

Aula 2

Reconhecendo padrões

► **Unidade**

**Segurança digital: utilizando
matemática para programar
senhas seguras**

O que vamos aprender?



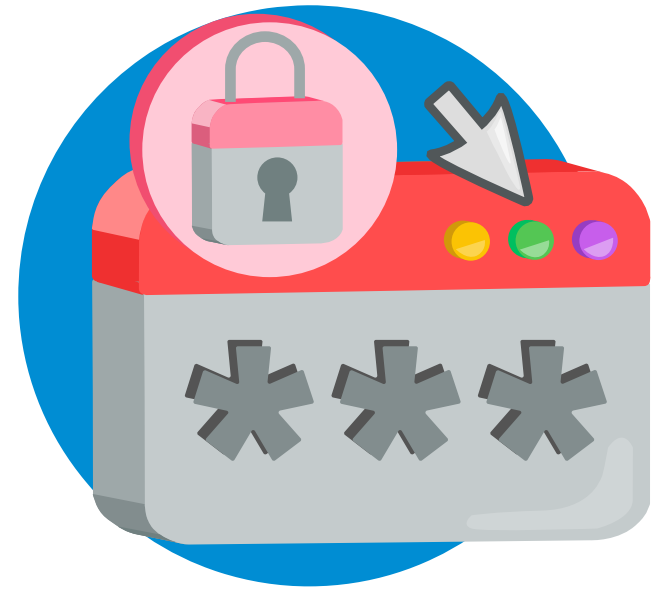
Estilizar o campo de inserção de senha.



Ajustar alinhamento dos textos e margens das seções.



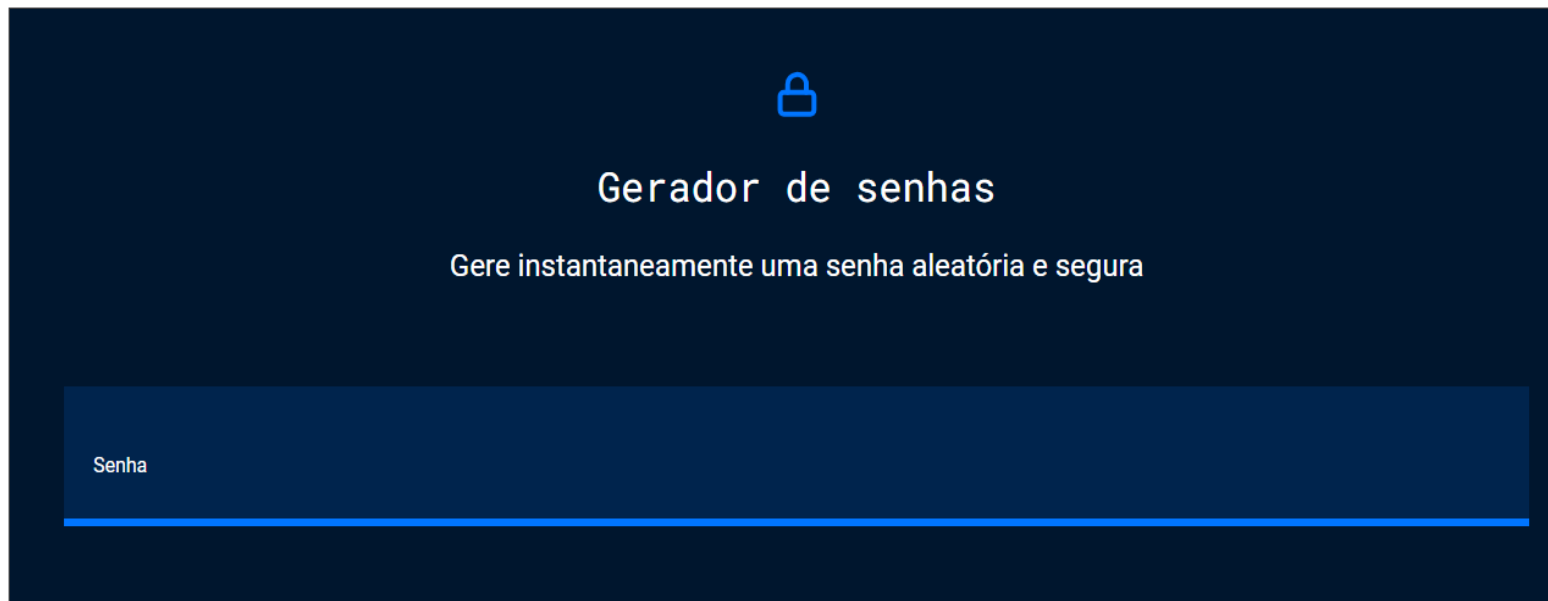
Criar mais estilizações utilizando CSS.



