LUIS MAÑEZ

Microsoft® Most Valuable Professional

DALL-E ❤ SPFx

Using DALL-E OpenAI Image generator API from SPFx

#SharePoint Framework community call 9th March 2023

About Me

Luis Mañez

Chief Architect @ ClearPeople

M365 Development MVP. More than 20 years of experience working with Microsoft Technologies. MS Certified specialist in Developing Azure and SharePoint MCPD. Passionate about technology, blogger, technical writer and MS community active member.

@luismanez

DALL·E 2

DALL·E 2 is an AI system that can create realistic images and art from a description in natural language.

Image generation | Outpainting | Inpainting | Variations

DALL·E 2 can create original, realistic images and art from a text description. It can combine concepts, attributes, and styles.

Input

An astronaut riding a horse in photorealistic style.

Output

Image generation | Outpainting | Inpainting | Variations

DALL·E 2 can expand images beyond what's in the original canvas, creating expansive new compositions.

Input

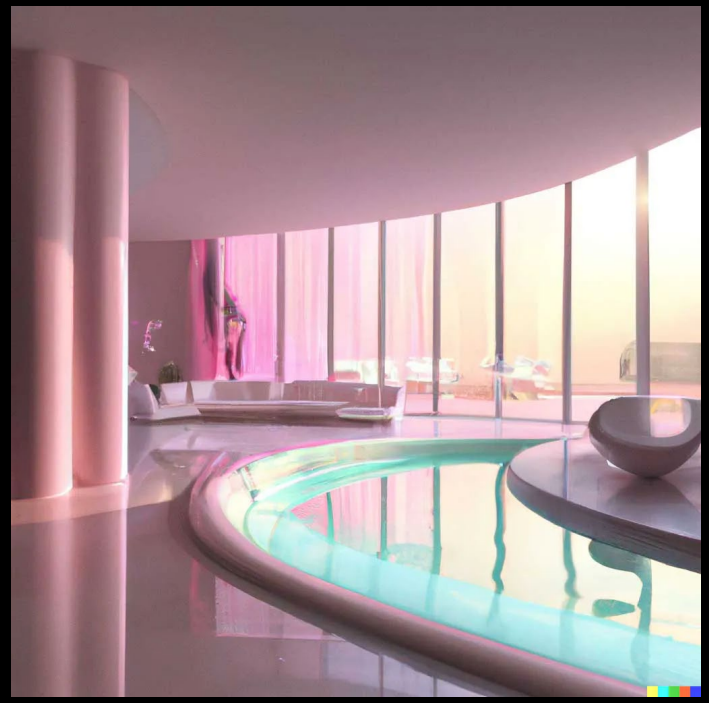
Output

Image generation Outpainting **Inpainting** Variations

DALL·E 2 can make realistic edits to existing images from a natural language caption. It can add and remove elements while taking shadows, reflections, and textures into account.

Input

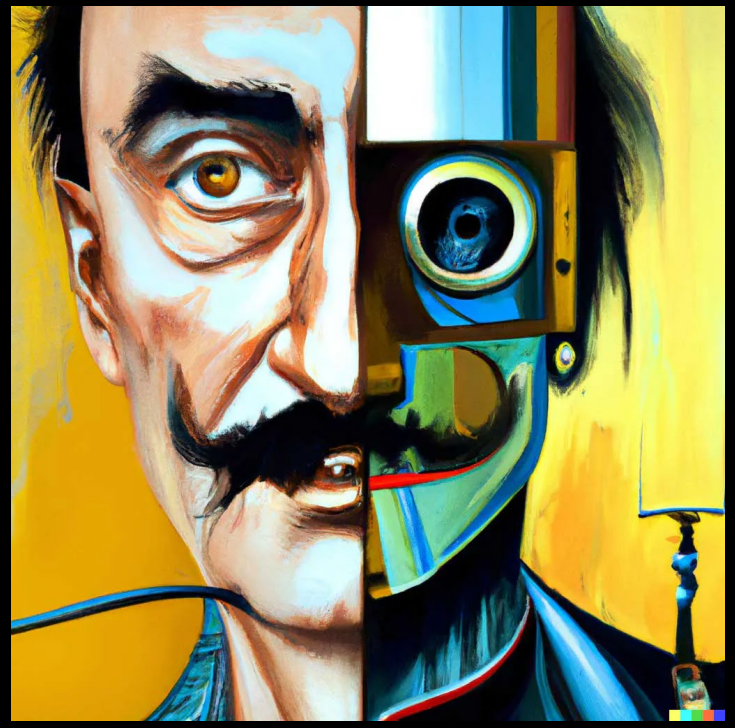
Add a flamingo beside the pool.



Output

DALL·E 2 can take an image and create different variations of it inspired by the original.

