



MATERIAL DO PROFESSOR

Lógica de programação: exploração
criativa - Parte 1



1

Introdução

O Material do Professor é um instrumento que oferece o suporte e o embasamento necessários para a condução dos estudantes nesta jornada de aprendizagem.



Aqui, você encontrará orientações pedagógicas para o trabalho com cada unidade, bem como os objetivos de aprendizagem, os conteúdos e habilidades, as estratégias didáticas utilizadas na construção da unidade e, por fim, materiais aula a aula que servirão de apoio ao seu planejamento e durante o processo avaliativo dos estudantes. Utilize o Material do Professor como guia durante o planejamento das aulas, adaptando as estratégias de acordo com as necessidades e interesses dos seus estudantes.

Este material é digital e pode sofrer atualizações periódicas. Certifique-se de verificar regularmente a plataforma para baixar as versões mais recentes e garantir que você tenha acesso às informações mais atualizadas e relevantes para suas aulas.

2

Conteúdo programático

Olá, professor(a)!

Nessa unidade, teremos dois projetos, sendo o primeiro chamado *Meu mundo* e o segundo, *Cartão animado*. Ambos são projetos introdutórios para mostrar como podemos criar algo simples e criativo com programação. Iniciaremos criando personagens e cenários únicos a partir de ferramentas e outros elementos disponíveis no ambiente de programação. Já no segundo projeto, criaremos um cartão para referenciar algo que gostamos. Além disso, animaremos as letras do nosso cartaz com efeitos visuais super divertidos. Ambos os projetos tem como objetivo fazer com que cada aluno consiga se expressar da sua maneira por meio de elementos e lógicas de programação.

Temática da unidade: pensamento criativo; pensamento computacional; reconhecimento de padrão e algoritmo.

Ferramenta utilizada: StartLab.

Formato dos projetos: animações interativas no StartLab.

Quantidade de aulas: 08

Objetivos de aprendizagem:

1. Compreender o funcionamento básico do ambiente de programação StartLab.
 2. Modificar personagens e cenários.
 3. Analisar padrões de código e suas aplicações em diferentes contextos.
 4. Aplicar algoritmos para modificar fantasias e cenários.
 5. Compreender as diferenças entre os blocos *repita* e *sempre*.
-

Habilidades:

- BNCC - Formação Geral Básica
(EF69AR06) Desenvolver processos de criação em artes visuais, com base em temas ou interesses artísticos, de modo individual, coletivo e colaborativo, fazendo uso de materiais, instrumentos e recursos convencionais, alternativos e digitais.
- BNCC - Computação
(EF06CO02) Elaborar algoritmos que envolvam instruções sequenciais, de repetição e de seleção usando uma linguagem de programação.

2

Conteúdo programático

AULA	CONTEÚDO	OBJETIVO DE APRENDIZAGEM
1	<ul style="list-style-type: none"> Ambientalização no StartLab. Criação e edição de personagens. Composição e personalização de personagens. 	<ol style="list-style-type: none"> Selecionar personagens na biblioteca do ambiente de programação. Duplicar e editar personagens para criar novos personagens combinados. Renomear os personagens criados.
2	<ul style="list-style-type: none"> Revisão e continuação do projeto anterior. Introdução à programação no StartLab. Criação de sequências de animação. 	<ol style="list-style-type: none"> Conhecer a área de código do StartLab. Localizar os blocos de eventos, controle e aparência. Implementar um algoritmo para alterar fantasias.
3	<ul style="list-style-type: none"> Introdução à programação no StartLab. Adição e modificação de fantasias no StartLab. Otimização de códigos com blocos de controle. 	<ol style="list-style-type: none"> Reconhecer padrões e automatizar tarefas. Depurar e refinar códigos para eficiência. Adicionar, modificar e nomear os personagens no StartLab.

AULA	CONTEÚDO	OBJETIVO DE APRENDIZAGEM
4	<ul style="list-style-type: none"> • Seleção e adição de cenários. • Copiando elementos visuais. • Diferenças entre as ferramentas de edição. 	<ol style="list-style-type: none"> 1. Diferenciar atores e cenários no StartLab. 2. Modificar cenários no StartLab. 3. Comparar e diferenciar ferramentas de edição de imagens.
5	<ul style="list-style-type: none"> • Criação de cenários personalizados. • Programação da troca de cenários. • Salvamento e compartilhamento do projeto. 	<ol style="list-style-type: none"> 1. Modificar cenários variados. 2. Implementar o algoritmo que troca os cenários. 3. Compartilhar o projeto finalizado.

3

Orientações pedagógicas

Pensando em prepará-lo(a) para a condução das aulas, construímos uma formação, em vídeo, sobre esta unidade com o objetivo de fornecer mais insumos para sua prática docente. Você pode acessá-la na plataforma Start. A seguir, temos um resumo, em texto, das orientações didáticas que você encontrará nas videoaulas da formação.

