

Unifying form validation with TCA

Anthony CHANLIAU MERLE

iOS lead dev
aviv group_

SeLogger

immoweb

immonet.de

immowelt

aviv group_

Summary

- The pain of validating forms
- Unified form validation
- Under the hood

Forms

Étape 8 sur 8

✕

1 Votre adresse email n'apparaît pas sur l'annonce

Nous utiliserons uniquement votre adresse email pour vous envoyer des demandes de contact.

Nom d'utilisateur *

Choisissez un nom ou un pseudo à afficher sur votre annonce.

Nom d'utilisateur

Numéro de téléphone (Optionnel)

Facilitez vos échanges avec vos futurs locataires.

Je ne souhaite pas que mes données soient transmises à des agences partenaires de SLG pour m'accorder dans la vente de mon bien

Je ne souhaite pas recevoir des communications de la part des membres du Groupe AVIV

[Preview privacy policy](#)

← Retour

Modifier mon annonce

Calculez votre loyer

Loyer de base (hors charges) *

Il ne doit pas dépasser le montant de la référence majeure.

1 000

Le loyer ne peut pas dépasser

Charges *

ex: 200

Saisissez un prix pour continuer

Complément de loyer (Optionnel)

ex: 300

1 000 € CC/m

Loyer total affiché sur la page

Précisez le dépôt de garantie souhaité

Dépôt de garantie *

Enregistrer

Modifier mon annonce

Résultats du DPE

DPE vierge : mon DPE n'indique pas de classement énergétique.

Énergie (Optionnel)

100

A B C D

Logement économe

GES (Optionnel)

200

A B C D

Peu d'émissions

Date de réalisation du DPE (Optionnel)

☒ Avant le 1 juillet 2021

☐ Entre le 1 juillet 2021 et le 31 décembre 2023

☐ À partir du 1 janvier 2023

[Preview terms and conditions \(FRBO\)](#)

← Back

Step 6 of 8

✕

Let's move on to the description of your flat?

* Mandatory informations

Description of the property *

Highlight its strengths using at least 100 characters.

Your text

Enter a description with more to continue.

Generate with AI

Step 7 of 8

✕

At what price are you renting your flat?

* Mandatory informations

Calculate your rent

Basic rent *

Without additional costs

ex: 1000

Additional costs *

ex: 200

☒ Heating costs included

Total rent: _ €/month

Add other costs

Security deposit *

3 months of basic rent maximum

← Back

Step 5 of 8

✕

Year of the construction

According to the energy certificate

Indicate a year

Certificate valid from

DD/MM/YYYY

Valid until

DD/MM/YYYY

Primary energy requirement

Final energy demand value

Efficiency class

Choose a class

[Preview terms and conditions \(FRBO\)](#)

← Back

Modifier mon annonce

Résultats du DPE

DPE vierge : mon DPE n'indique pas de classement énergétique.

Énergie (Optionnel)

0

A B C D

Logement économe

GES (Optionnel)

0

A B C D

Peu d'émissions

Date de réalisation du DPE (Optionnel)

☒ Avant le 1 juillet 2021

☐ Entre le 1 juillet 2021 et le 31 décembre 2023

☐ À partir du 1 janvier 2023

[Preview terms and conditions \(FRBO\)](#)

← Back

Étape 4 sur 8

✕

Enregistrer

Précisez les caractéristiques de votre maison

* Information obligatoire

Informations clés

Superficie *

Les biens en copropriété doivent respecter la loi Carrez

Ex: 110 m²

Cette information ne peut pas être modifiée après publication.

Nombre de pièces *

La cuisine, la salle de bain et les toilettes ne sont pas considérées comme des pièces

- 1 +

Nombre de chambres *

Un studio n'a pas de chambres

- 0 +

← Retour

Continuer

Forms

13:48 Étape 8 sur 8

1 Votre adresse email n'apparaît pas sur l'annonce

Nous utiliserons uniquement votre adresse email pour vous envoyer des demandes de contact.

Nom d'utilisateur *

Choisissez un nom ou un pseudo à afficher

Nom d'utilisateur

Numéro de téléphone (Optionnel)

Facilitez vos échanges avec vos futurs locataires

Je ne souhaite pas recevoir des communications de la part des agences partenaires de SLG pour m'accorder dans la vente de mon bien

Je ne souhaite pas recevoir des communications de la part des agences partenaires de SLG pour m'accorder dans la vente de mon bien

Preview privacy policy

Retour Enregistrer

13:50 Modifier mon ann

Calculez votre loyer

Loyer de base (hors charges) *

Il ne doit pas dépasser le montant de la référence majeure.

1 000

Le loyer ne peut pas dépasser

Charges *

ex: 200

Saisissez un prix pour continuer

Complément de loyer (Optionnel)

ex: 300

1 000 € CC/m

Loyer total affiché sur la plateforme

Précisez le dépôt de garantie

Dépôt de garantie *

Enregistrer

13:50 Modifier mon ann

Résultats du DPE

DPE vierge : mon DPE n'indique pas de classement énergétique.

Énergie (Optionnel)

100

Logement économe

GES (Optionnel)

200

Date de réalisation du DPE (Optionnel)

Avant le 1 juillet 2021

Entre le 1 juillet 2021 et le 31 décembre 2023

À partir du 1 janvier 2023

Enregistrer

14:03 Step 6 of 8

Let's move on to the description of your flat?

* Mandatory informations

Description of the property *

Highlight its strengths using at least 100 characters

Your text

Enter a description with more than 100 characters to continue.

Generate with AI

Back

14:04 Step 7 of 8

At what price are you renting your flat?

* Mandatory informations

Calculate your rent

Basic rent *

Without additional costs

ex: 1000

Additional costs *

ex: 200

Heating costs included

Total rent: _ €/month

Add other costs

Security deposit *

3 months of basic rent maximum

Back

14:03 Step 5 of 8

Year of the construction

According to the energy certificate

Indicate a year

Certificate valid from

DD/MM/YYYY

Valid until

DD/MM/YYYY

Primary energy requirement

Final energy demand value

Efficiency class

Choose a class

Preview terms and conditions (FRBO)

Back

13:51 Modifier mon ann

Résultats du DPE

DPE vierge : mon DPE n'indique pas de classement énergétique.

