



2022

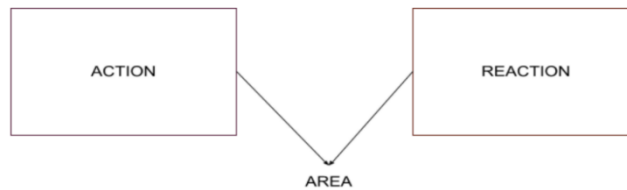
Documentation: AREA

EPITECH PARIS

BENOIT GASNIER, IBRAHIM ZAFAR, JULIEN GARSIA, THÉAU BENEZET

Le projet AREA consiste à créer une suite qui fonctionne de façon similaire à IFTTT et/ou Zapier.

Une AREA consiste à exécuter une réaction (à partir d'un service) quand une action est trouvée



Cette suite est divisée en 3 parties :

- un serveur implémentant les fonctionnalités
- une web app pour une utilisation via navigateur (également disponible sur netlify : <https://tech3area.netlify.app/>)
- une application mobile pour une utilisation via mobile

Stack :

- Server back-end -> Go (Framework Go Gin)
- Application Front-End -> ReactJS
- Application Mobile -> React Native

Authentification (Via firebase) :

- Créer un compte et être connecté
- Via Mail + MDP (création d'un compte)
- Via Google/Github etc (création automatique)

Services :

- ExchangeRatesAPI
- OpenWeatherMap
- League of Legends
- WAQI
- Coronavirus
- Nexmo SMS
- [Go's SMTP (Simple Mail Transfer Protocol)]
- Google

Actions configurables :

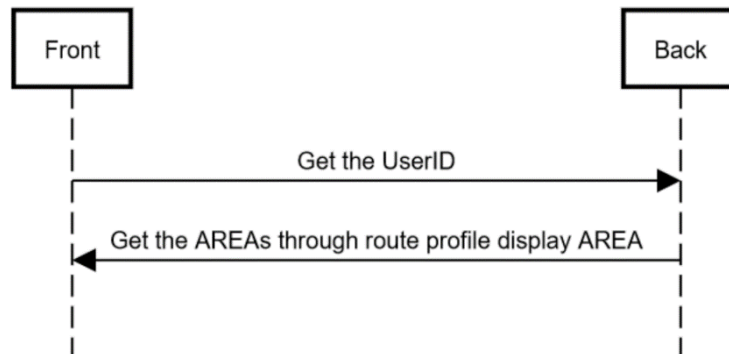
- ExchangeRatesAPI : Une devise est supérieure à l'euro €
- OpenWeatherMap : Météo d'une ville supérieure à une autre
- LoL : Quand les League Points sont supérieurs à 50
- Lol : Victoires du compte utilisateur dépassent l'objectif fixé
- LoL : Défaites du compte utilisateur dépassent l'objectif fixé
- LoL : Quand le tier du compte atteint le tier visé/fixé (ndlr : tier = rang)
- LoL : Quand le rang du compte atteint le rang visé/fixé
- LoL : Quand le compte utilisateur enchaîne des victoires (=Hotstreak)
- LoL : Quand le compte utilisateur atteint le rôle/statut de vétéran
- AQI : Qualité de l'air inférieur à la valeur demandé
- COVID-19 : hospitalisation liée au COVID-19 inférieur à la valeur entrée
- COVID-19 : hospitalisation journalière liée au COVID-19 inférieur à la valeur entrée
- COVID-19 : réanimation liée au COVID-19 inférieur à la valeur entrée
- COVID-19 : réanimation journalière liée au COVID-19 inférieur à la valeur entrée

Réactions :

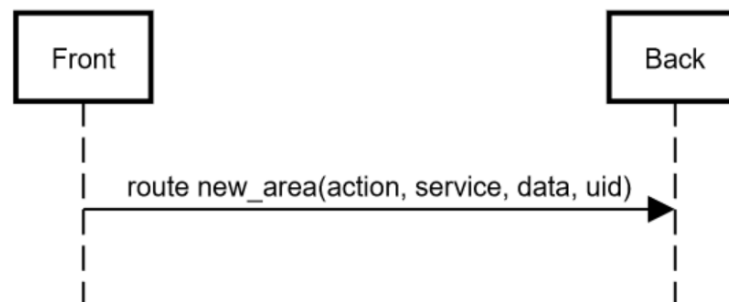
- Nexmo SMS : Envoie de SMS
- Go's SMTP : Envoie d'un mail
- Google : Ajout d'un évènement dans le google calendar

Diagramme de séquence :

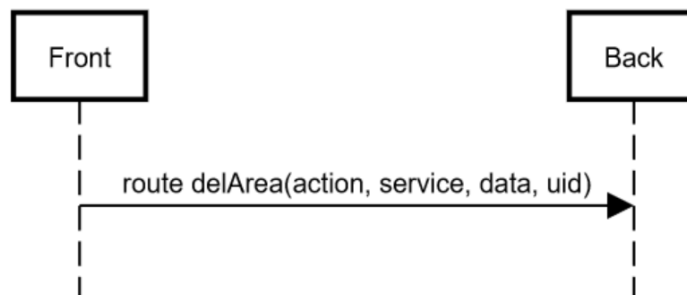
User login



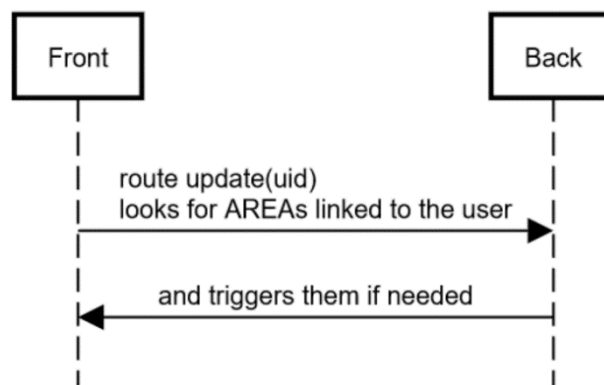
Create an AREA



Delete an AREA



Update an AREA



Créer un service :

Ajouter à la liste de service au fichier : main.go

```
type Services struct {
    Name      string    `json:"name"`
    Action    []Action  `json:"action"`
    Reaction   []Reaction `json:"reaction"`
}

type Action struct {
    Name        string `json:"name"`
    Description string `json:"description"`
}

type Reaction struct {
    Name        string `json:"name"`
    Description string `json:"description"`
}
```

Pour un service il faut :

- Nom
- Action
- Réaction

Pour une action il faut :

- Nom
- Description

Pour une réaction il faut :

- Nom
- Description

Ensuite, il faut implémenter à la fonction `send_rea(c *gin.Context)` pour que cela soit pris en compte lors des updates.