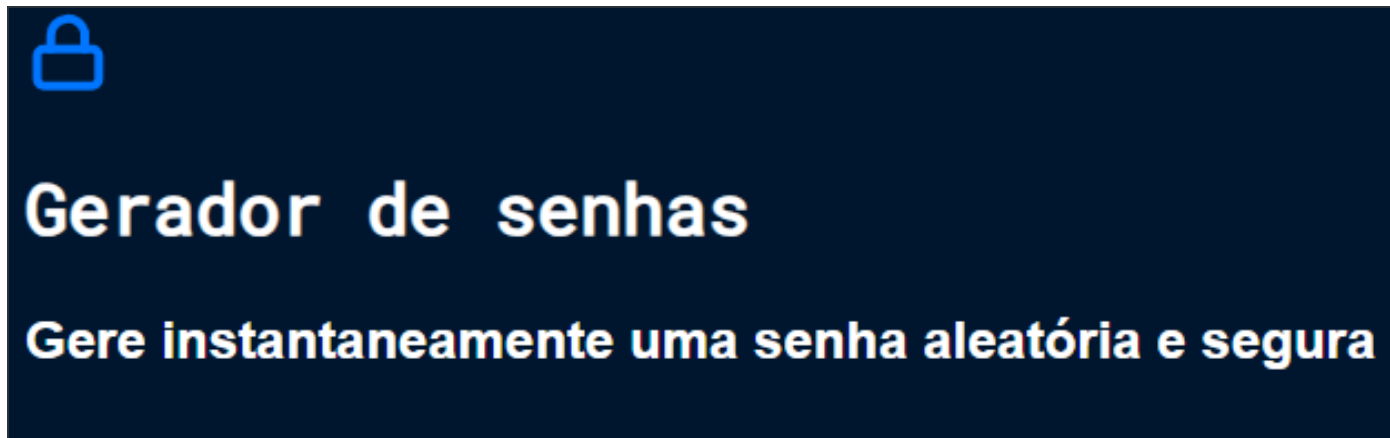
CLIQUE E ACOMPANHE A AULA NA ALURA

Estilizando a página

Na aula anterior, analisamos o nosso protótipo, disponibilizado no Figma, e realizamos as estilizações necessárias para criarmos um gerador de senhas, incluindo o título, subtítulo e imagem. Nosso objetivo agora é estilizar o campo de inserção de senha e ajustar as formatações conforme o modelo. Faremos isso utilizando HTML e CSS.

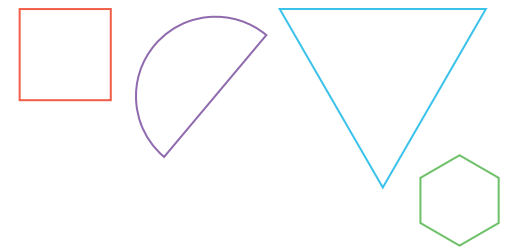


Finalizamos a aula anterior com a seguinte visualização do nosso projeto:



Percebemos que o texto está alinhado à esquerda e não condiz com o cabeçalho do protótipo. No Figma há, inclusive, um espaçamento no topo, entre o título e a página. Então, precisamos adicionar esses estilos ao nosso gerador de senha.

Se, no HTML, tivermos que trocar seletor por seletor para centralizar (**center**) o texto, repetiremos o processo diversas vezes. Dessa forma, podemos reconhecer esse padrão e agrupar os elementos que têm essa mesma estilização.



Para isso, criaremos uma divisão `<div>` para o conteúdo: **div.conteúdo-titulo**. Em seguida, adicionaremos todos os itens dentro das tags `<div></div>`, como mostrado a seguir:

