

# Sharing our Angular experiences ... and run-ins with certain caveats 🤷‍♀️ 💰

© [@michahell](#) @ [ZIVVER](#)



# overview

- overall opinion
- good parts
- bad parts
- summary
- other
- future Angular



© [@michahell](#) @ [ZIVVER](#)

# opinion

© [@michahell](#) @ [ZIVVER](#)

**pro:** great to work with, a bit  
verbose, conceptually very much  
OO with some FRP and  
immutability mixed in.

– **Michael**

• [@michahell](#) • [ZIVVER](#)

**con:** some parts under documented. Offers a lot, so also big and complex : long learning curve, not steep!

– **Michael**

• [@michaelhell](#) @ [ZIVVER](#)

dynamism gets little attention:  
generated components /  
structural directives, delegated  
forms, etc.

**pro:** full package framework, it has everything you need. Typescript and rxjs makes it more powerful.

– **Sarea**

**con:** sometimes it becomes too complex. the syntax makes me feel that it's almost a separate language. RXJS is very powerful, but be careful!

– **Sarea**

# good parts

- click ↗

# bad parts

- click ↗

# caveat summary

- **\*ngFor**
- **| async + \*ngLet**
- **custom form components**
- **view encapsulation**
- **RXJS operators, error handling**
- **state mgmt best practices**

# other

- **@Input getter setter**
- **component defaults**
- **moar articles**

# component defaults

```
const defaultComponentOptions = {  
  viewEncapsulation: ViewEncapsulation.ShadowDom,  
  changeStrategy: ChangeDetectionStrategy.OnPush  
  // more component defaults  
};  
  
@Component({  
  selector: 'owa-validation-details',  
  templateUrl: './validation-details.component.html',  
  styleUrls: ['./validation-details.component.scss'],  
  animations: [createWarningAnim('checkSetupAnim', '600ms 1000ms ease-in-out')],  
  ... defaultComponentOptions  
})
```

©2@michahell @ ZIVVER

# @Input setter / getter

```
@Input()
get value() : IPostPhonenumberValidateRequest | null {
    return this.phoneNumberInput.value;
}
set value (value: IPostPhonenumberValidateRequest | null) {
    value = value || { phoneNumber: '0', defaultRegion: 'NL' };
    this.phoneNumberInput.setValue(value.phoneNumber);
    this.stateChanges.next();
}
```

03@michahell @ ZIVVER

# moar articles

- [Create and Test Decorators in JavaScript - Netanel Basal](#)
- [Provider scope](#)

 [michahell](#) @ [ZIVVER](#)

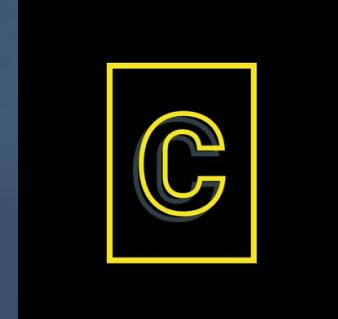
# future Angular

- **future: Angular Ivy improvements**
  - better treeshaking
- **metaprogramming, mixins, HOC ([article](#))**

# thank you

<https://github.com/michahell/ks-angular>

made with Deckset (Markdown), Carbon, StackBlitz and Figma



©[@michahell](#) @ [ZIVVER](#)