Hello World

Table of Contents

1. 仕様

2. 設計

2.1. TODOリスト
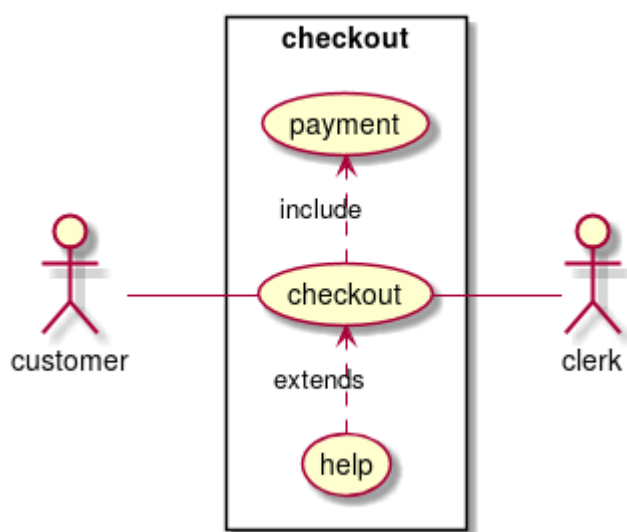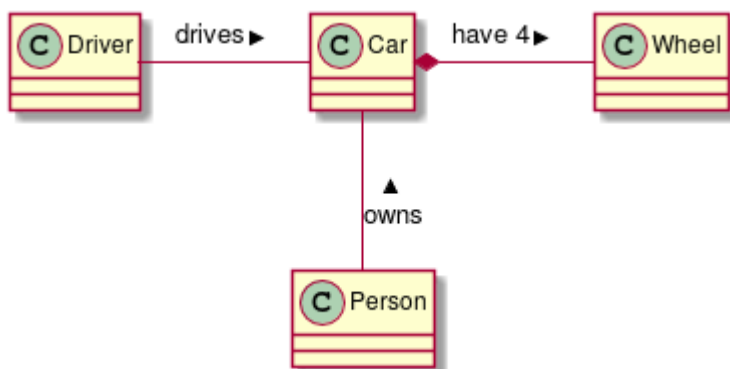
- ☐ TODO
- ☑ ~~TODO DONE~~

2.2. ユースケース図

2.3. クラス図

2.4. シーケンス図

3. 実装

```go
package main

import (
    "errors"
    "fmt"
    "io/ioutil"
    "net/http"

    "github.com/aws/aws-lambda-go/events"
    "github.com/aws/aws-lambda-go/lambda"
)

var (
    // DefaultHTTPGetAddress Default Address
    DefaultHTTPGetAddress = "https://checkip.amazonaws.com"

    // ErrNoIP No IP found in response
    ErrNoIP = errors.New("No IP in HTTP response")

    // ErrNon200Response non 200 status code in response
    ErrNon200Response = errors.New("Non 200 Response found")
)

func handler(request events.APIGatewayProxyRequest) (events.APIGatewayProxyResponse,
error) {
    resp, err := http.Get(DefaultHTTPGetAddress)
    if err != nil {
        return events.APIGatewayProxyResponse{}, err
    }

    if resp.StatusCode != 200 {
        return events.APIGatewayProxyResponse{}, ErrNon200Response
```

```go
    }

    ip, err := ioutil.ReadAll(resp.Body)
    if err != nil {
        return events.APIGatewayProxyResponse{}, err
    }

    if len(ip) == 0 {
        return events.APIGatewayProxyResponse{}, ErrNoIP
    }

    return events.APIGatewayProxyResponse{
        Body:       fmt.Sprintf("Hello, %v", string(ip)),
        StatusCode: 200,
    }, nil
}

func main() {
    lambda.Start(handler)
}
```

```go
package main

import (
    "fmt"
    "net/http"
    "net/http/httptest"
    "testing"

    "github.com/aws/aws-lambda-go/events"
)

func TestHandler(t *testing.T) {
    t.Run("Unable to get IP", func(t *testing.T) {
        DefaultHTTPGetAddress = "http://127.0.0.1:12345"

        _, err := handler(events.APIGatewayProxyRequest{})
        if err == nil {
            t.Fatal("Error failed to trigger with an invalid request")
        }
    })

    t.Run("Non 200 Response", func(t *testing.T) {
        ts := httptest.NewServer(http.HandlerFunc(func(w http.ResponseWriter, r
*http.Request) {
            w.WriteHeader(500)
        }))
        defer ts.Close()

        DefaultHTTPGetAddress = ts.URL
```

```
    _, err := handler(events.APIGatewayProxyRequest{})
        if err != nil && err.Error() != ErrNon200Response.Error() {
            t.Fatalf("Error failed to trigger with an invalid HTTP response: %v", err)
        }
    })

    t.Run("Unable decode IP", func(t *testing.T) {
        ts := httptest.NewServer(http.HandlerFunc(func(w http.ResponseWriter, r
*http.Request) {
            w.WriteHeader(500)
        }))
        defer ts.Close()

        DefaultHTTPGetAddress = ts.URL

        _, err := handler(events.APIGatewayProxyRequest{})
        if err == nil {
            t.Fatal("Error failed to trigger with an invalid HTTP response")
        }
    })

    t.Run("Successful Request", func(t *testing.T) {
        ts := httptest.NewServer(http.HandlerFunc(func(w http.ResponseWriter, r
*http.Request) {
            w.WriteHeader(200)
            fmt.Fprintf(w, "127.0.0.1")
        }))
        defer ts.Close()

        DefaultHTTPGetAddress = ts.URL

        _, err := handler(events.APIGatewayProxyRequest{})
        if err != nil {
            t.Fatal("Everything should be ok")
        }
    })
}
```

4. 参照

- PlantUML[http://plantuml.com]