Énergie (Optionnel)

0

Logement économe

GES (Optionnel)

0

Peu d'émissions

Date de réalisation du DPE (Optionnel)

Avant le 1 juillet 2021

Entre le 1 juillet 2021 et le 31 décembre 2023

À partir du 1 janvier 2023

Enregistrer

13:47 Étape 4 sur 8 Enregistrer

Précisez les caractéristiques de votre maison

* Information obligatoire

Informations clés

Superficie *

Les biens en copropriété doivent respecter la loi Carrez

Ex: 110 m²

Cette information ne peut pas être modifiée après publication.

Nombre de pièces *

La cuisine, la salle de bain et les toilettes ne sont pas considérées comme des pièces

1

Nombre de chambres *

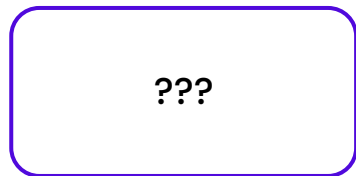
Un studio n'a pas de chambres

0

Retour Continuer

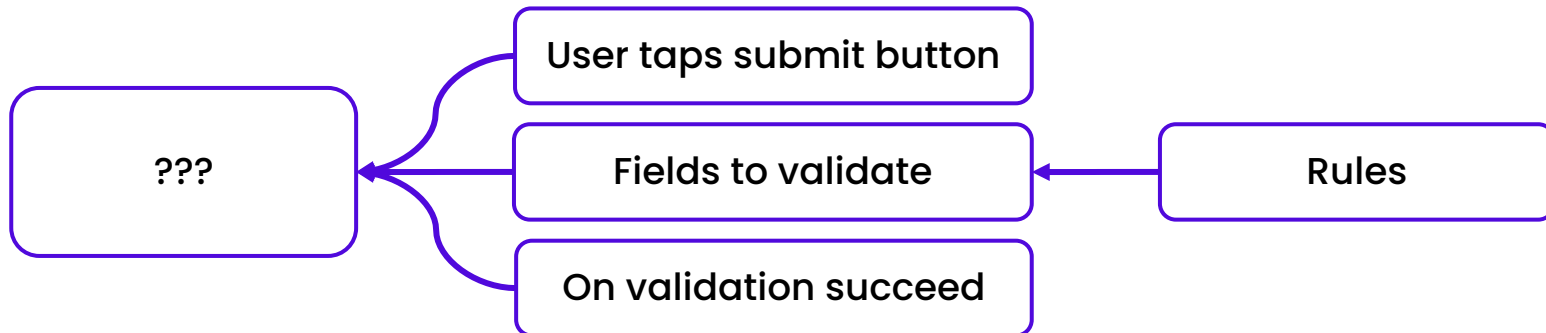
Forms

Brainstorming

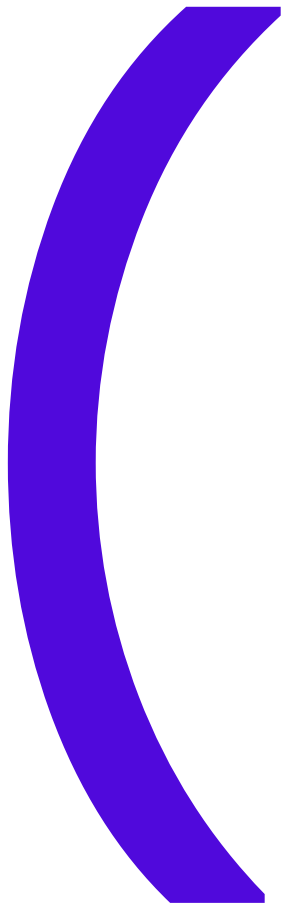


Validate all fields on form submission

Validate each field on a predefined condition
(i.e. value changes)

