

Designmönster för SkyVibe: Observer Pattern

<https://github.com/Ayub-Tech/skyVibe.git>

I SkyVibe bygger vi en väderapplikation som hämtar och visar aktuell väderinformation för användarens plats. Eftersom väderdata ständigt förändras vill vi att användargränssnittet (UI) uppdateras automatiskt utan att användaren behöver ladda om sidan manuellt.

För att uppnå detta använder vi Observer Pattern, ett designmönster som låter flera komponenter lyssna på och reagera på förändringar i väderdata i realtid. Det gör applikationen mer flexibel, modulär och lättare att underhålla.

Varför Observer Pattern?

Observer Pattern är ett mönster där en subject (det som observeras) håller en lista över observers (de som lyssnar på förändringar) och meddelar dem när något uppdateras.

Hur hjälper det SkyVibe?

- Automatiska uppdateringar – UI får direkt den senaste väderinformationen.
- Lös koppling – UI och väderlogik är separerade, vilket gör koden lättare att ändra och förbättra.-
- Enkel skalbarhet – Vi kan enkelt lägga till fler observers, t.ex. en modul för vädervarningar.

Use Case: Uppdatering av väderdata i UI

Pre-villkor:

- Användaren har öppnat appen och ser aktuell väderinformation.

Huvudflöde:

1. SkyVibe hämtar väderdata från en API-tjänst.
2. Observer Pattern notifierar alla registrerade observers (t.ex. UI-komponenter).

3. UI uppdateras automatiskt med den nya temperaturen och väderförhållandena.
4. Användaren ser den senaste väderinformationen utan att behöva interagera.

Post-villkor:

- UI visar alltid den senaste väderinformationen i realtid.