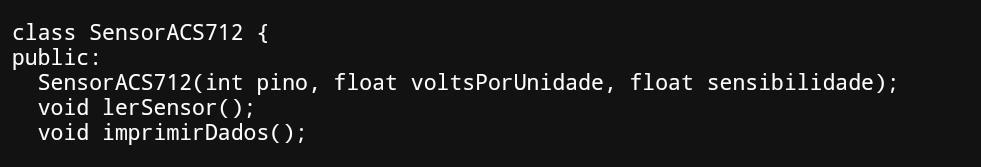
**Programação Orientada a Objetos**

Isabella Estella de Oliveira - RA: **1962380**

O projeto que criamos consiste na criação de um site que registra o seu consumo de energia, e permite que você monitore ele através da atualização em tempo real dos watts/volts/amperes, defina metas de consumo e ver o quanto de energia você consumiu em outros dias. Esses dados de consumo são gerados a partir de um sensor ACS712, ligado através de uma esp32.  
Grupo: Amanda Acosta, Giovana Castro, Giovana Oliveira, Isabella Estella, Marcela Coneglian.

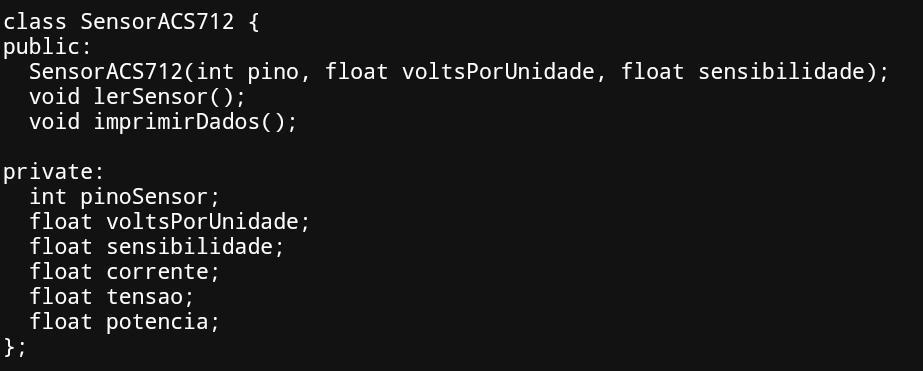
**Identificou e codificou classes de dados:**

Uma classe chamada SensorACS712 foi desenvolvida com o propósito de representar uma estrutura que organiza e gerencia os dados adquiridos pelo sensor ACS712.

****

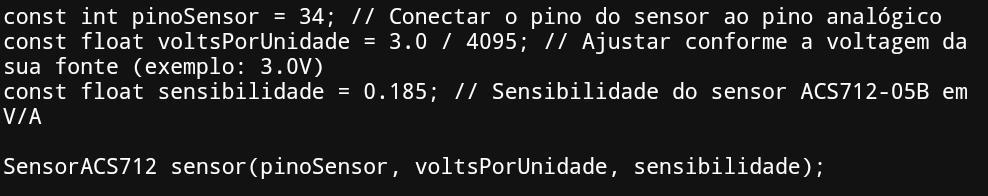
**Identificou e codificou classes de comportamento:**

Utilizando um objeto da classe SensorACS712, foi possível definir as ações e funcionalidades correspondentes a esses objetos.

****

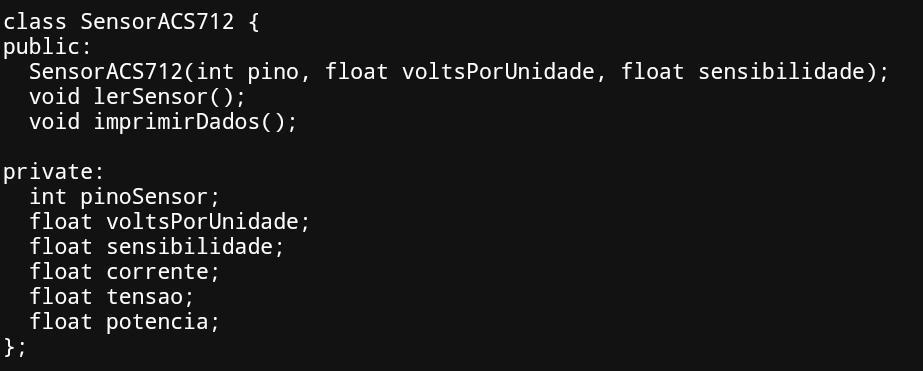
**Usou objetos imutáveis:**

Exemplos como o número do pino do sensor, a voltagem por unidade e a sensibilidade do sensor são exemplos de elementos que permanecem inalterados, mantendo sua constância ao longo do processo.

