

Projeto Integrador

REGRAS DO JOGO

DigitalHouse >
Coding School

FS Node

O que é o Projeto Integrador



O Projeto Integrador (PI) é uma aplicação prática dos conceitos teóricos vistos nas aulas ao longo do seu curso.

Esse projeto é uma estratégia didática que consiste na realização de um conjunto de atividades articuladas entre si, com início, desenvolvimento e fim, e que tem por objetivo desenvolver um produto final.

Essa metodologia favorece a aprendizagem através da articulação de conhecimentos teóricos e práticos, além de aproximar o aluno da realidade do mercado profissional.

Benefícios do Projeto Integrador

O **Projeto Integrador** é uma experiência rica de aprendizagem, uma vez que se propõe a ter certas condições e considerações que o tornam ideal para o aprendizado.

Veja, a seguir, quais são as **vantagens**:

1	Está enquadrado na aprendizagem e no conteúdo de cada módulo.
2	Permite aplicar as ferramentas aprendidas nas diferentes aulas do curso.
3	Constitui o desafio de levar as ideias à prática.
4	Convida a colocar em prática as habilidades sociais que queremos que nossos(as) alunos(as) desenvolvam.

Como funciona (dentro/fora do horário de aula)

O Projeto Integrador, também chamado de PI, é parte fundamental da metodologia da aprendizagem da Digital House, e está presente em todos os cursos.



No caso do curso de **Programação Full Stack**, ele é destinado a criação de uma aplicação web completa, e seu desenvolvimento ocorre em paralelo aos assuntos abordados em aula.

O PI é dividido em **6 sprints**, cada uma com um conjunto de entregas específicas, para as quais, o grupo receberá o feedback e orientação para o aprimoramento do que for necessário.

Sprints do PI de Full Stack

Número	Tema	Entrega
1	Wireframe e padrões de layout	Deverá ser entregue um protótipo de baixa qualidade, com ao menos, as principais telas do sistema.
2	Front-end I	Implementação das telas do protótipo e demais, em HTML e CSS de forma estática. Com dados mockados e navegação simples.
3	Front-end II	Criação de um projeto node, convertendo as páginas criadas anteriormente para EJS, e aplicar o sistema de rotas do Express.
4	Bancos de Dados	Modelagem e criação do banco de dados do sistema a ser desenvolvido.
5	Back-end	Implementação das features planejadas de acordo com as telas, criando a ligação do UI com o backend da aplicação e salvando informações no banco de dados.
6	Api	Deverá ser implementada um API externa, condizente com o tema do projeto. E criar validações de formulários usando JS.

Orientação dos professores

Os professores devem orientar cada grupo, separadamente, sobre:

1

Validade da
proposta do
projeto

2

Definição das
tarefas de cada
sprint

3

Uso adequado
das tecnologias

4

Planejamento e
organização das
tarefas de cada
sprint

Acompanhamentos

Além disso, os grupos são acompanhados frequentemente:

- ➡ em reuniões síncronas quinzenais de colearning
- ➡ em aulas síncronas específicas do PI
- ➡ constantemente de forma assíncrona no canal do slack do grupo
- ➡ pelos commits realizados no Github

Ferramentas utilizadas

Para o acompanhamento do PI são utilizadas 3 ferramentas básicas, que alunos e professores devem ter acesso:

Trello	Slack	Github
		
Gerenciamento das tarefas das sprints	Canal oficial de comunicação assíncrona	Gerenciamento dos códigos do projeto

Exemplos

- Veja alguns projetos desenvolvidos por alunos:

Moovebike


Kombucha

Wined +

TraDinDin

Scranner

Band



Onde queremos chegar?

Apresentação final

Ocorre no último dia de aula, após todos os ajustes apontados durante os acompanhamentos do desenvolvimento. O grupo deve organizar a apresentação levando em conta:



A justificativa da
ideia



As funcionalidades
desenvolvidas



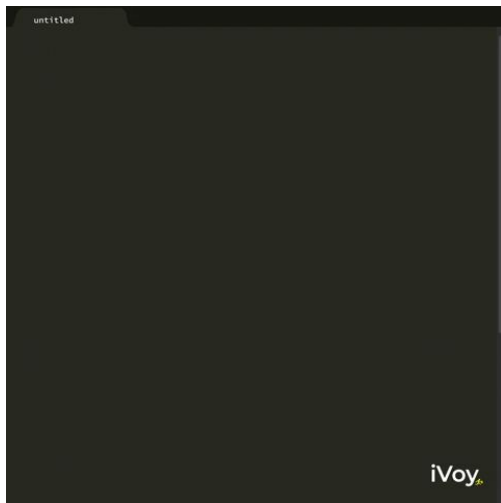
As tecnologias
empregadas

Além disso, deve apresentar a versão final funcionando, e um breve relato das dificuldades encontradas e como foram solucionadas.

Qual a entrega esperada?

Ao final, o grupo deve apresentar o sistema web proposto funcionando, dentro de um repositório Git.

O sistema deve abordar as tecnologias apresentadas no curso, sendo elas:



- **Node**
- **HTML**
- **CSS**
- **Javascript**
- **MySQL**

Qual a entrega esperada?