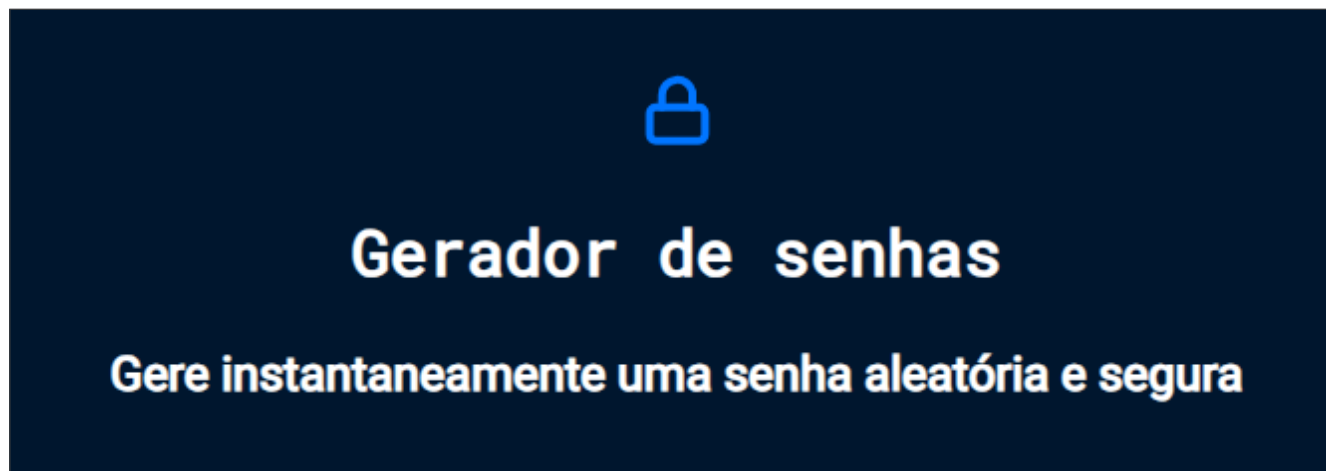
```
<div class="conteudo-titulo">  
    
  <h2 class="titulo-principal">Gerador de senhas</h2>  
  <h3 class="titulo-secundario">Gere instantaneamente  
uma senha aleatória e segura</h3>  
</div>
```

No `style.css`, podemos adicionar `.conteudo-titulo{}` e colocar a propriedade `text-align: center;`:

```
.conteudo-titulo{  
  text-align: center;  
}
```

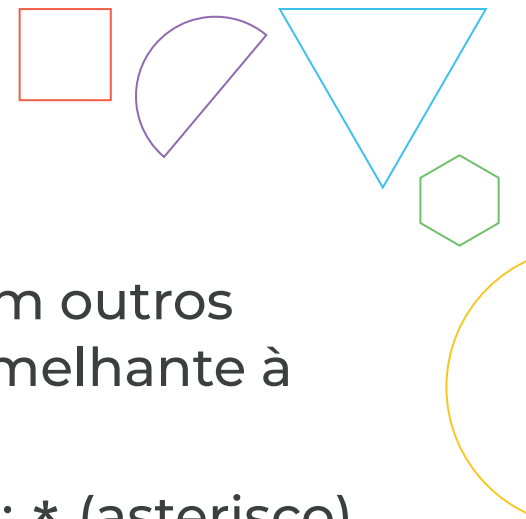
Se visualizarmos o projeto, identificaremos que a imagem do cabeçalho fica colada ao topo da página, diferente do que temos no protótipo.

Para resolver isso, adicionaremos **margin** ou **padding** dentro do seletor do título. Neste caso, usaremos **margin-top: 80px;**. Agora sim, ao atualizar a página, teremos:



Percebemos que os elementos estão estilizados na mesma proporção do protótipo, ou seja, a aplicação das margens está correta.

Em seguida, observem que a fonte do subtítulo tem uma grafia um pouco mais fina no protótipo. Para configurar esse efeito, em **.titulo-secundario**, colocaremos **font-weight: 400;**.

