**由于时间有限只实现了部分功能，实现部分基本可以模拟一个简易的分布式系统了。**

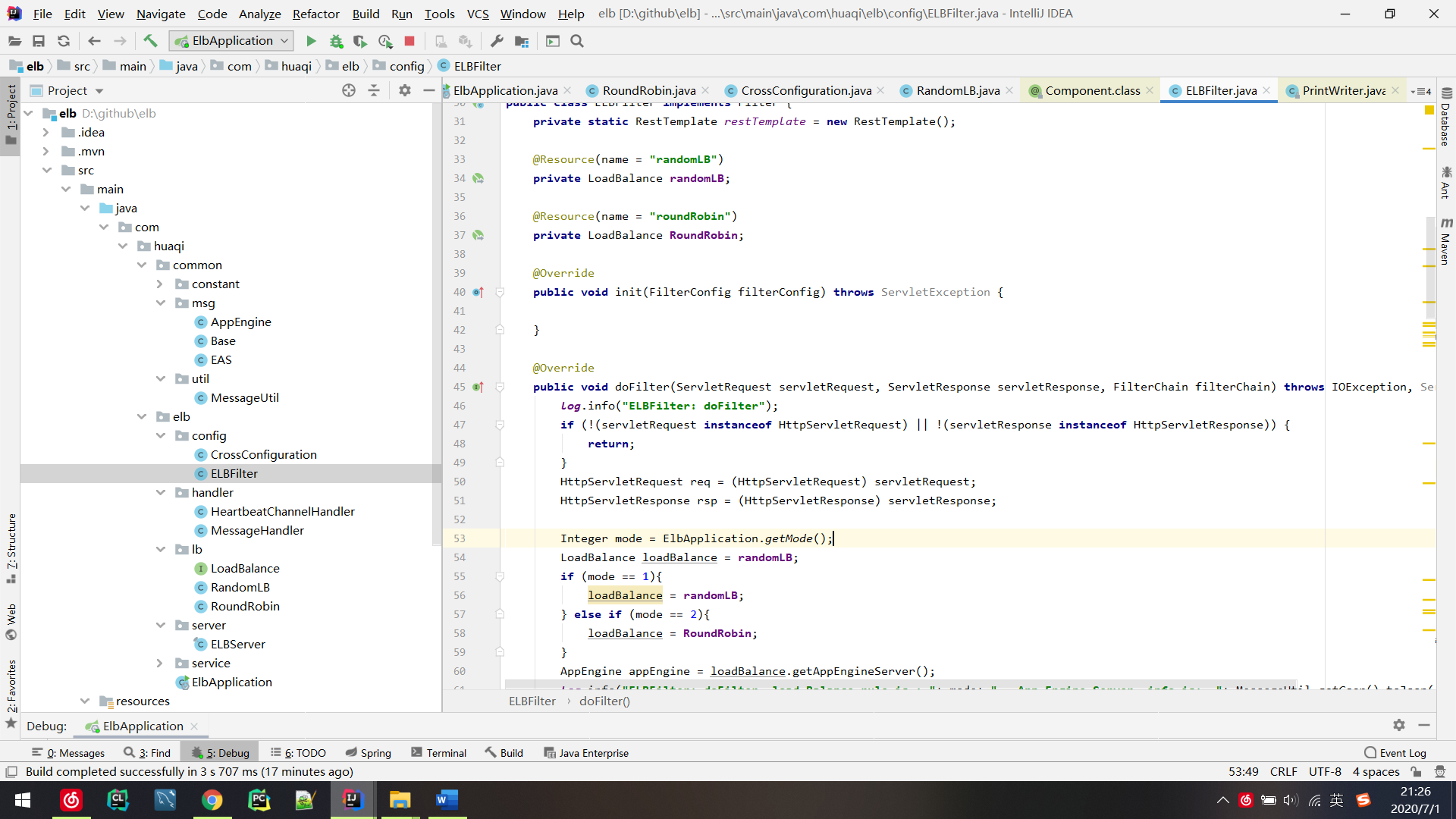
开发环境：windows、idea、java8、maven