Input



Output

Billing

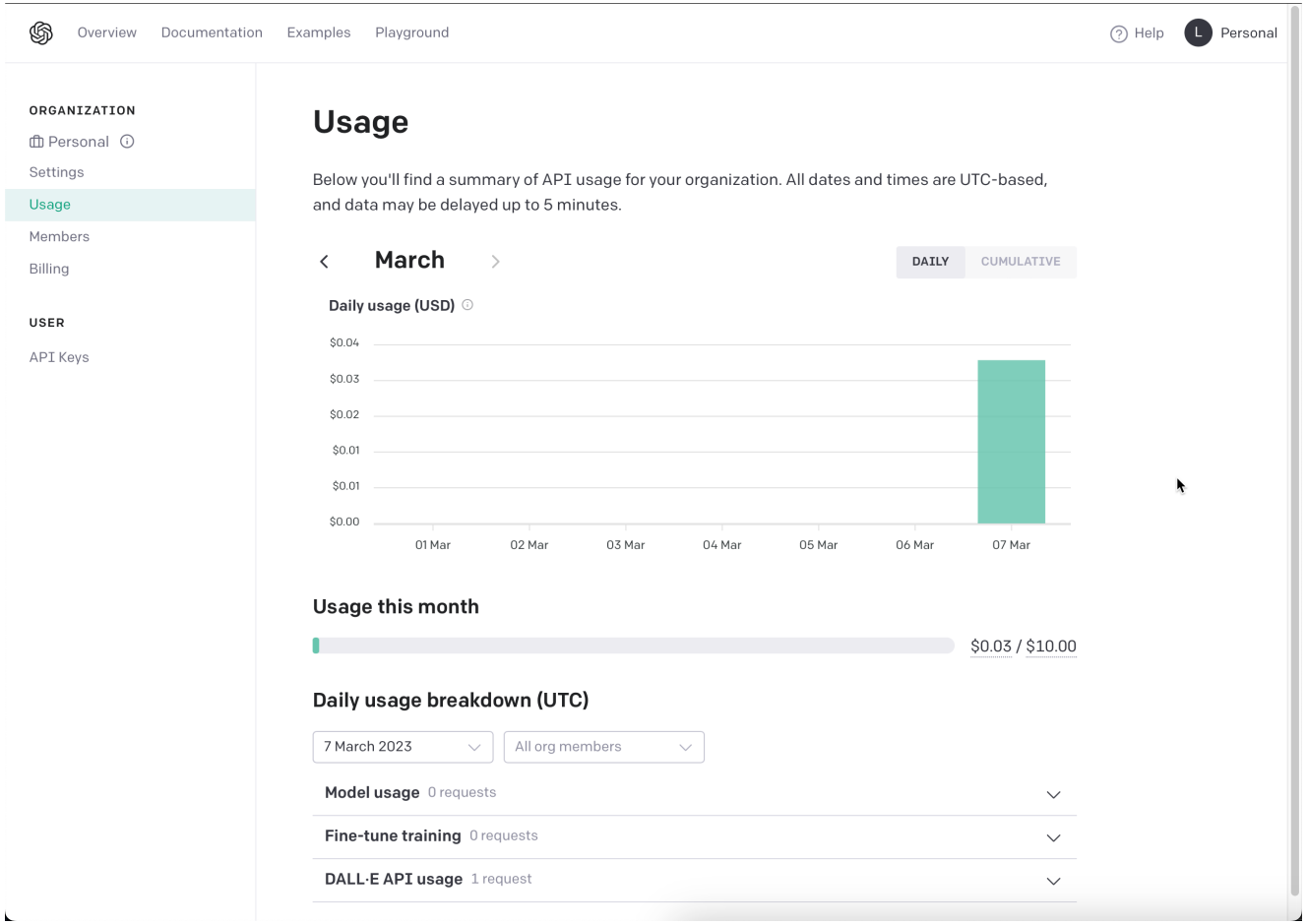
Image models

Build DALL·E directly into your apps to generate and edit novel images and art. Our image models offer three tiers of resolution for flexibility.

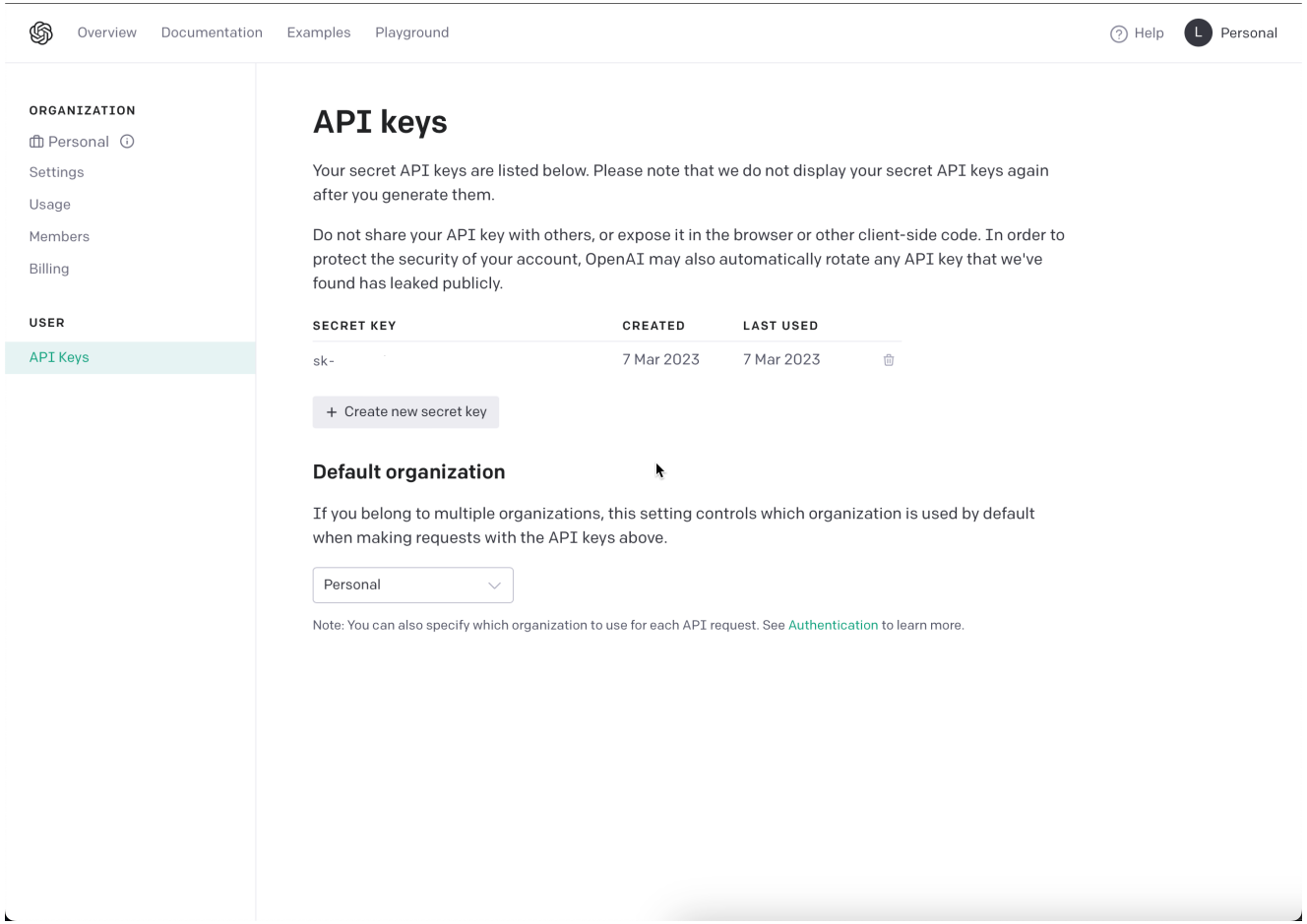
Learn more ↗

Resolution	Price
1024×1024	$0.020 / image
512×512	$0.018 / image
256×256	$0.016 / image

Create an account



Generate an API Key



Create image Beta

`POST https://api.openai.com/v1/images/generations`

Creates an image given a prompt.

Request body

`prompt` string Required

A text description of the desired image(s). The maximum length is 1000 characters.

`n` integer Optional Defaults to 1

The number of images to generate. Must be between 1 and 10.

`size` string Optional Defaults to 1024x1024

The size of the generated images. Must be one of `256x256`, `512x512`, or `1024x1024`.

`response_format` string Optional Defaults to url

The format in which the generated images are returned. Must be one of `url` or `b64_json`.

`user` string Optional

A unique identifier representing your end-user, which can help OpenAI to monitor and detect abuse. **Learn more**.

Example request curl Copy

```
curl https://api.openai.com/v1/images/generati
  -H 'Content-Type: application/json' \
  -H 'Authorization: Bearer YOUR_API_KEY' \
  -d '{
  "prompt": "A cute baby sea otter",
  "n": 2,
  "size": "1024x1024"
}'
```

Parameters Copy

```
{
  "prompt": "A cute baby sea otter",
  "n": 2,
  "size": "1024x1024"
}
```

