-2025 a 03.06.2025)

- Realização de testes finais em ambiente de pré-produção.

- Testes com dados reais (em ambiente controlado).

**Realizar testes com dados reais em ambiente controlado é uma prática essencial para garantir que um sistema ou produto se comportará como esperado em um cenário real, sem afetar os usuários finais ou a operação de produção. Seguindo um planejamento cuidadoso e realizando testes abrangentes, você pode minimizar riscos, identificar problemas antes do lançamento e otimizar o desempenho do sistema para um lançamento bem-sucedido. (Será feito durante todo o processo do desenvolvimento do sistema)**

- Ajustes finais no desempenho.

Semana 3-4: (04-06-2025 a 18.06.2025)

- Implementação de estratégias de backup e recuperação de desastres.

- Reunião de revisão final do projeto.

- Documentação do código e do sistema.

Mês 9: Pré-Produção e Ajustes Finais

Semana 1-2: (19-06-2025 a 03.07.2025)

- Deploy no ambiente de pré-produção.

- Testes finais em produção (testes de carga, testes de performance em ambiente real).

- Correção de eventuais falhas encontradas.

**Os testes finais em produção são essenciais para garantir que o sistema atenda às expectativas de desempenho e capacidade sob condições reais de uso. A combinação de testes de carga e testes de performance permite identificar problemas de escalabilidade, capacidade de resposta e falhas críticas antes de um lançamento em grande escala. A correção de falhas e a aplicação de ajustes finais são passos importantes para assegurar que o sistema esteja estável, rápido e pronto para lidar com o tráfego real de usuários.**

Semana 3-4: (04-07-2025 a 18.07.2025)

- Treinamento para os usuários finais.

- Documentação do usuário (se aplicável).

- Testes de aceitação com stakeholders.

Treinamento para os usuários finais, documentação do usuário e testes de aceitação com stakeholders são componentes essenciais para garantir que o sistema seja adotado com sucesso, que os usuários saibam utilizá-lo com confiança e que os stakeholders estejam satisfeitos com o produto final.

* O treinamento prepara os usuários para utilizar o sistema de maneira eficiente e sem erros.
* A documentação fornece apoio contínuo, permitindo que os usuários resolvam dúvidas e problemas de forma independente.
* Os testes de aceitação garantem que o sistema atenda às necessidades de negócio e que os stakeholders estejam satisfeitos com o resultado antes do lançamento final.

Mês 10: Lançamento e Pós-Lançamento

Semana 1-2: (19-07-2025 a 02.08.2025)

- Deploy final para produção.

- Monitoramento de logs e métricas pós-lançamento.

- Correção de bugs críticos após o lançamento.

Semana 3-4: : (03-08-2025 a 17.08.2025)

- Análise de performance e uso.

- Entrega da documentação completa (técnica e do usuário).

- Reunião de fechamento do projeto com stakeholders.

- Planejamento de manutenção e atualizações futuras.

### Conclusão

A finalização de um projeto de desenvolvimento de sistema ou software não é apenas sobre entregar o produto, mas também garantir que ele continue funcionando de forma eficiente e alinhado com as necessidades do negócio no futuro. A **análise de performance e uso** permite avaliar o comportamento do sistema em ambiente real, enquanto a **documentação completa** oferece a base necessária para a operação e manutenção do sistema. A **reunião de fechamento** com stakeholders assegura que todos os envolvidos estão alinhados, e o **planejamento de manutenção e atualizações futuras** prepara o sistema para evoluir de acordo com as novas demandas e tecnologias. Essas etapas formam um ciclo contínuo de melhoria e suporte que é crucial para o sucesso e longevidade do sistema.