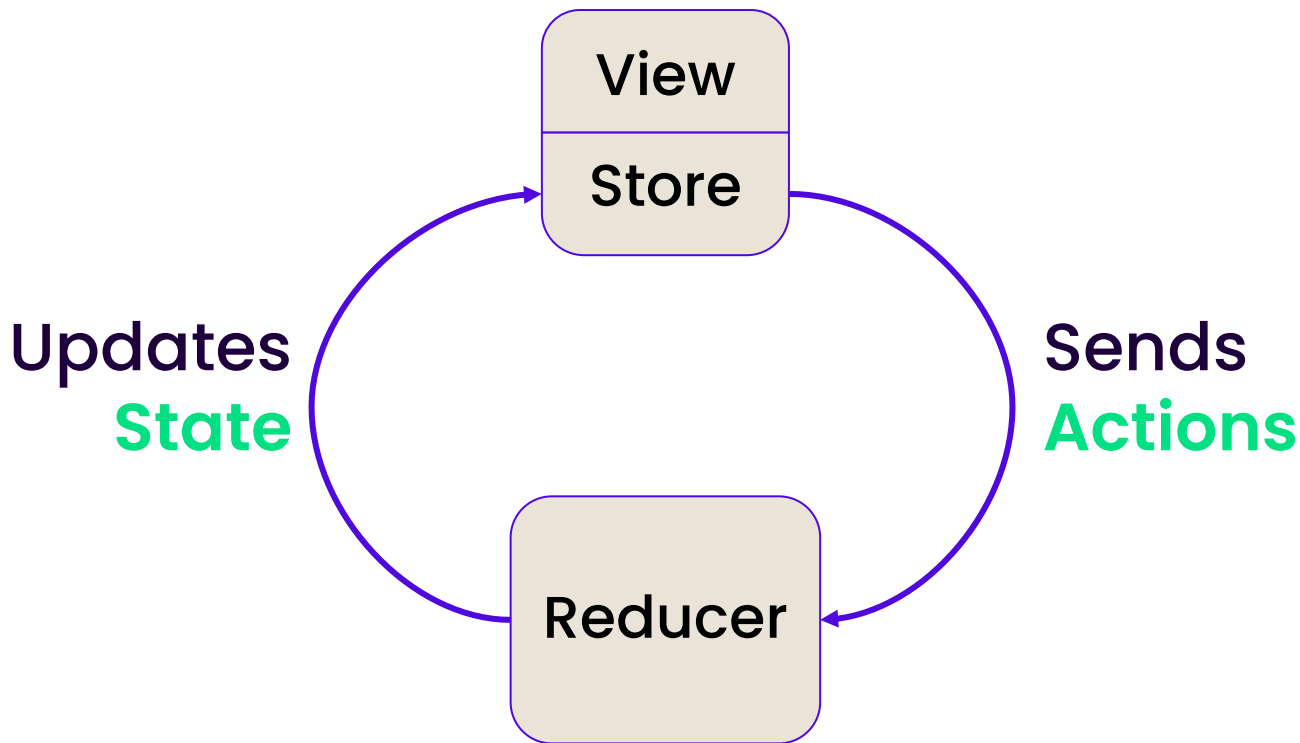


Unified form validation

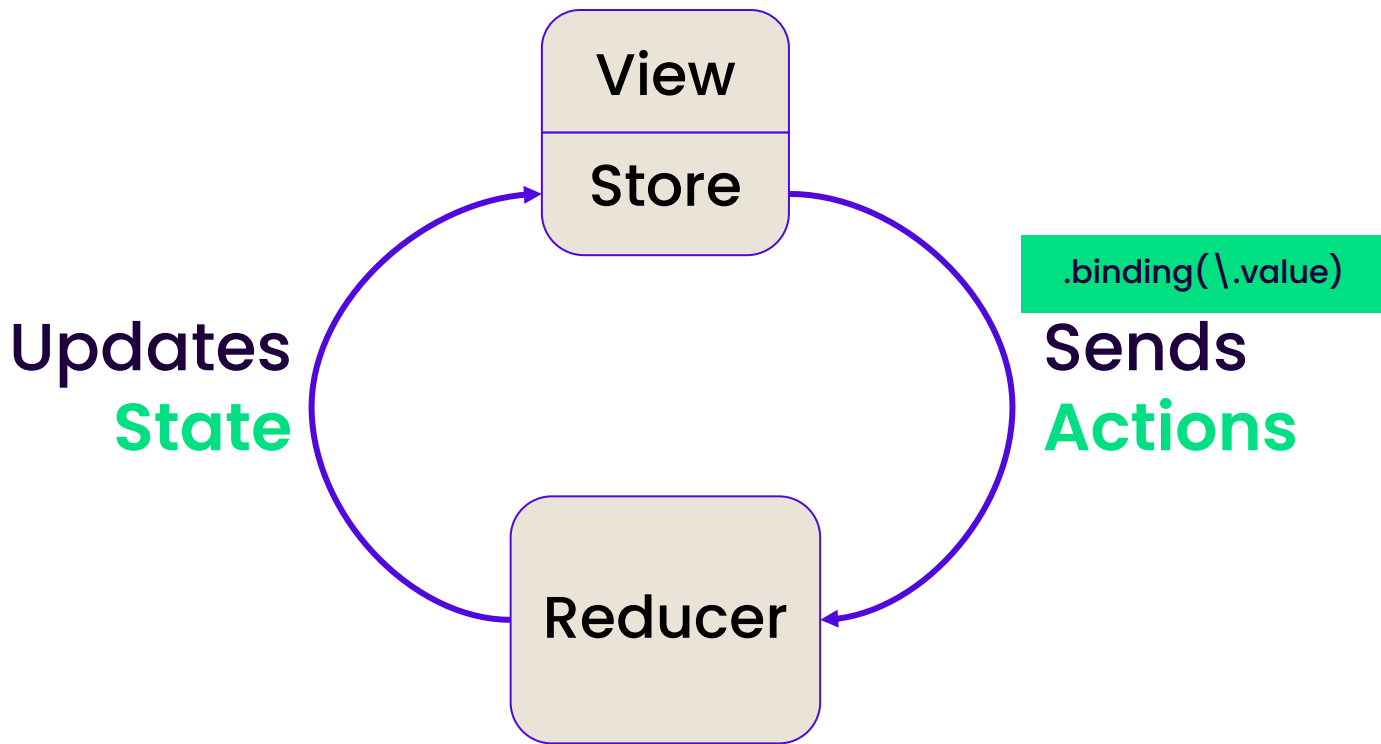


The Composable Architecture (TCA)

