

第16周展示——地理信息交互部分实现

beego框架

model：每条地理位置信息都是一个struct。全部地理位置信息保存在一个结构体slice中，定义插入、查询、更新操作。

```
type Location struct {
    Id int
    Task_id int
    Soldier_id int
    Longitude float64
    Latitude float64
}
```

controller：定义处理客户端请求的各个函数，包括 GetAll（获取所有位置信息），GetById（获取某人位置信息），Post（更新某人位置信息）。各个函数通过调用model中的操作实现。

```
// @Title 获得所有地址
// @Description 返回所有的地址数据
// @Success 200 {object} models.Location
// @router / [get]
func (u *LocationController) GetAll() {
    ss := models.GetAllLocations()
    u.Data["json"] = ss
    u.ServeJSON()
}
```

router：使用beego的namespace方式绑定路由。

```
func init() {
    ns := beego.NewNamespace("/loc",
        beego.NSNamespace("/location",
            beego.NSInclude(
                &controllers.LocationController{},
            ),
        ),
    )
    beego.AddNamespace(ns)
}
```

测试：通过Swagger UI进行测试。（以获取所有地址为例）

```
curl -X GET "http://localhost:8080/loc/location/" -H "accept: application/json"
```

Server response

Code

200

Details

Response body

```
[
  {
    "Id": 1,
    "Task_id": 1,
    "Soldier_id": 1,
    "Longitude": 1.1,
    "Latitude": 1.1
  },
  {
    "Id": 1,
    "Task_id": 2,
    "Soldier_id": 3,
    "Longitude": 4.4,
    "Latitude": 4.5
  }
]
```

Response headers

```
content-length: 208
content-type: application/json; charset=utf-8
date: Thu, 21 Jun 2018 16:16:55 GMT
server: beegoServer:1.9.2
```

工作进度：周末可以将此部分部署到rancher。