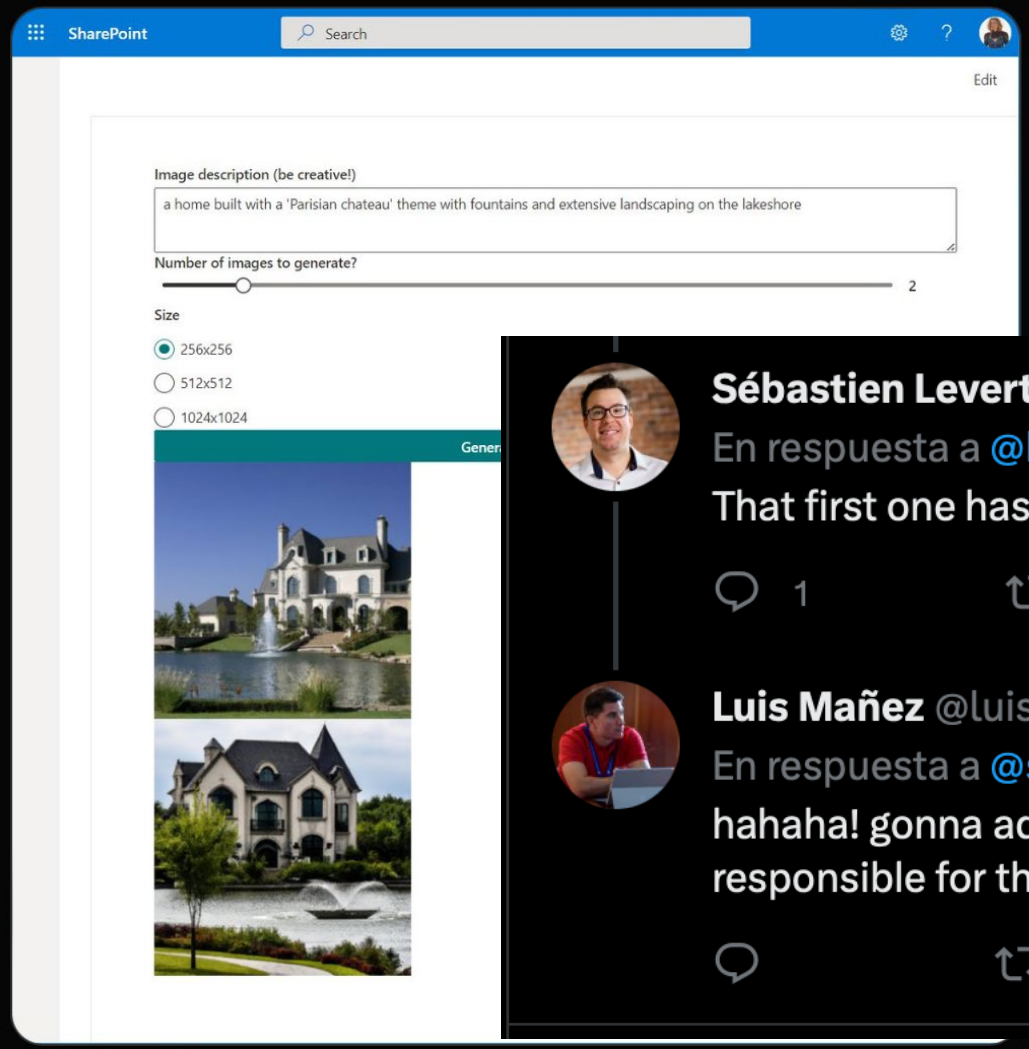
Response Copy

```
{
  "created": 1589478378,
  "data": [
    {
      "url": "https://..."
    },
    {
      "url": "https://..."
    }
  ]
}
```


Let´s See it in action!

Luis Mañez @luismanez · 5 nov. 2022

#SPFx spoiler alert! ... what about a webpart that generates images using DALL-E api? ... available soon. stay tune! #officedevpnp @m365pnp



1 5 26

DISCLAIMER

Sébastien Levert @sebastienlevert · 5 nov. 2022

En respuesta a @luismanez y @m365pnp

That first one has a weird driveway 🤣 Or they have a floating car!

1 4

Luis Mañez @luismanez · 5 nov. 2022

En respuesta a @sebastienlevert y @m365pnp

hahaha! gonna add a disclaimer message to the webpart "I´m not responsible for the generated content" XD

3

Let´s Dive In!

POST Generate images base6 | POST Generate images | Sambori Teamified

DALL-E Image generator / **Generate images** — Save

POST https://api.openai.com/v1/images/generations — Send

Params | Authorization | Headers (10) | Body | Pre-request Script | Tests | Settings — Cookies

none | form-data | x-www-form-urlencoded | raw | binary | GraphQL | JSON — Beautify

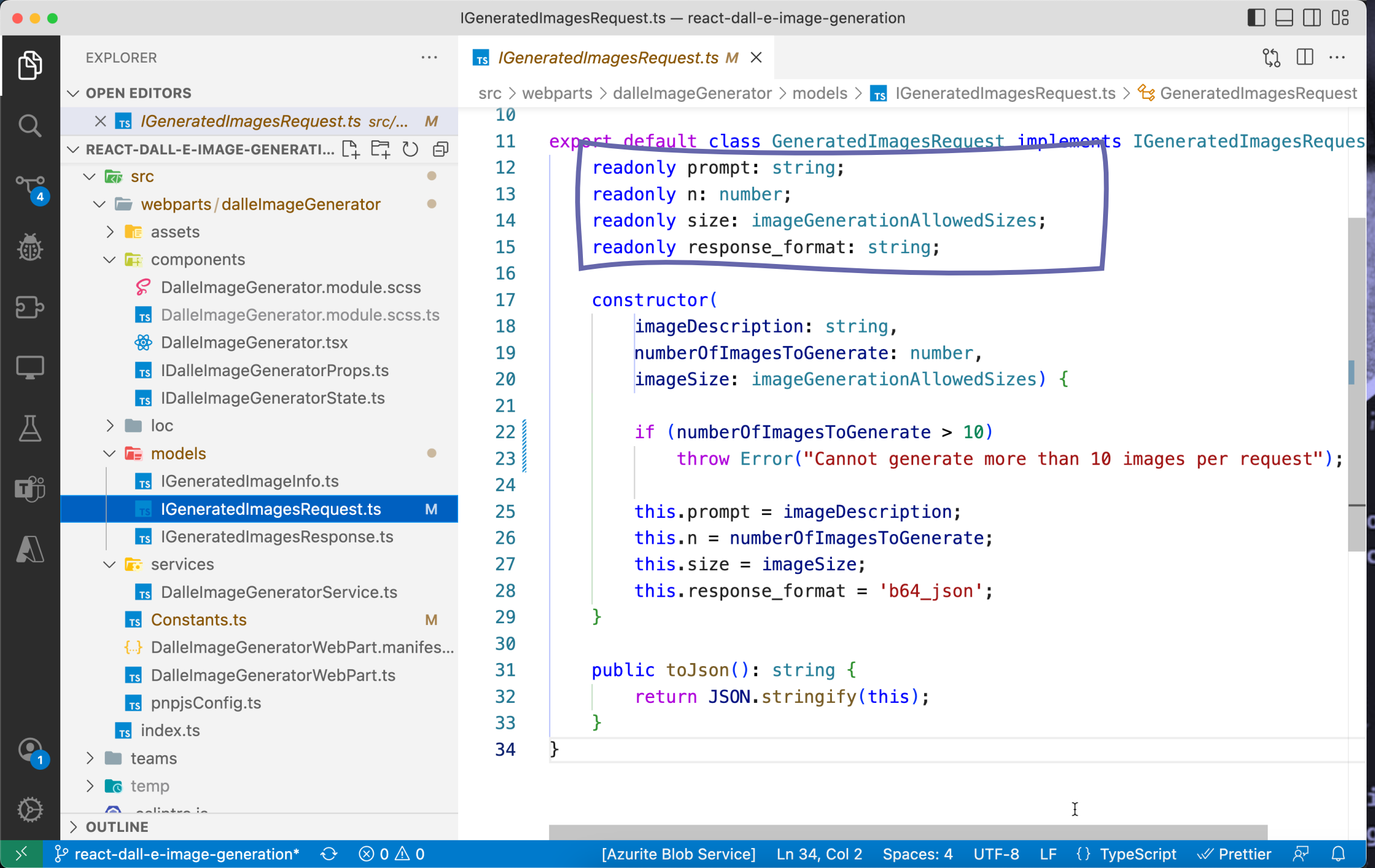
```
{
  "prompt": "a pencil and watercolor drawing of a bright city in the future with flying cars",
  "n": 2,
  "size": "256x256",
  "response_format": "url",
  "user": "luisman"
}
```