By Point-Free

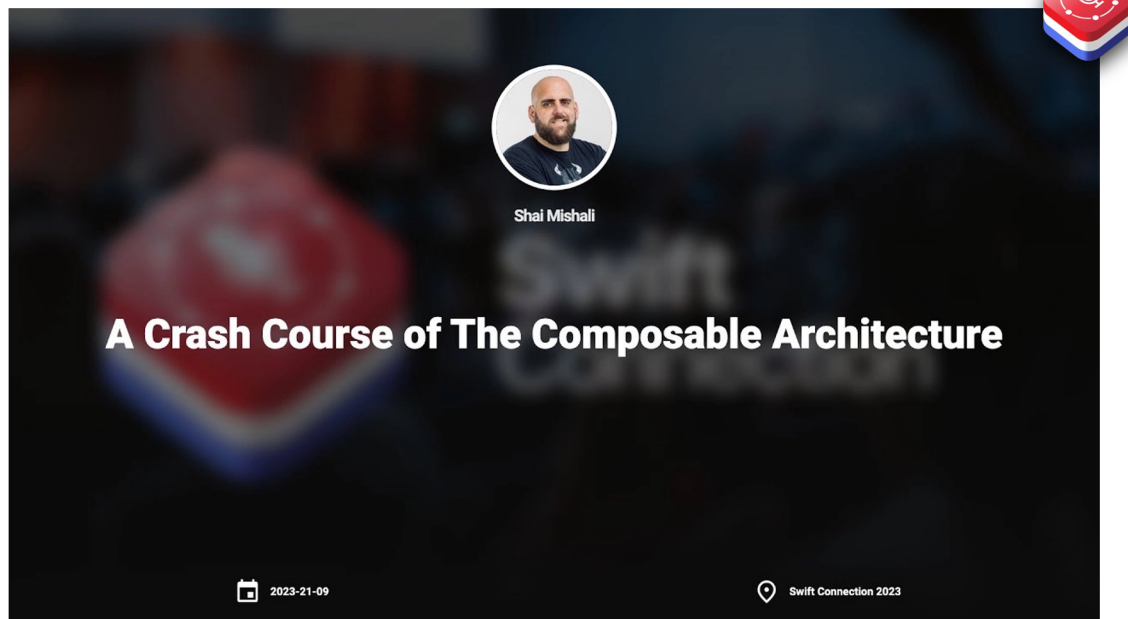


Bindings

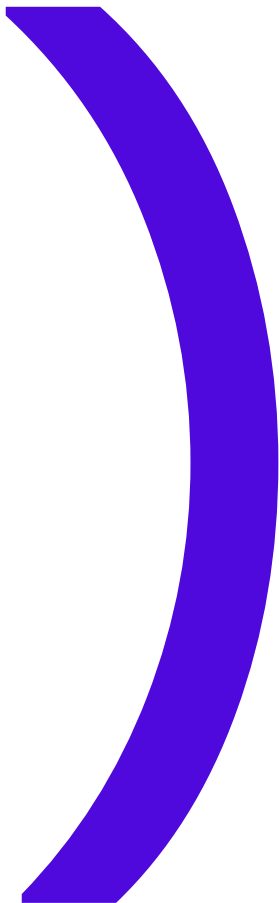


More on TCA

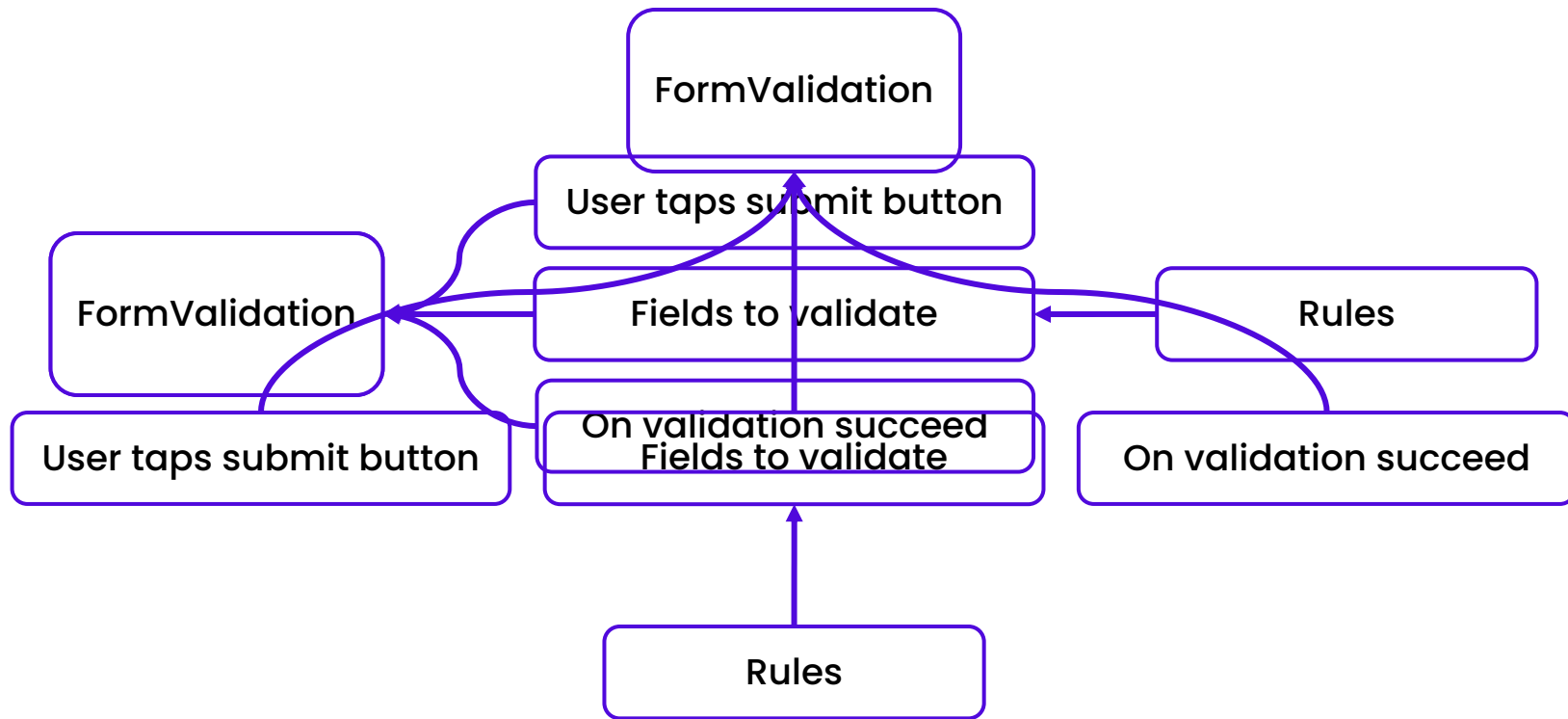
Swift Connection 2023



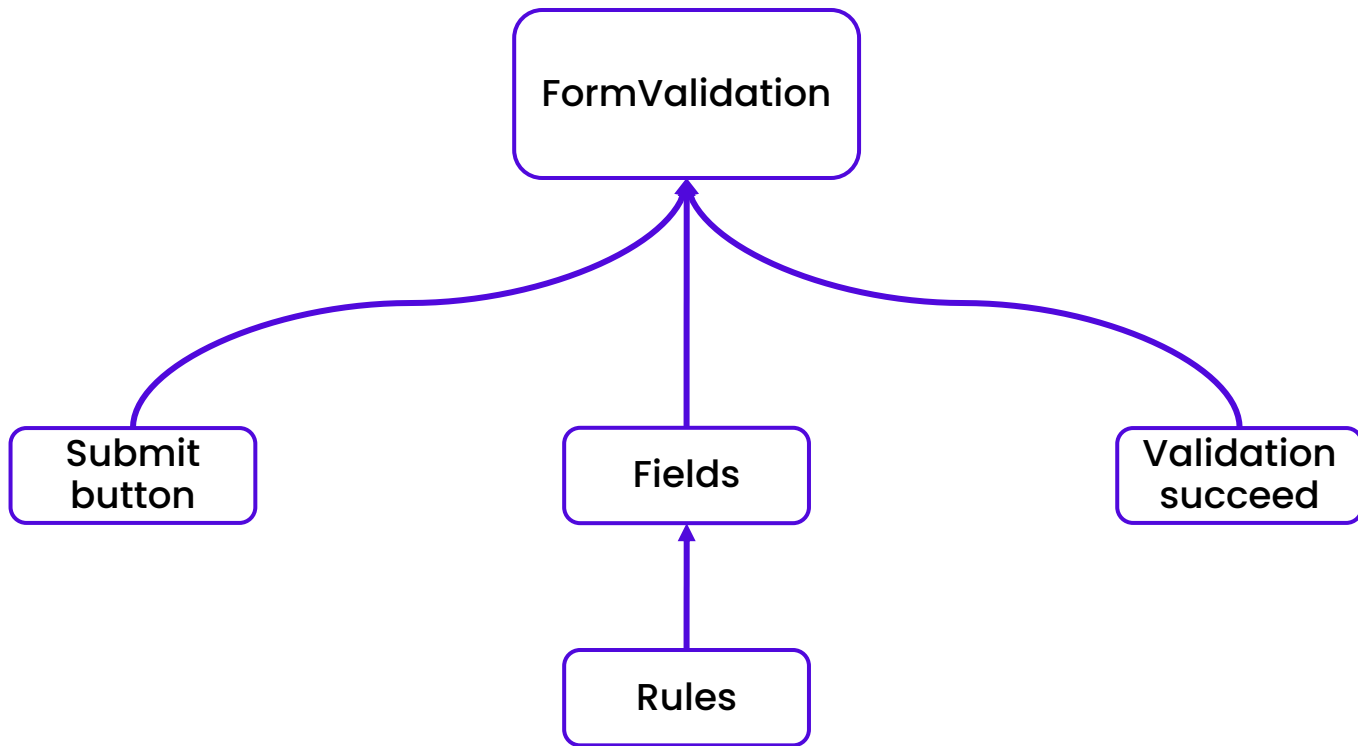
async.techconnection.io/frenchkit



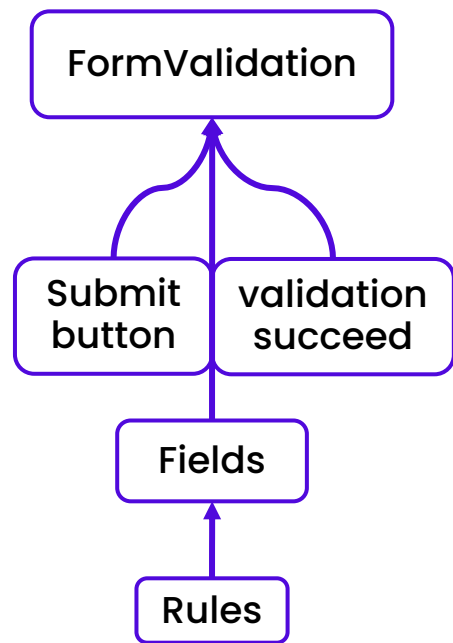
Unified form validation



Unified form validation