****

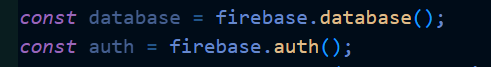
**Ocultou informações usando atributos e ou métodos protected/private:**

Na classe SensorACS712, adotamos os modificadores de acesso public e private. O private foi utilizado nos atributos pinoSensor, voltsPorUnidade, sensibilidade, corrente, tensão e potência, restringindo o acesso a esses elementos somente à própria classe. Já o public foi utilizado nos métodos construtor SensorACS712, lerSensor e imprimirDados, possibilitando sua utilização em outras partes do código.

****

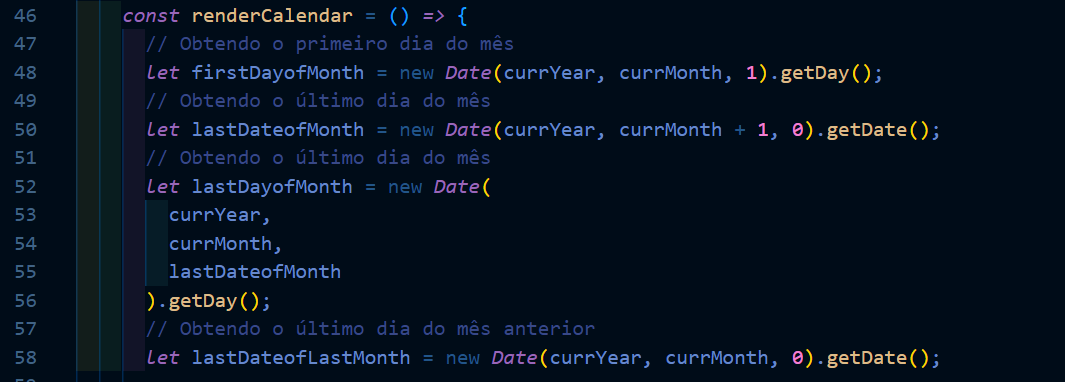
**Instanciou objetos:**

Criamos instâncias de uma classe e atribuímos os objetos "database" e "auth" a variáveis para facilitar sua utilização ao longo do código.



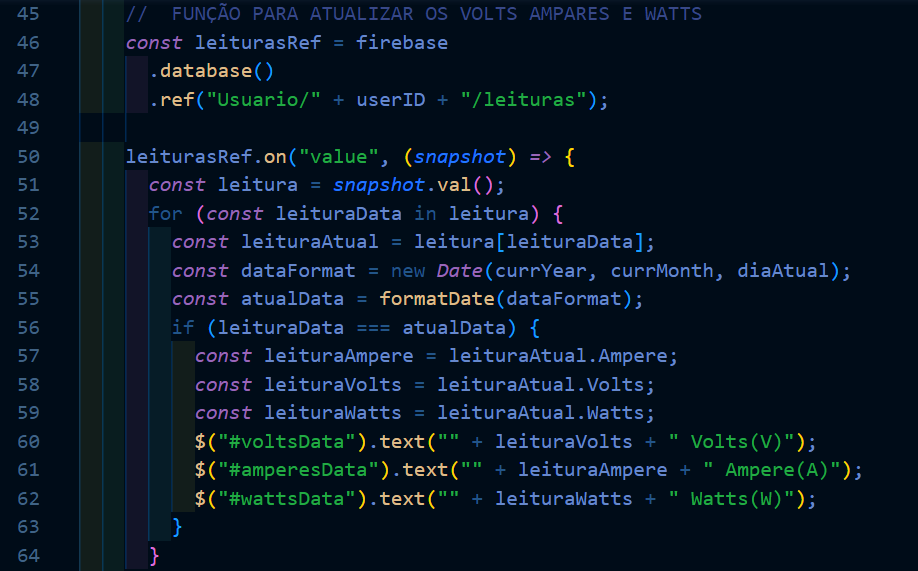
**Usou conceitos de código limpo:**

Implementamos conceitos de código limpo ao empregar funções claras e específicas, tornando o código mais legível e facilitando o entendimento do mesmo. Embora alguns princípios de código limpo possam não estar totalmente presentes, é possível visualizar as práticas de organização.



**Codificou propriedades:**

No código, são integradas propriedades que atribuem características ou atributos aos objetos, como as propriedades associadas ao objeto de leitura.



**Codificou métodos destrutores:**

Implementamos métodos destrutores no código para realizar operações de limpeza, removendo dados específicos do armazenamento quando necessário. Isso contribuiu para uma gestão eficiente dos recursos e uma manutenção mais organizada do código.