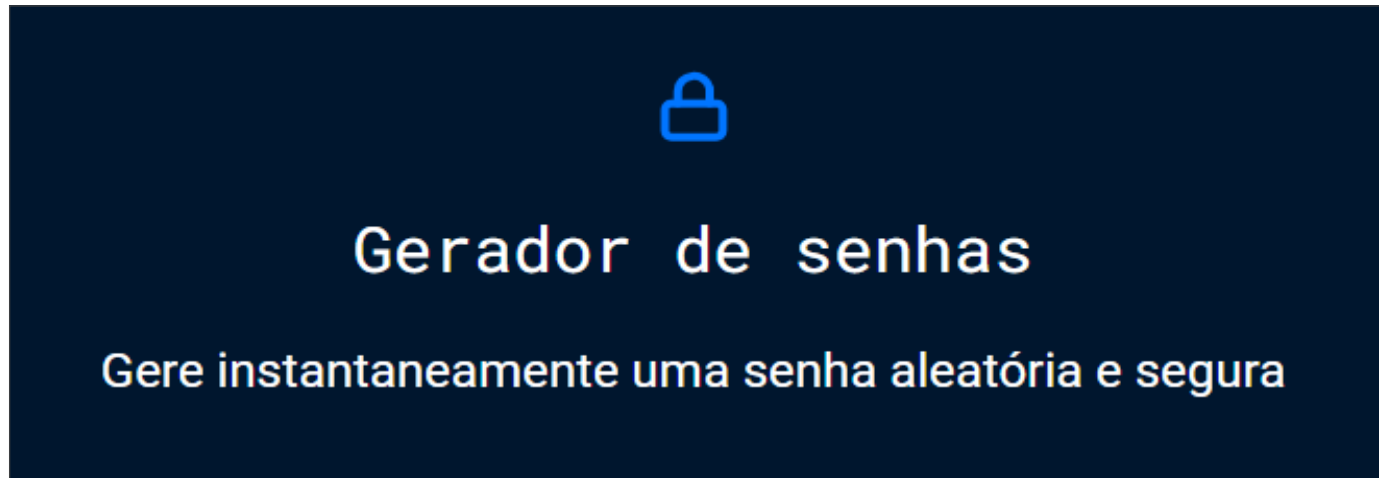


Olhando atentamente para o protótipo, veremos que existem outros elementos no projeto que também possuem uma grafia semelhante à do subtítulo, com letras mais finas.

Para configurar esse padrão, usaremos um seletor universal: `*` (asterisco), e um abre chaves `{ }` e ao adicionar **`font-weight: 400;`**. Assim, seu código ficará da seguinte forma:

```
*{  
    font-weight: 400;  
}
```

Observe o resultado ao atualizar a página do projeto:

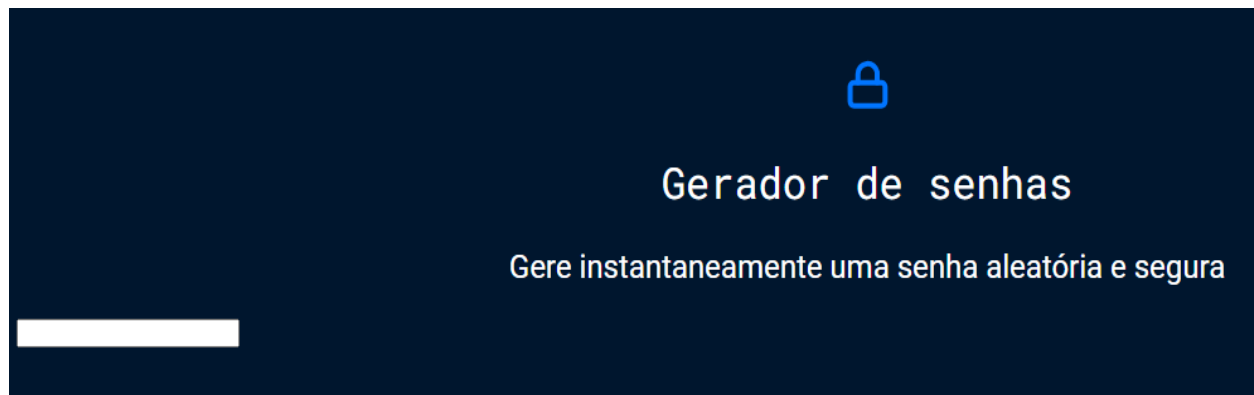


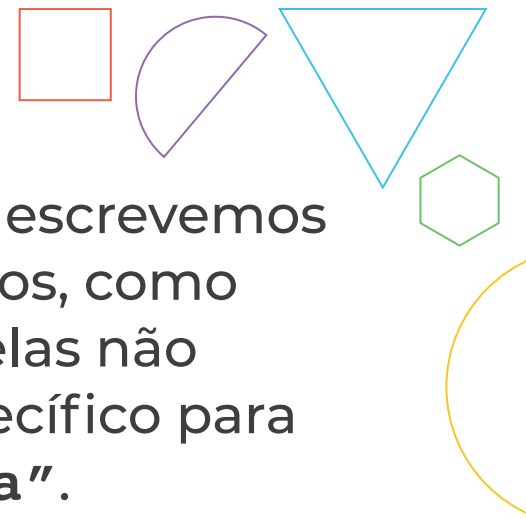
Agora, caso seja necessário adicionar um peso diferente para as outras classes, ficará mais fácil utilizar **font-weight: 400px;**. Após finalizarmos o título, vamos configurar a parte de baixo do projeto.