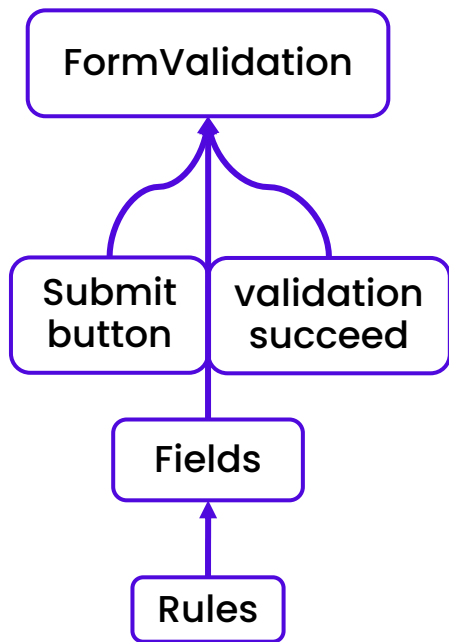


Unified form validation



Result

FormValidation



aviv group_

```
var body: some Reducer0 f<Self> {
```

```
  BindingReducer()
```

```
  Form Validation(...)
```

```
  Reduce {state, action in
```

```
    switch action {
```

```
      case .form ValidationSucceed :
```

```
        return none
```