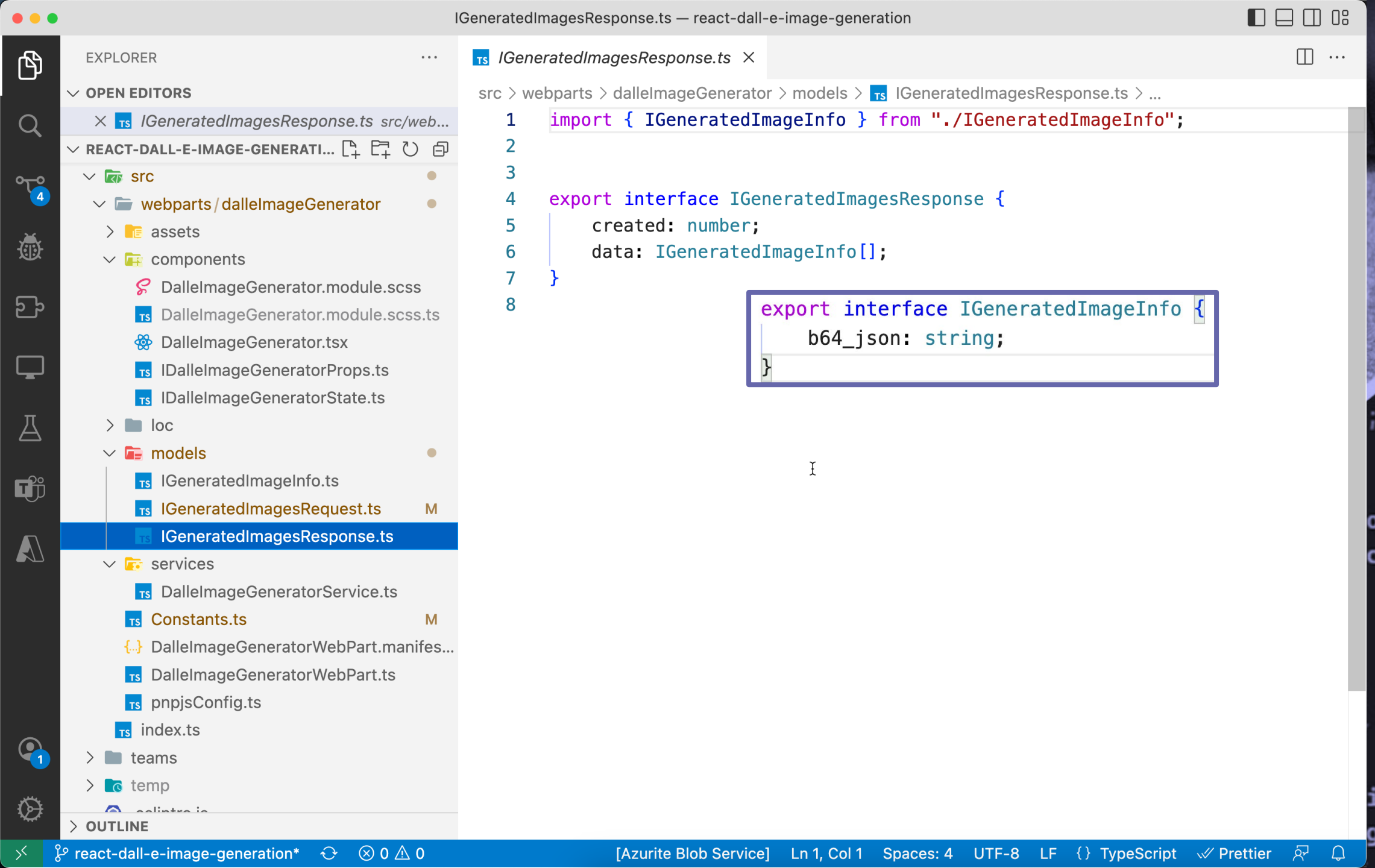
Body | Cookies | Headers (10) | Test Results — 200 OK 4.33 s 1.4 KB — Save Response

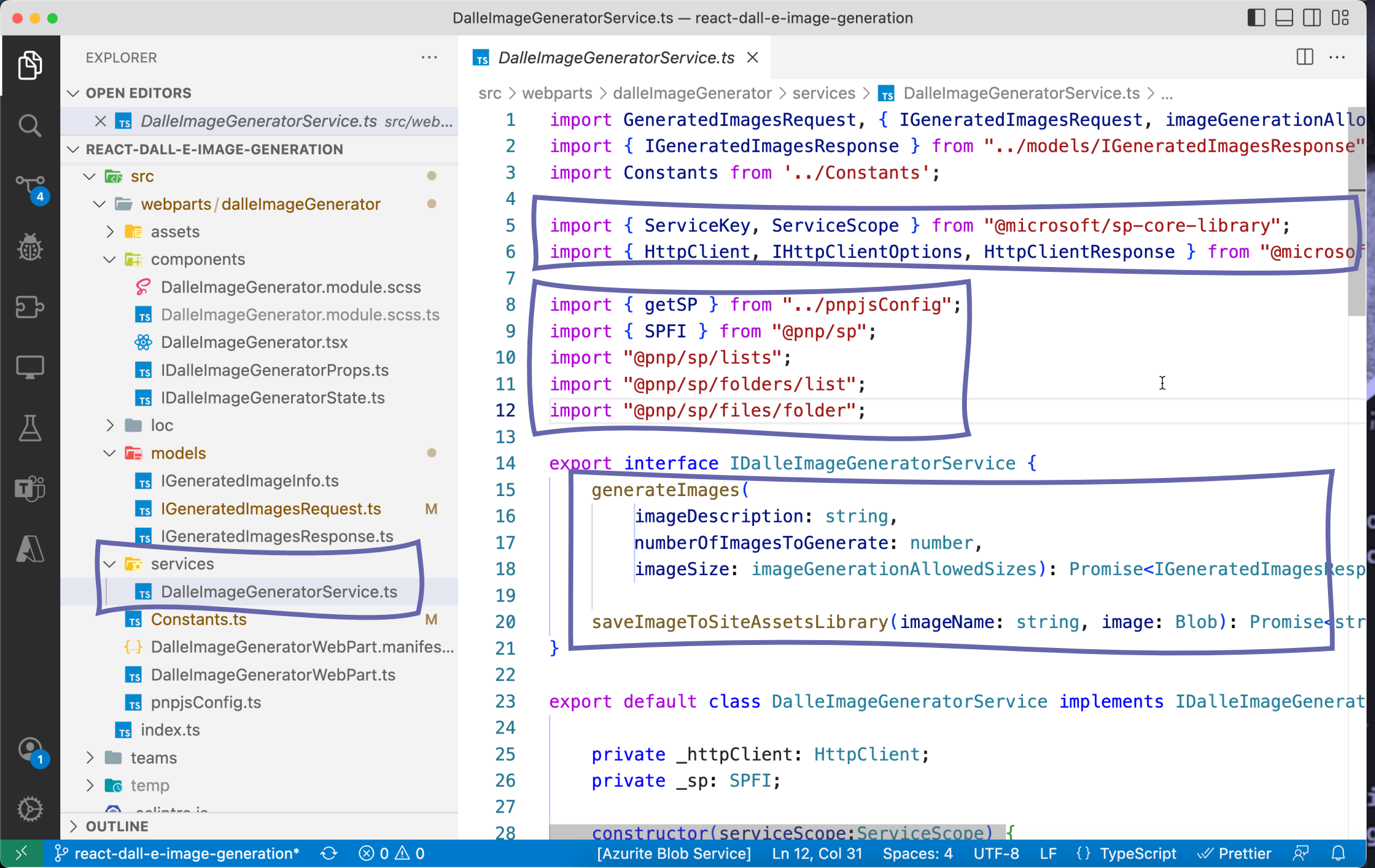
Pretty | Raw | Preview | Visualize | JSON