Ao analisarmos nosso modelo, identificamos que existe uma cor de fundo para a seção de senha diferente do fundo da seção cabeçalho, e ela tem um padrão. Para configurar esse estilo, criaremos uma nova **div**, colocando o texto Senha e o resultado da senha, **div.conteudo-senha**. Utilizaremos um campo que será gerado pelo JavaScript e, para isso, podemos usar uma tag um pouco diferente, a tag **<input>**. Nela, especificamos o tipo (**type**) da informação a ser inserida. No nosso caso, teremos **text** (texto). Observe:

```
<div class="conteudo-senha">  
  <input type="text">  
</div>
```

Ao atualizar a página, veremos que um campo que permite a inserção de informações foi criado:





Observe que o novo campo criado está em branco. Quando escrevemos nele, sua estilização não está dentro do padrão que adotamos, como as estilizações implementadas em **:root** e **.body**. Já que elas não estão sendo aplicadas, teremos que colocar algo muito específico para identificarmos esse campo, como um id: **id="campo-senha"**.

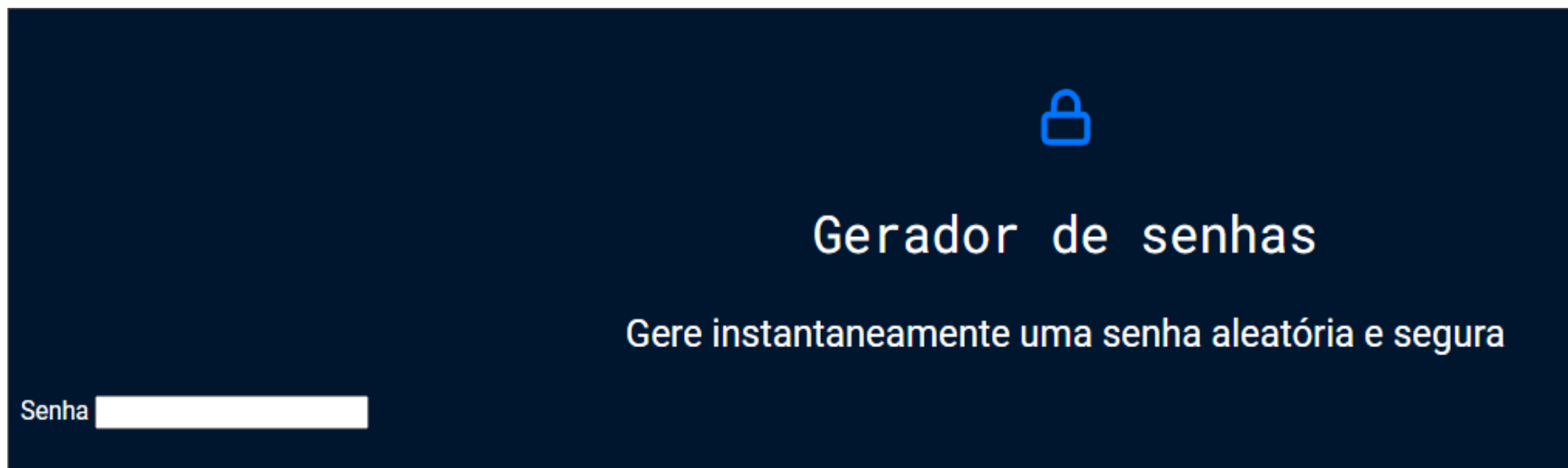
Perceba que, se deixarmos somente uma caixa de preenchimento sem nenhuma orientação, o usuário não saberá o que fazer. Por isso, ao lado esquerdo desse campo, colocaremos o texto Senha. Para fazermos esse processo antes do **input**, vamos usar outra tag chamada **<label>** **</label>**.

Junto dessa tag existe um atributo chamado **for**, que não é o mesmo com o qual já estamos acostumados no JavaScript. Esse **for** significa que, quando escrevemos uma **label**, espera-se que ela tenha um **input**, colocado também dentre as tags de abertura e fechamento da label.


Precisamos deixar bem claro que essa label pertence àquele input, então, é necessário linkar estes dois elementos: **for="senha"**. Dentro do input, precisamos adicionar mais um elemento que será específico: **name="senha"**. Esse processo funciona como se você fosse mandar uma carta para uma pessoa, *for*, e precisa identificar quem recebe, *name*. Seu código ficará da seguinte forma:

```
<div class="conteudo-senha">  
  <label for="senha">Senha</label>  
  <input name="senha" type="text" id="campo-senha">  
</div>
```

Ao atualizar a página do projeto, teremos a identificação do campo:

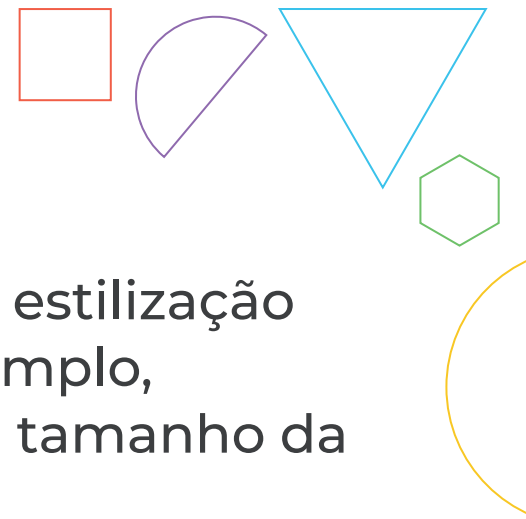


The screenshot shows a web application with a dark blue background. At the top center is a blue padlock icon. Below it, the text "Gerador de senhas" is displayed in a large, white, sans-serif font. Underneath that, in a smaller white font, is the text "Gere instantaneamente uma senha aleatória e segura". At the bottom left, the word "Senha" is followed by a white rectangular input field.



Podemos agora estilizar estes dois elementos: o campo de senha e seu respectivo texto. Sabemos que existe uma cor de fundo diferente nessa seção. Assim, no arquivo *style.css*, adicionaremos os estilos para `<div class="conteudo-senha">`, `.conteudo-senha{}`. A cor de fundo precisa ser alterada e existem espaçamentos entre a escrita e a caixa, além de uma borda na parte inferior no campo senha. Ao realizar todas essas configurações, seu código ficará da seguinte forma:

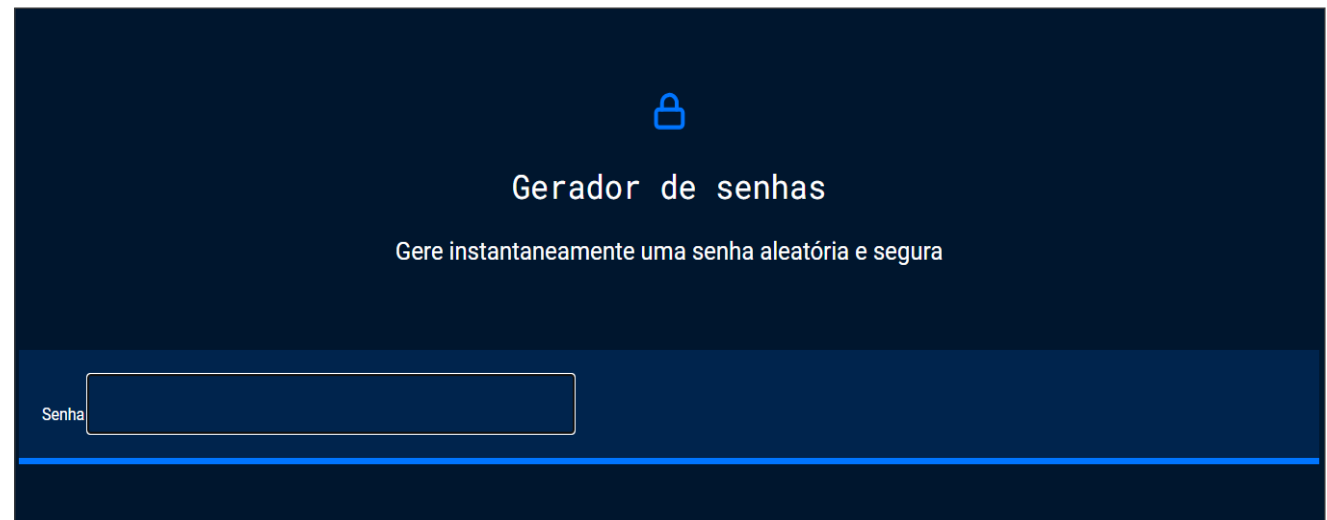
```
.conteudo-senha{  
  margin-top: 80px;  
  background: var(--fundo-senha);  
  padding: 24px;  
  border-bottom: 6px solid var(--borda);  
}
```



Podemos seguir editando outros elementos, já que temos a estilização da página. Quando usarmos o **id="campo-senha"**, por exemplo, precisamos mudar a cor de fundo, retirar a borda, cor, tipo e tamanho da fonte. Observe essas alterações no código a seguir:

```
#campo-senha{  
  background-color: var(--fundo-senha);  
  border: none;  
  color: var(--branco);  
  font-family: var(--roboto-mono);  
  font-size: 40px;  
}
```

Ao atualizar a página do projeto e observar as mudanças realizadas, veremos que o campo de senha está mais semelhante ao do protótipo:




Perceba que, ao selecionarmos o campo de senha, existe uma borda. Não queremos esse efeito. Para corrigir isso, existe uma pseudoclasse que precisamos trocar. Assim, ao invés de escrever direto em **#campo-senha**, vamos escrever em **#campo-senha:focus**:

```
#campo-senha:focus{  
  outline: none;  
}
```

Dessa forma, quando selecionarmos o elemento campo de senha, seu outline (linha de fora) precisa ser **none**, ou seja, quando clicarmos naquele campo, não existirá mais a borda.