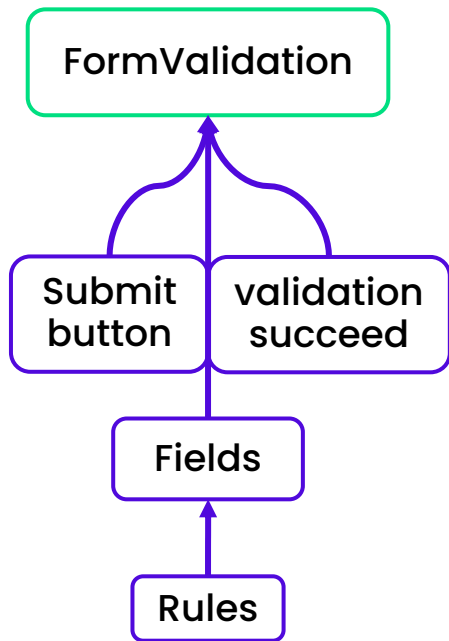
```
    }
```

```
  }
```

```
}
```


Result

FormValidation



aviv group_

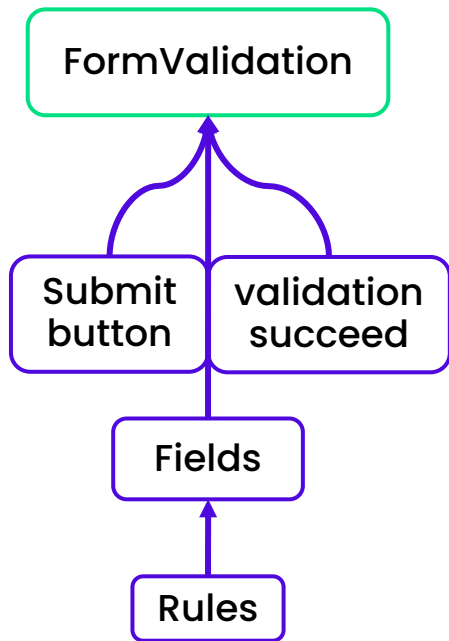
```
var body: some ReducerOf<Self> {  
  BindingReducer()  
}
```

FormValidation(...)

```
  Reduce { state, action in  
    switch action {  
      case .formValidationSucceed:  
        return .none  
    }  
  }  
}
```

Result

FormValidation



aviv group_

FormValidation(

userAction: \.validationButtonTap,

onSucceed: .formValidationSucceed,

fields: [

FieldValidation(

name: \.name,

error: \.nameError,

rules: [

.notEmpty("Username should not be empty"),

.length(min: 5, "Username is too short")

]

),

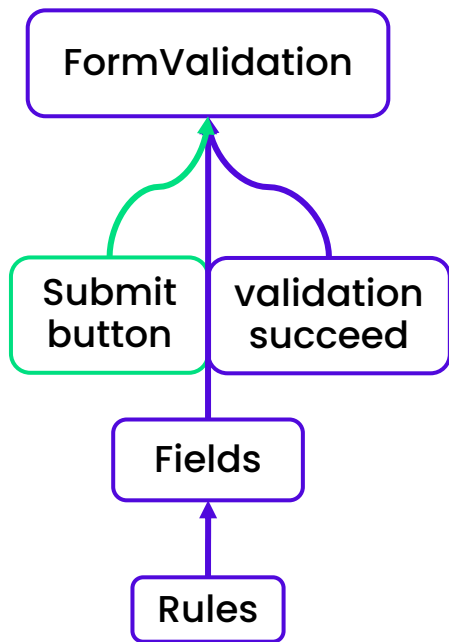
FieldValidation(...)

]

)

Result

FormValidation

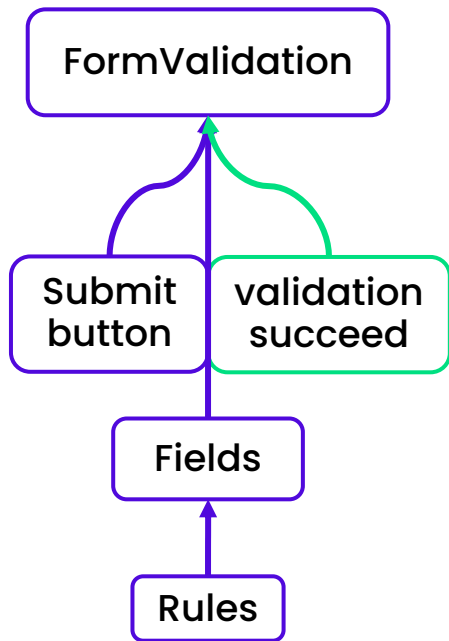


aviv group_

```
FormValidation(  
    userAction: \.validationButtonTap,  
    onSuccess: .formValidationSucceed,  
    fields: [  
        FieldValidation(  
            name: \.name,  
            error: \.nameError,  
            rules: [  
                .notEmpty("Username should not be empty"),  
                .length(min: 5, "Username is too short")  
            ]  
        ),  
        FieldValidation(...)  
    ]  
)
```

Result

FormValidation

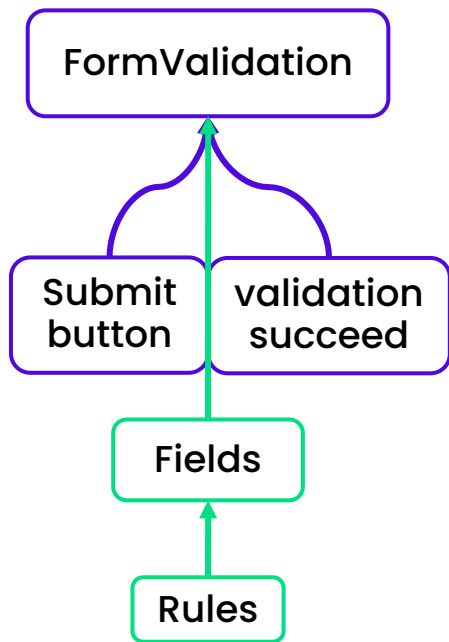


aviv group_

```
FormValidation(  
    userAction: \.validationButtonTap,  
    onSuccess: .formValidationSucceed,  
    fields: [  
        FieldValidation(  
            name: \.name,  
            error: \.nameError,  
            rules: [  
                .notEmpty("Username should not be empty"),  
                .length(min: 5, "Username is too short")  
            ]  
        ),  
        FieldValidation(...)  
    ]  
)
```