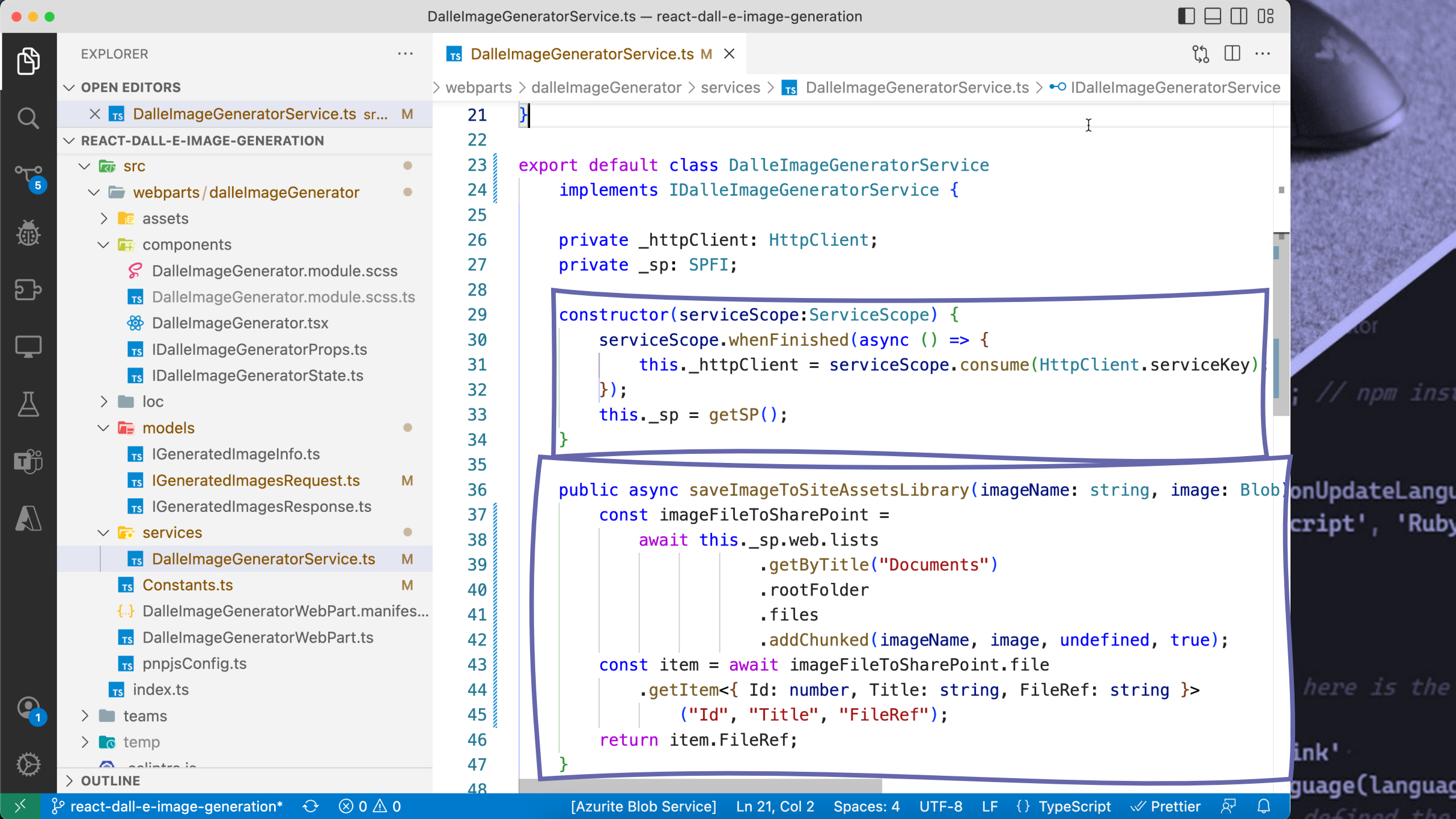
```
{
    "created": 1678268028,
    "data": [
        {
            "url": "https://oaidalleapiprodscus.blob.core.windows.net/private/org-j1v16Gdh4Y72ygxNBW1QpUKE/user-GX9JbKJYVKgl9oLyDvzLFJRz/img-QxMc4Uu8a8yxtpKgxmQvY90F.png?st=2023-03-08T08%3A33%3A48Z&se=2023-03-08T10%3A33%3A48Z&sp=r&sv=2021-08-06&sr=b&rscd=inline&rsct=image/png&skoid=6aaadede-4fb3-4698-a8f6-684d7786b067&sktid=a48cca56-e6da-484e-a814-9c849652bcb3&skt=2023-03-07T21%3A43%3A41Z&ske=2023-03-08T21%3A43%3A41Z&sks=b&skv=2021-08-06&sig=T2CGqVANdHz9tiKvmzDSqfRgBlBz26lxE/OcmleHF6s%3D"
        },
        {
            "url": "https://oaidalleapiprodscus.blob.core.windows.net/private/org-j1v16Gdh4Y72ygxNBW1QpUKE/user-GX9JbKJYVKgl9oLyDvzLFJRz/img-SAmWtLGNjkPHsHg8wHDW4rVC.png?st=2023-03-08T08%3A33%3A48Z&se=2023-03-08T10%3A33%3A48Z&sp=r&sv=2021-08-06&sr=b&rscd=inline&rsct=image/png&skoid=6aaadede-4fb3-4698-a8f6-684d7786b067&sktid=a48cca56-e6da-484e-a814-9c849652bcb3&
```

Cookies | Capture requests | Runner | Trash


IGeneratedImagesRequest.ts — react-dall-e-image-generation
EXPLORER
OPEN EDITORS
IGeneratedImagesRequest.ts src/... M
REACT-DALL-E-IMAGE-GENERATI...
src
webparts / dalleImageGenerator
assets
components
DalleImageGenerator.module.scss
DalleImageGenerator.module.scss.ts
DalleImageGenerator.tsx
IDalleImageGeneratorProps.ts
IDalleImageGeneratorState.ts
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest.ts M
IGeneratedImagesResponse.ts
services
DalleImageGeneratorService.ts
Constants.ts M
DalleImageGeneratorWebPart.manifes...
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
teams
temp
OUTLINE
IGeneratedImagesRequest.ts M
src > webparts > dalleImageGenerator > models > IGeneratedImagesRequest.ts > GeneratedImagesRequest
export default class GeneratedImagesRequest implements IGeneratedImagesReques
readonly prompt: string;
readonly n: number;
readonly size: imageGenerationAllowedSizes;
readonly response_format: string;
constructor(
imageDescription: string,
numberOfImagesToGenerate: number,
imageSize: imageGenerationAllowedSizes) {
if (numberOfImagesToGenerate > 10)
throw Error("Cannot generate more than 10 images per request");
this.prompt = imageDescription;
this.n = numberOfImagesToGenerate;
this.size = imageSize;
this.response_format = 'b64_json';
}
public toJson(): string {
return JSON.stringify(this);
}
}
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 34, Col 2
Spaces: 4
UTF-8
LF
TypeScript
Prettier