Ao longo das primeiras aulas, conduza a turma a experimentar os blocos de eventos, controle e aparência, sempre mostrando a relação entre o que é programado e o que acontece na tela, incentivando a personalização e a combinação de atores disponíveis na galeria. Provoque a curiosidade com perguntas norteadoras e exemplos visuais, estimulando a criatividade na criação de personagens e cenários novos.

Durante o desenvolvimento do projeto *Meu mundo*, explore a ideia de criação de “híbridos” como estratégia interdisciplinar, conectando programação, artes e ciências. Oriente os estudantes a duplicar personagens, alterar fantasias e cenários e observar padrões de repetição no código. Utilize exemplos concretos para discutir quando usar blocos como *sempre* e *repita*, incentivando a organização e a otimização dos algoritmos. Estimule a troca entre os colegas e reserve momentos para compartilhar os projetos, fortalecendo a autoria.

Incentive-os a otimizarem o código utilizando blocos como *repita* e *sempre*, que ajudam a eliminar repetições e aumentam a eficiência do script. Destaque as diferenças entre os blocos de atores e cenários, explicando que os atores possibilitam movimento e interação, enquanto os cenários são usados para modificar o ambiente visual. Utilize atividades práticas para explorar essas distinções e ampliar as possibilidades criativas dos projetos.

Finalize a unidade estimulando a experimentação e a combinação de códigos já conhecidos para criar efeitos mais complexos, mostrando que reutilizar e adaptar programações faz parte do processo criativo. Acompanhe o desempenho dos estudantes por meio da observação contínua, considerando critérios como algoritmos, originalidade e reconhecimento de padrões. Reserve um tempo para revisão, salvamento dos trabalhos e encerramento adequado das sessões, reforçando rotinas de segurança online. Promova um momento de socialização dos resultados, valorizando as conquistas e o percurso desenvolvido ao longo do projeto.

4

Materiais de apoio

Para auxiliar na condução das aulas desta unidade, disponibilizamos materiais de apoio aula a aula para professores e estudantes. Esses materiais podem ser acessados na plataforma Start, no campo “Material de apoio” de cada aula.

4.1 Professores

O **guia do professor** possui o passo a passo descritivo da aula, bem como os comandos necessários para a construção do projeto. Ele pode ser utilizado para o planejamento e preparação da aula, contando com direcionamentos pedagógicos, dicas para condução das aulas, explicação de conceitos de programação e soluções para possíveis dificuldades que os estudantes possam ter. Ao final de cada guia, há uma explicação para o desafio da aula, contando com a expectativa de resolução por parte dos estudantes.

O **gabarito do professor** possui a resolução comentada dos exercícios que os estudantes devem realizar em cada aula. Essas atividades funcionam como um complemento à prática de programação pela construção de projetos, fornecendo uma devolutiva rápida ao professor sobre a proficiência dos estudantes.

4.2 Estudantes

Os **slides dos estudantes** contêm um passo a passo descritivo, aula a aula, dos comandos que os estudantes deverão realizar para a construção do projeto da unidade. Por se tratar de um direcionamento da aula, ele pode ser disponibilizado individualmente ou projetado para a turma. Ao final de cada slide, há um desafio para aprofundar os conteúdos que foram trabalhados naquela aula, sendo uma ferramenta importante para a prática da programação e para fornecer uma atividade extra para aqueles estudantes que finalizarem mais rapidamente a construção dos códigos.

As **listas de exercícios dos estudantes** são um conjunto de atividades de fixação dos conteúdos que foram trabalhados em aula. É esperado que, a cada aula, os estudantes realizem os exercícios e tirem eventuais dúvidas sobre eles com o professor.

4.3 Rubrica de avaliação

Por fim, a rubrica é uma valiosa ferramenta para auxiliar na correção dos projetos que são submetidos pelos estudantes. Elas possuem quatro características a serem avaliadas de acordo com cada projeto da unidade e, para cada característica, há a descrição de quatro níveis de proficiência em que o estudante pode se encaixar. Dessa forma, ao avaliar um projeto, o professor consegue analisar o que é esperado em cada nível, fazendo com que sua correção seja baseada em critérios claros e que o feedback de desenvolvimento para o estudante seja melhor direcionado. Essas rubricas também podem ser acessadas no painel de gestão.

Acesse aqui a rubrica do projeto:

[Projeto 01 - Meu mundo](#)



Minhas anotações

Utilize o seu editor de PDF para realizar anotações neste espaço.

The background of the entire image is a colorful gradient from red at the top to green at the bottom, overlaid with a pattern of white geometric shapes like circles, squares, and hexagons. A grey web browser window frame is positioned at the top, containing three dots on the left and five colored icons (green hexagon, blue triangle, yellow circle, red square, purple pentagon) on the right, followed by a hamburger menu icon. The main content area of the browser is white and contains the 'start by alura' logo.

start

by alura



Contato:

Suporte

@startbyalura

www.startalura.com.br