Result

FormValidation



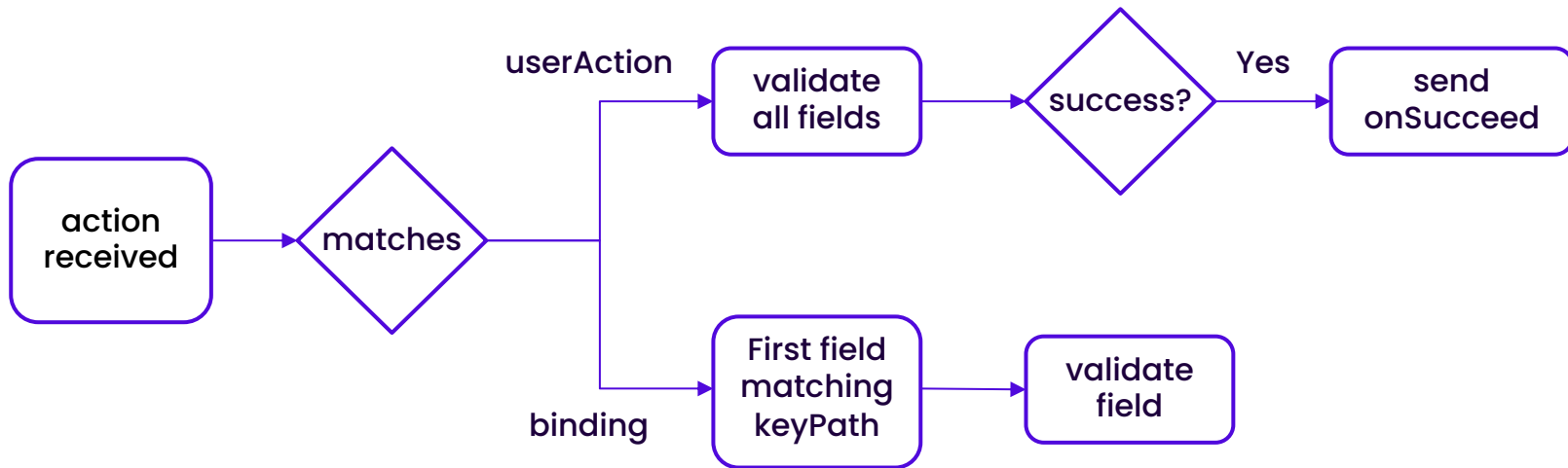
aviv group_

```
FormValidation(  
    userAction: \.validationButtonTap,  
    onSuccess: .formValidationSucceed,  
    fields: [  
        FieldValidation(  
            name: \.name,  
            error: \.nameError,  
            rules: [  
                .notEmpty("Username should not be empty"),  
                .length(min: 5, "Username is too short")  
            ]  
        ),  
        FieldValidation(...)  
    ]  
)
```

Under the hood

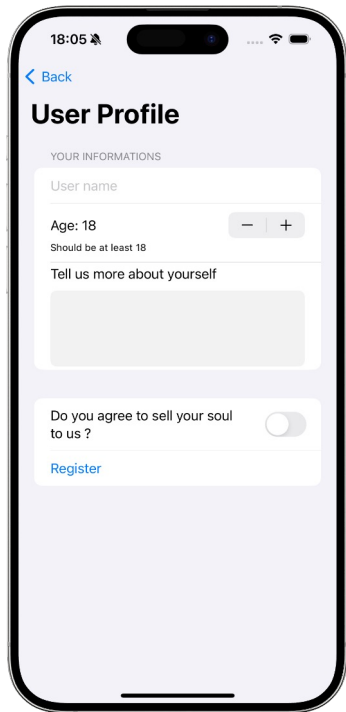
Under the hood

FormValidation



Under the hood

FieldValidation

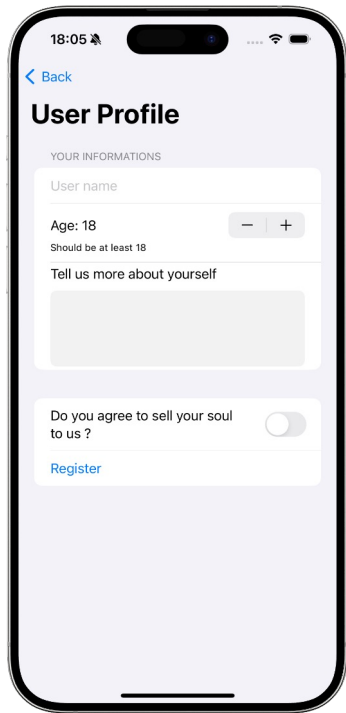


aviv group_

```
struct FieldValidation {  
    func validate(_ state: inout State) -> Bool {  
        let value = state[keyPath: name]  
  
        for rule in rules {  
            if let errorMsg =  
rule.validate(value) {  
                state[keyPath: error] = errorMsg  
                return false  
            }  
        }  
  
        state[keyPath: error] = nil  
        return true  
    }  
}
```


Under the hood

FieldValidation

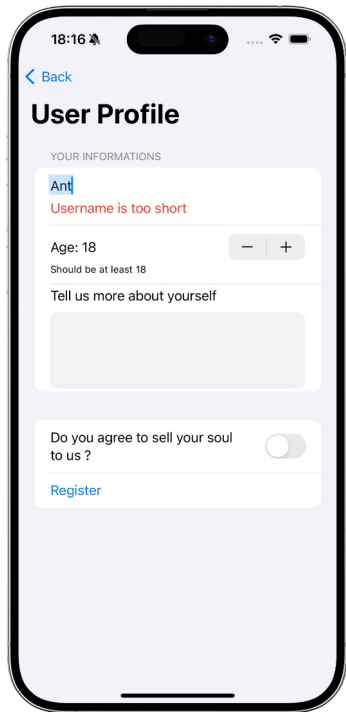


aviv group_

```
struct FieldValidation {  
    func validate(_ state: inout State) -> Bool {  
        let value = state[keyPath: name]  
  
        for rule in rules {  
            if let errorMsg =  
rule.validate(value) {  
                state[keyPath: error] = errorMsg  
                return false  
            }  
        }  
  
        state[keyPath: error] = nil  
        return true  
    }  
}
```