IGeneratedImagesResponse.ts — react-dall-e-image-generation
EXPLORER
OPEN EDITORS
IGeneratedImagesResponse.ts src/web...
REACT-DALL-E-IMAGE-GENERATI...
src
webparts / dalleImageGenerator
assets
components
DalleImageGenerator.module.scss
DalleImageGenerator.module.scss.ts
DalleImageGenerator.tsx
IDalleImageGeneratorProps.ts
IDalleImageGeneratorState.ts
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest.ts M
IGeneratedImagesResponse.ts
services
DalleImageGeneratorService.ts
Constants.ts M
DalleImageGeneratorWebPart.manifes...
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
teams
temp
OUTLINE
IGeneratedImagesResponse.ts
src > webparts > dalleImageGenerator > models > IGeneratedImagesResponse.ts > ...
1 import { IGeneratedImageInfo } from "./IGeneratedImageInfo";
2
3
4 export interface IGeneratedImagesResponse {
5 created: number;
6 data: IGeneratedImageInfo[];
7 }
8
export interface IGeneratedImageInfo {
b64_json: string;
}
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 1, Col 1
Spaces: 4
UTF-8
LF
TypeScript
Prettier


DalleImageGeneratorService.ts — react-dall-e-image-generation
EXPLORER
OPEN EDITORS
DalleImageGeneratorService.ts src/web...
REACT-DALL-E-IMAGE-GENERATION
src
webparts / dalleImageGenerator
assets
components
DalleImageGenerator.module.scss
DalleImageGenerator.module.scss.ts
DalleImageGenerator.tsx
IDalleImageGeneratorProps.ts
IDalleImageGeneratorState.ts
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest.ts M
IGeneratedImagesResponse.ts
services
DalleImageGeneratorService.ts
Constants.ts M
DalleImageGeneratorWebPart.manifes...
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
teams
temp
OUTLINE
DalleImageGeneratorService.ts
src > webparts > dalleImageGenerator > services > DalleImageGeneratorService.ts > ...
1 import GeneratedImagesRequest, { IGeneratedImagesRequest, imageGenerationAllo
2 import { IGeneratedImagesResponse } from "../models/IGeneratedImagesResponse"
3 import Constants from '../Constants';
4
5 import { ServiceKey, ServiceScope } from "@microsoft/sp-core-library";
6 import { HttpClient, IHttpClientOptions, HttpClientResponse } from "@microsof
7
8 import { getSP } from "../pnpjsConfig";
9 import { SPFI } from "@pnp/sp";
10 import "@pnp/sp/lists";
11 import "@pnp/sp/folders/list";
12 import "@pnp/sp/files/folder";
13
14 export interface IDalleImageGeneratorService {
15 generateImages(
16 imageDescription: string,
17 numberOfImagesToGenerate: number,
18 imageSize: imageGenerationAllowedSizes): Promise<IGeneratedImagesResp
19
20 saveImageToSiteAssetsLibrary(imageName: string, image: Blob): Promise<str
21 }
22
23 export default class DalleImageGeneratorService implements IDalleImageGenerat
24
25 private _httpClient: HttpClient;
26 private _sp: SPFI;
27
28 constructor(serviceScope:ServiceScope) {
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 12, Col 31
Spaces: 4
UTF-8
LF
TypeScript
Prettier


DalleImageGeneratorService.ts — react-dall-e-image-generation
EXPLORER
OPEN EDITORS
DalleImageGeneratorService.ts sr... M
REACT-DALL-E-IMAGE-GENERATION
src
webparts/dalleImageGenerator
assets
components
DalleImageGenerator.module.scss
DalleImageGenerator.module.scss.ts
DalleImageGenerator.tsx
IDalleImageGeneratorProps.ts
IDalleImageGeneratorState.ts
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest.ts M
IGeneratedImagesResponse.ts
services
DalleImageGeneratorService.ts M
Constants.ts M
DalleImageGeneratorWebPart.manifes...
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
teams
temp
OUTLINE
DalleImageGeneratorService.ts M
webparts > dalleImageGenerator > services > DalleImageGeneratorService.ts > IDalleImageGeneratorService
21 }
22
23 export default class DalleImageGeneratorService
24 implements IDalleImageGeneratorService {
25
26 private _httpClient: HttpClient;
27 private _sp: SPFI;
28
29 constructor(serviceScope:ServiceScope) {
30 serviceScope.whenFinished(async () => {
31 this._httpClient = serviceScope.consume(HttpClient.serviceKey)
32 });
33 this._sp = getSP();
34 }
35
36 public async saveImageToSiteAssetsLibrary(imageName: string, image: Blob)
37 const imageFileToSharePoint =
38 await this._sp.web.lists
39 .getByTitle("Documents")
40 .rootFolder
41 .files
42 .addChunked(imageName, image, undefined, true);
43 const item = await imageFileToSharePoint.file
44 .getItem<{ Id: number, Title: string, FileRef: string }>
45 ("Id", "Title", "FileRef");
46 return item.FileRef;
47 }
48
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 21, Col 2
Spaces: 4
UTF-8
LF
TypeScript
Prettier

DalleImageGeneratorService.ts — react-dall-e-image-generation

EXPLORER

OPEN EDITORS
- DalleImageGeneratorService.ts sr... M

REACT-DALL-E-IMAGE-GENERATION
- src
  - webparts/dalleImageGenerator
    - assets
    - components
      - DalleImageGenerator.module.scss
      - DalleImageGenerator.module.scss.ts
      - DalleImageGenerator.tsx
      - IDalleImageGeneratorProps.ts
      - IDalleImageGeneratorState.ts
    - loc
    - models
      - IGeneratedImageInfo.ts
      - IGeneratedImagesRequest.ts M
      - IGeneratedImagesResponse.ts
    - services
      - DalleImageGeneratorService.ts M
    - Constants.ts M
    - DalleImageGeneratorWebPart.manifes...
    - DalleImageGeneratorWebPart.ts
    - pnpjsConfig.ts
  - index.ts
- teams
- temp