Além disso, também é possível identificar que a largura não condiz com o nosso protótipo. Desse modo, podemos criar uma divisão para agrupar todos os itens e mantê-los em um padrão. Para isso, logo abaixo de **<body>**, criaremos uma **<section>** de **conteudo** e tudo vai ficar dentro desta. Na última linha, colocaremos **</section>**, logo após **</div>**. Observe:

```
<body>
  <section class="conteudo">
    <div class="conteudo-titulo">
      
      <h2 class="titulo-principal">Gerador de senhas</h2>
      <h3 class="titulo-secundario">Gere instantaneamente
uma senha aleatória e segura</h3>
    </div>
    <div class="conteudo-senha">
      <label for="senha">Senha</label>
      <input name="senha" type="text" id="campo-senha">
    </div>
  </section>
</body>
```



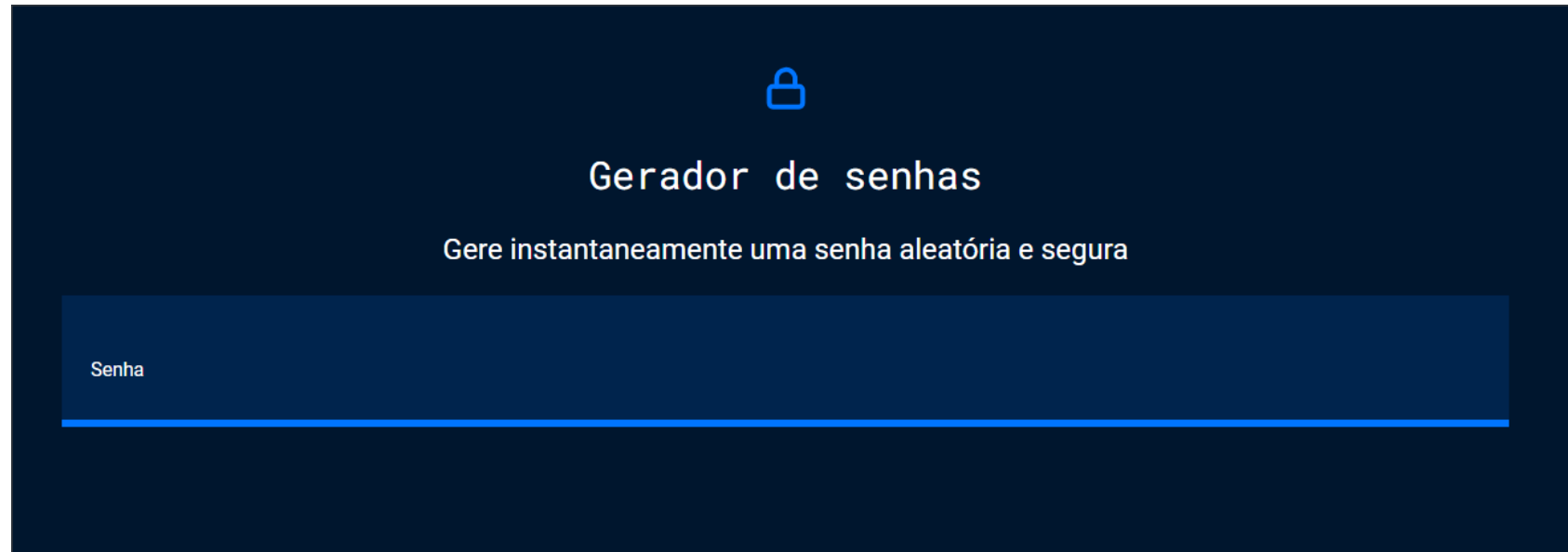
No arquivo *style.css*, adicionaremos a classe **.conteudo** e ajustaremos sua largura máxima, ou seja, se esse elemento passar de um valor, não aumentará mais que o estipulado:

```
.conteudo{  
    max-width: 1200px;  
}
```

