

해결하기: Icona

어떻게하면 피그마에 존재하는 아이콘을
빠르게 개발단으로 옮길 수 있을까

버튼 클릭 한번으로 모든 것이 동기화

svg로 다양한 포맷의 아이콘 생성하기

당근에서만 사용되지 않고 아이콘을 관리하고 싶다면 누구나 사용 가능



“아~ 그러면 SVG 하나마다 uploadBlob(uploadFile) API로
깃허브에 보내면 되겠구나~”

Blob = File

```
1  async function uploadBlob({  
2    content: string,  
3    encoding: "utf-8" | "base64" = "utf-8"  
4  }): Promise<{ sha: string; model: string; type: string }>  
5    return fetch(`${API_URL}/git/blobs`, {  
6      method: "POST",  
7      headers: {  
8        Authorization: `token ${ACCESS_TOKEN}`,  
9        "X-GitHub-API-Version": GITHUB_API_VERSION,  
10     },  
11     body: JSON.stringify({  
12       content,  
13       encoding,  
14     }),  
15   }).then((res) => res.json());  
16 }
```

해결하기: Icona

어떻게하면 피그마에 존재하는 아이콘을
빠르게 개발단으로 옮길 수 있을까



버튼 클릭 한번으로 모든 것이 동기화

svg로 다양한 포맷의 아이콘 생성하기

당근에서만 사용되지 않고 아이콘을 관리하고 싶다면 누구나 사용 가능

“아~ 그러면 SVG 하나마다 uploadBlob(uploadFile) API로
깃허브에 보내면 되겠구나~”

Blob = File

```
1  async function uploadBlob(  
2    content: string,  
3    encoding: 'utf-8' | 'base64' = 'utf-8',  
4  ): Promise<{ sha: string; url: string; type: string }> {  
5    return fetch(`${API_URL}/git/blobs`, {  
6      method: "POST",  
7      headers: {  
8        Authorization: `token ${ACCESS_TOKEN}`,  
9        "X-GitHub-API-Version": GITHUB_API_VERSION,  
10     },  
11     body: JSON.stringify({  
12       content,  
13       encoding,  
14     }),  
15   }).then((res) => res.json());  
16 }
```