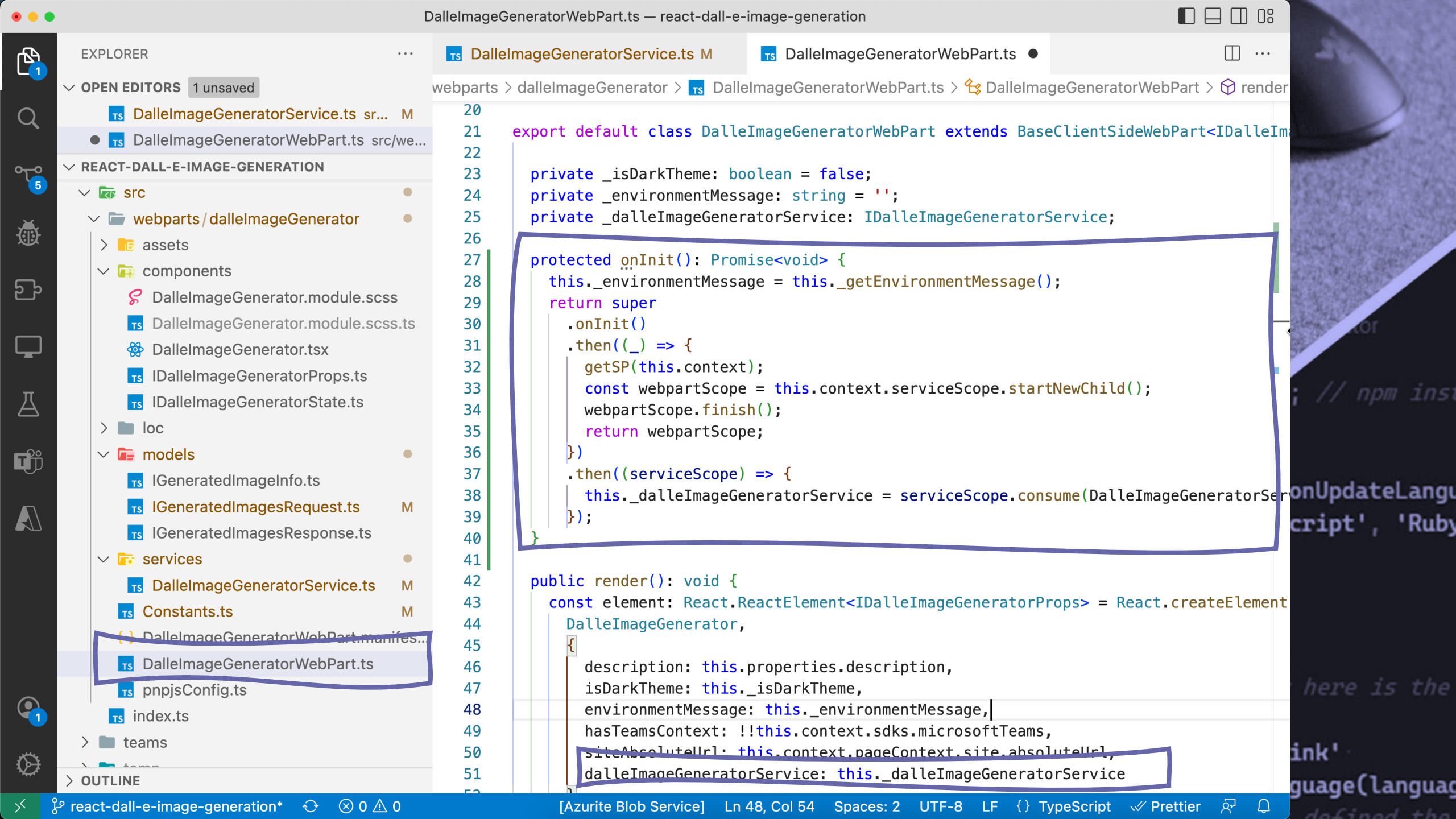
OUTLINE

DalleImageGeneratorService.ts M

webparts > dalleImageGenerator > services > DalleImageGeneratorService.ts > DalleImageGeneratorService

```
49      public async generateImages(
50          imageDescription: string,
51          numberOfImagesToGenerate: number,
52          imageSize: imageGenerationAllowedSizes):
53          Promise<IGeneratedImagesResponse> {
54
55          const generateImagesRequest: IGeneratedImagesRequest =
56              new GeneratedImagesRequest(
57                  imageDescription, numberOfImagesToGenerate, imageSize);
58
59          const requestHeaders: Headers = new Headers();
60          requestHeaders.append('Content-type', 'application/json');
61          requestHeaders.append('Authorization', `Bearer ${Constants.DalleApiKey}`);
62
63          const httpClientOptions: IHttpClientOptions = {
64              body: generateImagesRequest.toJson(),
65              headers: requestHeaders
66          };
67
68          const response: HttpClientResponse =
69              await this._httpClient.post(
70                  "https://api.openai.com/v1/images/generations",
71                  HttpClient.configurations.v1,
72                  httpClientOptions);
73
74          const generateImagesResponse: IGeneratedImagesResponse =
75              await response.json();
76
77          return generateImagesResponse;
78      }
79  }
80
```

react-dall-e-image-generation* | 0 ⚠ 0 | [Azurite Blob Service] | Ln 79, Col 2 | Spaces: 4 | UTF-8 | LF | TypeScript | Prettier


DalleImageGeneratorWebPart.ts — react-dall-e-image-generation
EXPLORER
OPEN EDITORS 1 unsaved
DalleImageGeneratorService.ts sr... M
DalleImageGeneratorWebPart.ts src/we...
REACT-DALL-E-IMAGE-GENERATION
src
webparts / dalleImageGenerator
assets
components
DalleImageGenerator.module.scss
DalleImageGenerator.module.scss.ts
DalleImageGenerator.tsx
IDalleImageGeneratorProps.ts
IDalleImageGeneratorState.ts
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest.ts M
IGeneratedImagesResponse.ts
services
DalleImageGeneratorService.ts M
Constants.ts M
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
teams
OUTLINE
DalleImageGeneratorService.ts M
DalleImageGeneratorWebPart.ts
webparts > dalleImageGenerator > DalleImageGeneratorWebPart.ts > DalleImageGeneratorWebPart > render
20
21 export default class DalleImageGeneratorWebPart extends BaseClientSideWebPart<IDalleIma
22
23 private _isDarkTheme: boolean = false;
24 private _environmentMessage: string = '';
25 private _dalleImageGeneratorService: IDalleImageGeneratorService;
26
27 protected onInit(): Promise<void> {
28 this._environmentMessage = this._getEnvironmentMessage();
29 return super
30 .onInit()
31 .then((_) => {
32 getSP(this.context);
33 const webpartScope = this.context.serviceScope.startNewChild();
34 webpartScope.finish();
35 return webpartScope;
36 })
37 .then((serviceScope) => {
38 this._dalleImageGeneratorService = serviceScope.consume(DalleImageGeneratorSer
39 });
40 }
41
42 public render(): void {
43 const element: React.ReactElement<IDalleImageGeneratorProps> = React.createElement
44 DalleImageGenerator,
45 {
46 description: this.properties.description,
47 isDarkTheme: this._isDarkTheme,
48 environmentMessage: this._environmentMessage,
49 hasTeamsContext: !!this.context.sdks.microsoftTeams,
50 siteAbsoluteUrl: this.context.pageContext.site.absoluteUrl,
51 dalleImageGeneratorService: this._dalleImageGeneratorService
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 48, Col 54
Spaces: 2
UTF-8
LF
TypeScript
Prettier

DalleImageGenerator.tsx — react-dall-e-image-generation

EXPLORER

OPEN EDITORS 1 unsaved

- DalleImageGeneratorService.ts sr... M
- DalleImageGeneratorWebPart.ts src/we...
- DalleImageGenerator.tsx src/webparts/d...

REACT-DALL-E-IMAGE-GENERATI...