No protótipo, ainda temos um espaço do lado direito, que precisamos ajustar em nosso projeto. Para isso, adicionaremos a propriedade **margin:0** para cima e para baixo e **auto** para direita e esquerda, da seguinte forma:

```
.conteudo{  
    max-width: 1200px;  
    margin: 0 auto;  
}
```

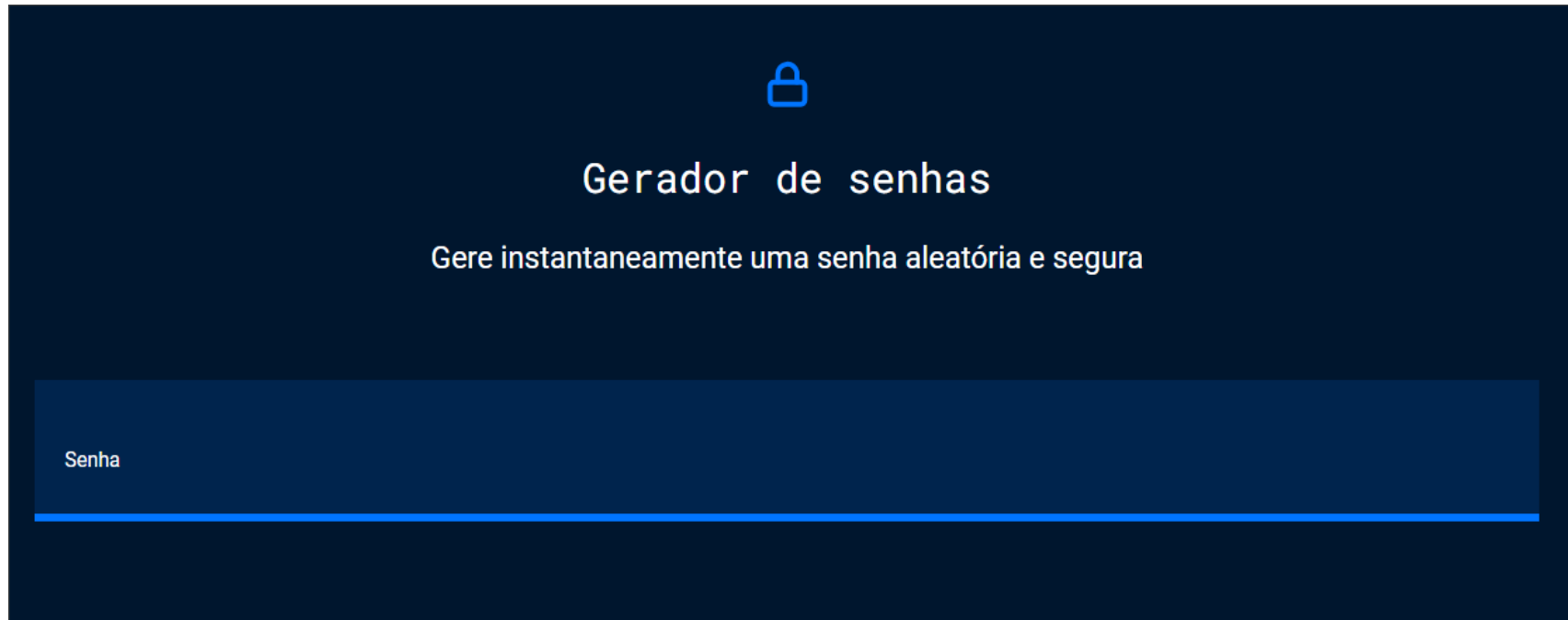

Feitas essas configurações, ao atualizarmos nossa página, teremos:



Observe que o espaçamento entre o cabeçalho e a seção de senha ainda é pequeno. Para resolver isso, adicionaremos um espaçamento externo **`margin-top: 80px;`** ao campo de senha; observe:

```
.conteudo-senha{  
  margin-top: 80px;  
  background: var(--fundo-senha);  
  padding: 24px;  
  border-bottom: 6px solid var(--borda);  
}
```

Tendo realizado todas essas confirmações, nosso projeto está muito mais parecido com o protótipo do Figma:



Porém, ainda faltam alguns itens a serem configurados. Na próxima aula, continuaremos a estilização do projeto e o desenvolvimento da nossa aplicação.

Até breve!

► Desafio

Nesta aula, aprendemos a identificar padrões e a criar um campo, devidamente identificado, para inserção de informações. Seu desafio será pesquisar como desenvolver um formulário, buscando saber quais campos devemos ter e como esses campos se comportam. Compartilhe suas ideias e pesquise com seus colegas.



CLIQUE **AQUI** PARA AVALIAR ESTE MATERIAL