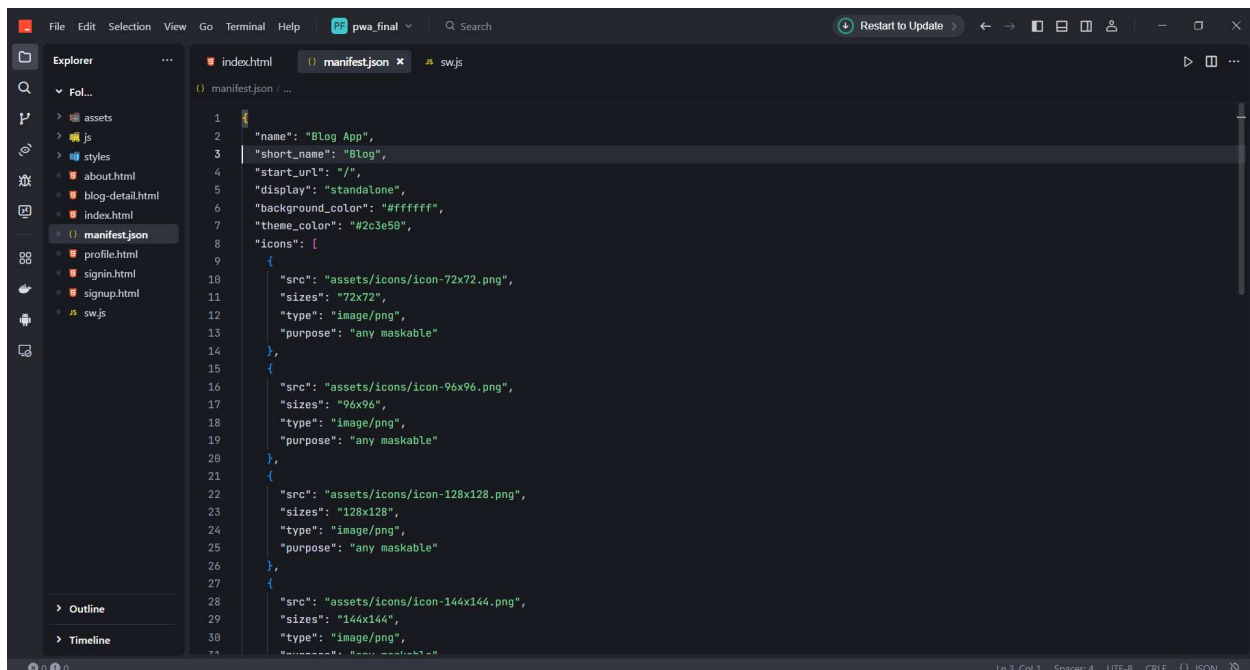


## Steps to Make App a PWA

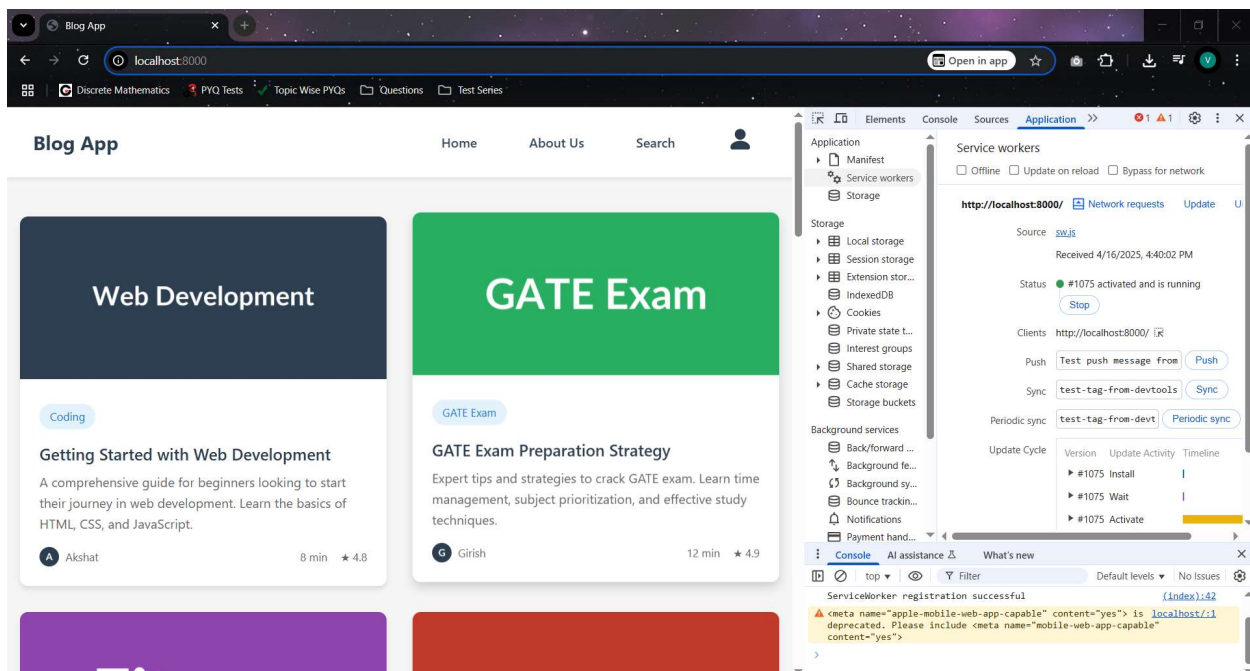
### 1. Add manifest.json File

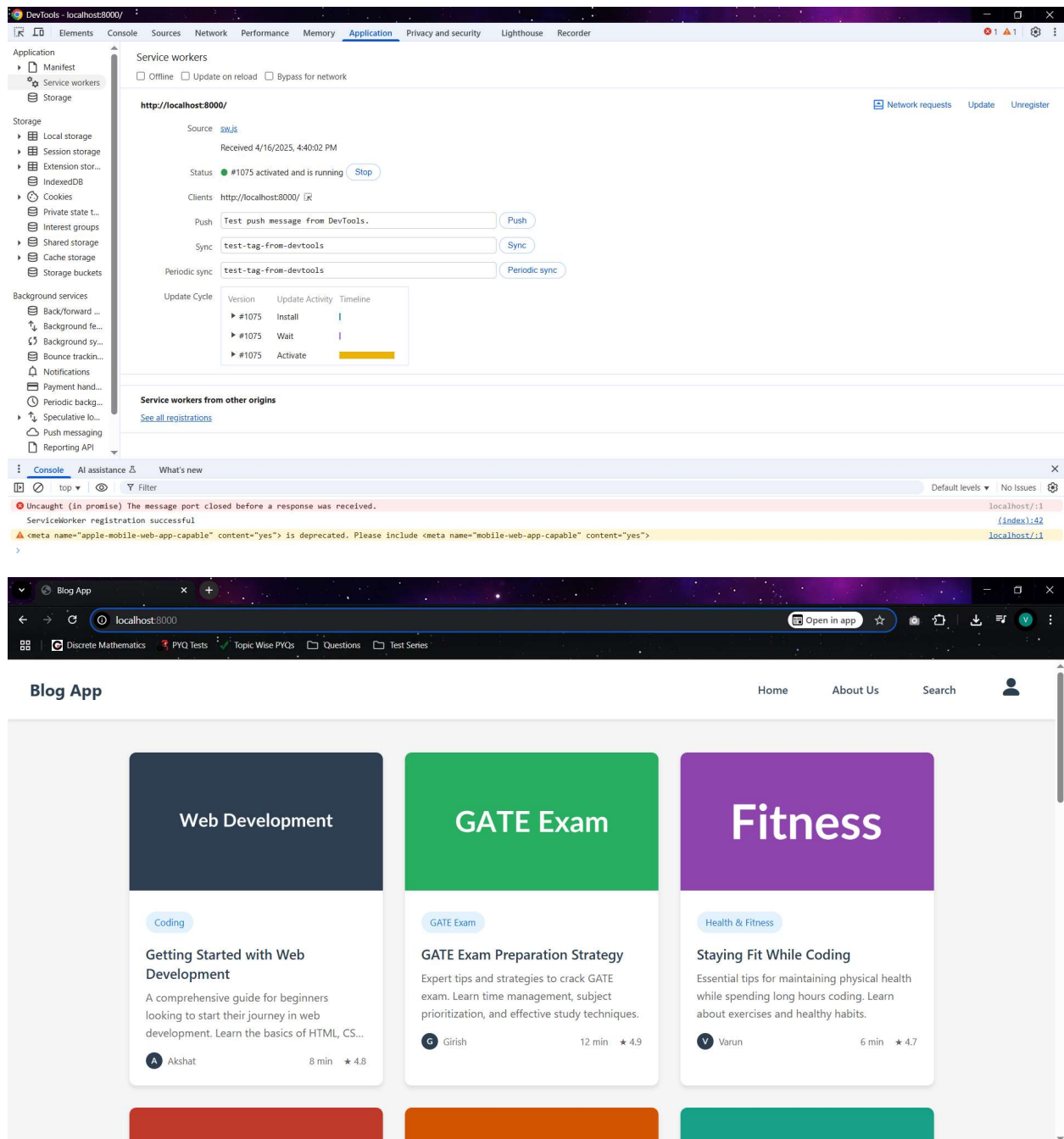
1. Go to the public folder in your project.
2. Create a new file named manifest.json.
3. Add details like the app name, icons, theme color, and display mode.
4. This file helps browsers recognize your app as installable.

### OUTPUT :



```
1 {
2   "name": "Blog App",
3   "short_name": "Blog",
4   "start_url": "/",
5   "display": "standalone",
6   "background_color": "#ffffff",
7   "theme_color": "#2c3e50",
8   "icons": [
9     {
10      "src": "assets/icons/icon-72x72.png",
11      "sizes": "72x72",
12      "type": "image/png",
13      "purpose": "any maskable"
14    },
15    {
16      "src": "assets/icons/icon-96x96.png",
17      "sizes": "96x96",
18      "type": "image/png",
19      "purpose": "any maskable"
20    },
21    {
22      "src": "assets/icons/icon-128x128.png",
23      "sizes": "128x128",
24      "type": "image/png",
25      "purpose": "any maskable"
26    },
27    {
28      "src": "assets/icons/icon-144x144.png",
29      "sizes": "144x144",
30      "type": "image/png",
31      "purpose": "any maskable"
32    }
33  ]
34 }
```





## Conclusion:

By following these steps, we successfully converted our Vite React application into a Progressive Web App. This enables offline accessibility, caching for better performance, and an installable app experience on mobile and desktop devices. Implementing PWA features enhances user engagement and makes the app function seamlessly even in limited network conditions.