- src
  - webparts / dalleImageGenerator
    - assets
    - components
      - DalleImageGenerator.module.scss
      - DalleImageGenerator.module.scss.ts
      - DalleImageGenerator.tsx
      - IDalleImageGeneratorProps.ts
      - IDalleImageGeneratorState.ts
    - loc
    - models
      - IGeneratedImageInfo.ts
      - IGeneratedImagesRequest.ts M
      - IGeneratedImagesResponse.ts
    - services
      - DalleImageGeneratorService.ts M
    - Constants.ts M
    - DalleImageGeneratorWebPart.manifes...
    - DalleImageGeneratorWebPart.ts
    - pnpjsConfig.ts
  - index.ts

OUTLINE

DalleImageGeneratorService.ts M | DalleImageGeneratorWebPart.ts | DalleImageGenerator.tsx

src > webparts > dalleImageGenerator > components > DalleImageGenerator.tsx > ...

```
68      <Stack>
69        <TextField
70          label="Image description (be creative!)"
71          multiline
72          rows={3}
73
74          text="Generate Images with DALL-E"
75          onClick={this._sendImagesGenerationRequest} />
76
77        <Stack tokens={{ padding: 10, childrenGap: 10 }}>
78          <MessageBar>{this.state.actionMessageInfo}</MessageBar>
79          {imageSavedInfo}
80          {images}
81        </Stack>
82      </Stack>
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Image description (be creative!)

A green dog eating a big plate of pasta

Number of images to generate? 2

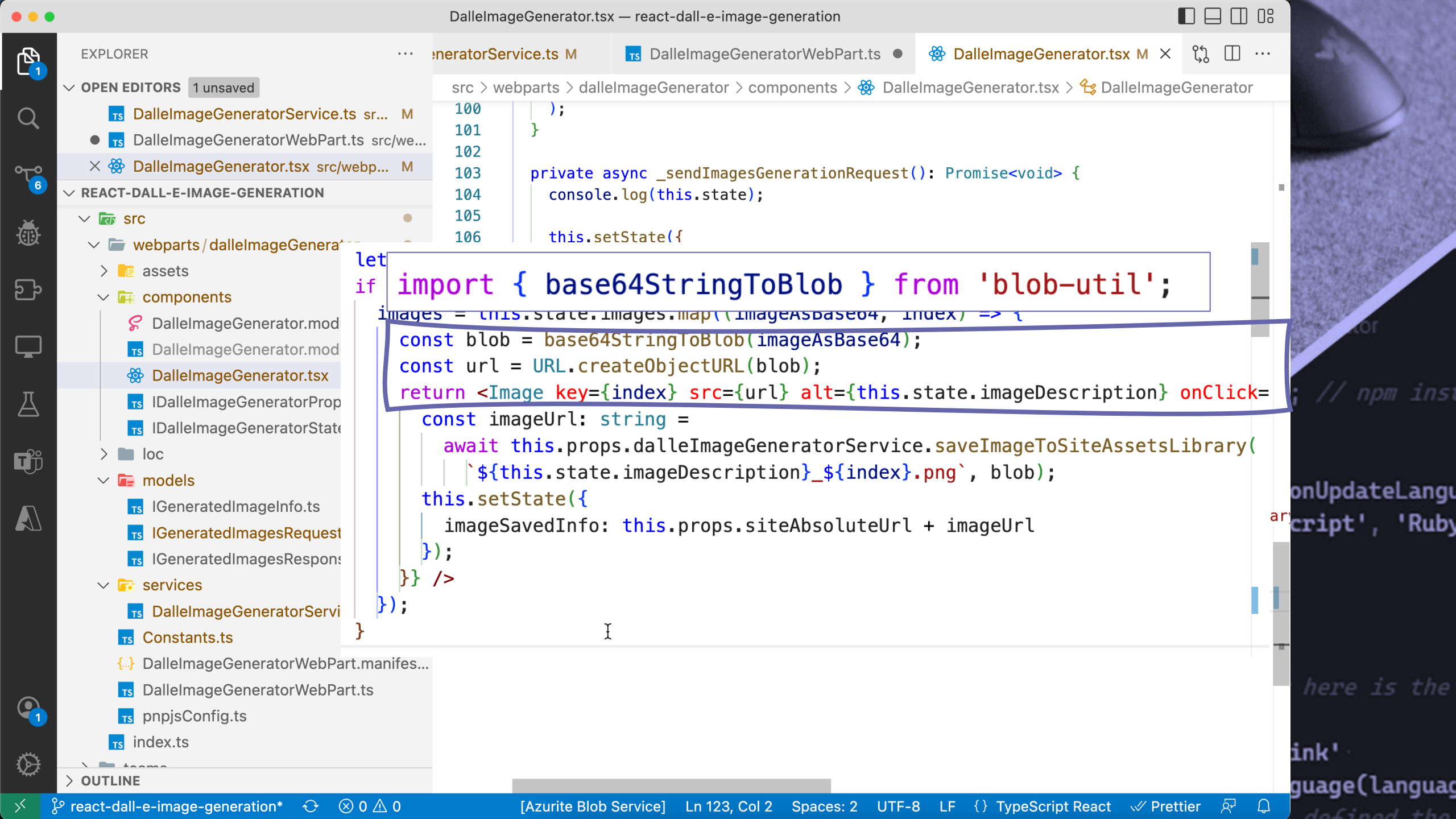
Size

- (•) 256x256
- ( ) 512x512
- ( ) 1024x1024

Generate Images with DALL-E

Click on the image to save it into the site documents library

react-dall-e-image-generation* | 0 ⚠ 0 | [Azurite Blob Service] | Ln 16, Col 26 | Spaces: 2 | UTF-8 | LF | TypeScript React | Prettier


DalleImageGenerator.tsx — react-dall-e-image-generation
EXPLORER
OPEN EDITORS 1 unsaved
DalleImageGeneratorService.ts sr... M
DalleImageGeneratorWebPart.ts src/we...
DalleImageGenerator.tsx src/webp... M
REACT-DALL-E-IMAGE-GENERATION
src
webparts / dalleImageGenera
assets
components
DalleImageGenerator.mod
DalleImageGenerator.mod
DalleImageGenerator.tsx
IDalleImageGeneratorProp
IDalleImageGeneratorState
loc
models
IGeneratedImageInfo.ts
IGeneratedImagesRequest
IGeneratedImagesRespons
services
DalleImageGeneratorServi
Constants.ts
DalleImageGeneratorWebPart.manifes...
DalleImageGeneratorWebPart.ts
pnpjsConfig.ts
index.ts
OUTLINE
DalleImageGeneratorService.ts M
DalleImageGeneratorWebPart.ts
DalleImageGenerator.tsx M
src > webparts > dalleImageGenerator > components > DalleImageGenerator.tsx > DalleImageGenerator
100);
101 }
102
103 private async _sendImagesGenerationRequest(): Promise<void> {
104 console.log(this.state);
105
106 this.setState({
let
if
import { base64StringToBlob } from 'blob-util';
images = this.state.images.map((imageAsBase64, index) => {
const blob = base64StringToBlob(imageAsBase64);
const url = URL.createObjectURL(blob);
return <Image key={index} src={url} alt={this.state.imageDescription} onClick=
const imageUrl: string =
await this.props.dalleImageGeneratorService.saveImageToSiteAssetsLibrary(
`${this.state.imageDescription}_${index}.png`, blob);
this.setState({
imageSavedInfo: this.props.siteAbsoluteUrl + imageUrl
});
}} />
});
}
react-dall-e-image-generation*
0 0
[Azurite Blob Service]
Ln 123, Col 2
Spaces: 2
UTF-8
LF
TypeScript React
Prettier

https://github.com/pnp/sp-dev-fx-webparts/tree/main/samples/react-dall-e-image-generation

Thanks!

#sharingiscaring

Microsoft® Most Valuable Professional

@luismanez