Under the hood

FieldValidation

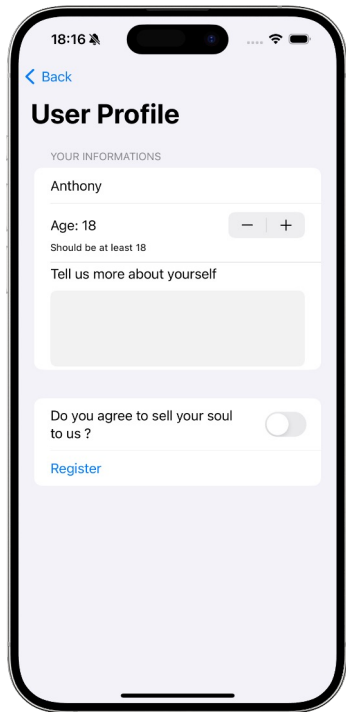


aviv group_

```
struct FieldValidation {  
    func validate(_ state: inout State) -> Bool {  
        let value = state[keyPath: name]  
  
        for rule in rules {  
            if let errorMsg =  
rule.validate(value) {  
                state[keyPath: error] = errorMsg  
                return false  
            }  
        }  
  
        state[keyPath: error] = nil  
        return true  
    }  
}
```

Under the hood

FieldValidation



aviv group_

```
struct FieldValidation {  
    func validate(_ state: inout State) -> Bool {  
        let value = state[keyPath: name]  
  
        for rule in rules {  
            if let errorMsg =  
rule.validate(value) {  
                state[keyPath: error] = errorMsg  
                return false  
            }  
        }  
  
        state[keyPath: error] = nil  
        return true  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .notEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
struct Rule<Value> {  
    let check: (Value) -> Bool  
    let errorMessage: String  
  
    func validate(_ value: Value) -> String? {  
        guard check(value) else {  
            return errorMessage  
        }  
        return nil  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .notEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
struct Rule<Value> {  
    let check: (Value) -> Bool  
    let errorMessage: String  
  
    func validate(_ value: Value) -> String? {  
        guard check(value) else {  
            return errorMessage  
        }  
        return nil  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .notEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
struct Rule<Value> {  
    let check: (Value) -> Bool  
    let errorMessage: String  
  
    func validate(_ value: Value) -> String? {  
        guard check(value) else {  
            return errorMessage  
        }  
        return nil  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .notEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
struct Rule<Value> {  
    let check: (Value) -> Bool  
    let errorMessage: String  
  
    func validate(_ value: Value) -> String? {  
        guard check(value) else {  
            return errorMessage  
        }  
        return nil  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .nonEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
extension Rule {  
    static func length(min: UInt, error: String) ->  
Self  
    where Value: Collection {  
        Self.init(  
            check: { $0.count >= min },  
            errorMessage: error  
        )  
    }  
}
```


Under the hood

Rule

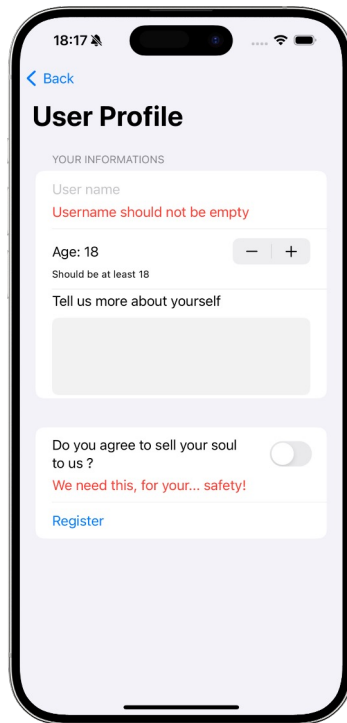
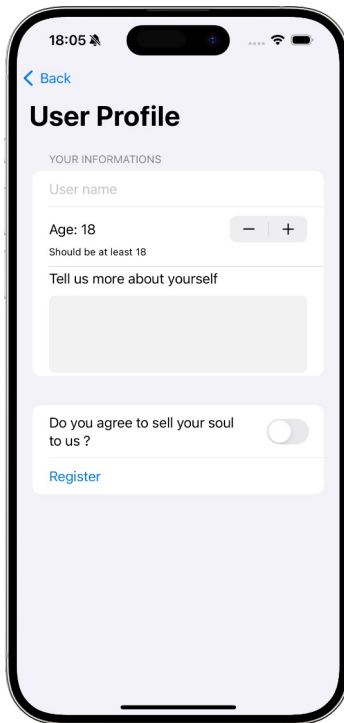
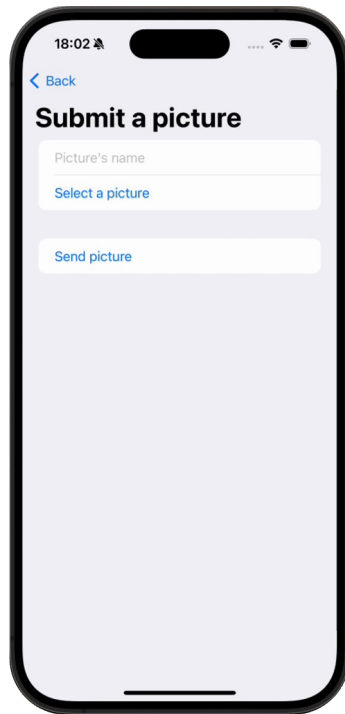
```
rules: [  
    .notEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
extension Rule {  
    static func length(min: UInt, error: String) ->  
Self  
    where Value: Collection {  
        Self.init(  
            check: { $0.count >= min },  
            errorMessage: error  
        )  
    }  
}
```

Under the hood

Rule

```
rules: [  
    .nonEmpty("Username should not be empty"),  
    .length(min: 5, "Username is too short")  
]  
  
extension Rule {  
    static func length(min: UInt, error: String) ->  
Self  
    where Value: Collection {  
        Self.init(  
            check: { $0.count >= min },  
            errorMessage: error  
        )  
    }  
}
```

Result



[https://github.com/
Aviv-public/swift-form-validation](https://github.com/Aviv-public/swift-form-validation)

Thank You

Gmail



anthonymerle.pro
@gmail.com

LinkedIn



Anthony
Chanliau Merle

GitHub



gsl-anthonymerle