O sistema deve abordar manipulações de dados por meio de interfaces (UI),

1

Autenticação de
usuário

2

Páginas dinâmicas
(Informações do
banco de dados)

3

Além de features
de acordo com o
tema do projeto

Rubrica

Como vamos avaliar o Projeto? O projeto deve cumprir os seguintes requisitos:

Área restrita com login

Banco de dados configurado, para guardar as informações

Back-end próprio

Fazer as 4 operações básicas do banco de dados (CRUD): ler, criar, atualizar e apagar dados

Responsividade (se adaptar a diferentes tamanhos de tela)

Commits de todos os integrantes durante toda a jornada do curso

Seu grupo de trabalho

Por que recomendamos trabalhar em grupo?



O **Projeto Integrador** é realizado em pequenos grupos, estimulando a **comunicação, a inovação e criatividade** a partir do conhecimento individual e coletivo, permitindo um trabalho colaborativo entre os alunos. Além disso, coloca em prática habilidades sociais favorecendo o desenvolvimento de **soft skills**.

Todo o processo de elaboração do Projeto Integrador (PI) pelo grupo trará desafios que são, na verdade, oportunidades para o desenvolvimento de habilidades que são tão necessárias para o dia a dia de um profissional de mercado (e para a vida também!),

Como dividimos os grupos?

Os grupos são formados na sprint 1, e devem conter entre 4 e 6 alunos.

A separação do grupo pode ocorrer por:



Organização dos
próprios alunos



Identificação
com os projetos
propostos



Organização dos
professores
após análise do
perfil dos alunos

Dedicação ao PI

Para cumprir com todas as entregas propostas, e a apresentação final, o grupo deve se manter engajado, utilizando as ferramentas oficiais do projeto.

Além disso, cada aluno deve se dedicar ao desenvolvimento das tarefas que ficaram a seu encargo contribuindo para o sucesso do grupo.

OBS: apesar do curso oferecer algumas aulas dedicadas ao PI, a maior parte de seu desenvolvimento ocorre fora dos horários de aula ao vivo.

Dedicação ao PI

Dentro dessa dedicação, espera-se que todos os alunos do grupo:

- ➡ participem da comunicação no canal do grupo no slack
- ➡ participem dos checkpoints quinzenais de acompanhamento com os professores
- ➡ participem das reuniões organizadas pelo próprio grupo
- ➡ mantenham suas tarefas organizadas no Trello
- ➡ realizem os commits regularmente de suas etapas no GitHub

Mensagem da Coordenação

Mensagem da Coordenação



WE CAN
THIS

Olá futuro colega de trabalho!

A importância do PI está no **trabalho em equipe**, na **colaboração** e na **tolerância**.

Este é o momento de colocar em prática tudo o que está sendo visto no curso, aproveite esta oportunidade para **fixar o conhecimento**, para **aprender novas formas de fazer**, para **errar** e para **solucionar problemas** que sempre aparecem!!!

Nós da coordenação estaremos sempre perto de você para o que for necessário, para que esta experiência seja a melhor possível.

Tarefas das Sprints

Sprint 1 - Wireframe e padrões de layout

Considerando a proposta do projeto seu grupo deverá:

1. Identificar cada página que será necessária
2. Desenvolver o wireframe de cada página identificando:
 - a. as principais funcionalidades
 - b. elementos de interação por parte do usuário
 - c. elementos de navegação
 - d. links
3. Desenvolver a Identidade Visual do projeto (logo, paleta de cores e tipografia)

Sprint 2 - Front-end I

A partir dos wireframes da sprint anterior, seu grupo deverá:

1. Criar o projeto no Git
2. Desenvolver o HTML e CSS de cada uma das páginas projetadas, respeitando a identidade visual proposta
3. Inserir os dados mockados nas páginas desenvolvidas
4. Realizar a navegação entre as páginas

Sprint 3 - Front-end II

Agora que as telas estão desenvolvidas, sua equipe deve:

1. Transportar o HTML para o EJS
2. Realizar a navegação entre páginas usando o sistema de rotas do Express
3. Fazer com que dados enviados via formulário sejam recebidos no servidor
4. Ter sistema de uploads configurado e funcional
5. Regras do sistema configuradas
6. Middlewares (se necessário) configurados e implementados

Sprint 4 - Bancos de Dados

Baseado nas telas e no projeto desenvolvido até aqui, ao final da sprint vocês devem possuir:

1. Modelo D.E.R (Diagrama Entidade-Relacionamento)
2. Modelagem do banco de dados
 - a. tabelas
 - b. campos das tabelas
3. Relacionamento entre tabelas
 - a. categoria de relacionamento
 - b. campos que terão relacionamento

Sprint 5 - Back-end

Agora que já temos os modelos prontos, está na hora de colocar no projeto e integrar com as regras que já temos:

1. Criação e configuração de relações dos modelos e migrações no Sequelize
2. Transportar o que foi modelado na etapa anterior
3. Configurar os controllers para se comunicar com o banco de dados

Sprint 6 - Api

Por fim e, com certeza, não menos importante, nossos sistemas precisam conseguir conversar com outros. É isso que esperamos nessa parte:

1. Front-end ou back-end conversando com alguma API (própria ou de terceiros)

Também, esperamos nesse momento do curso que o projeto esteja nos ajustes finais ou próximo disso, para manter o cronograma de entrega final sem atrasos.