相关技术：spring、springboot、netty、lombok等

有三个项目：elb、eas、app-engine，为了方便运行均为springboot。

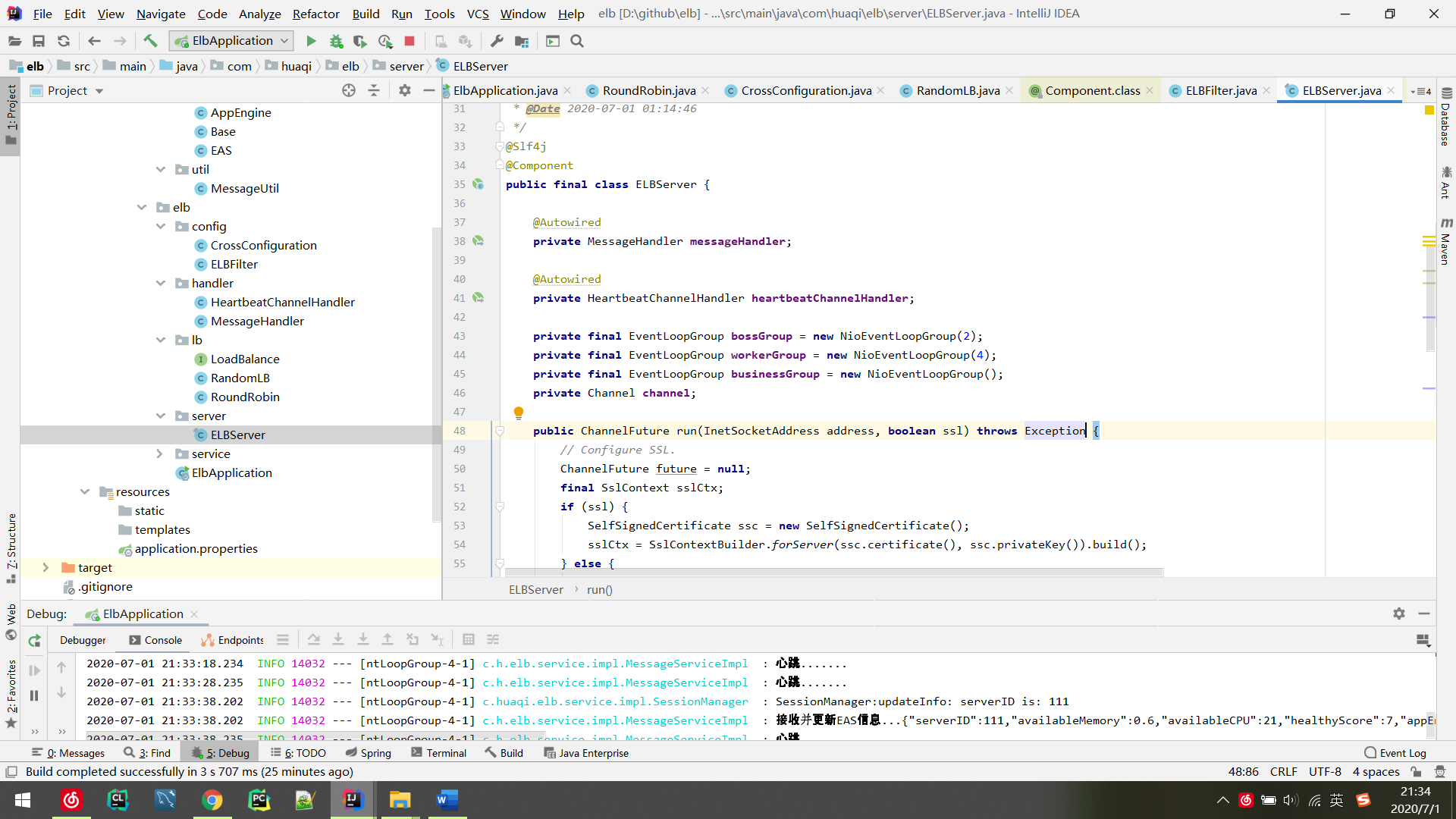
整体思路：

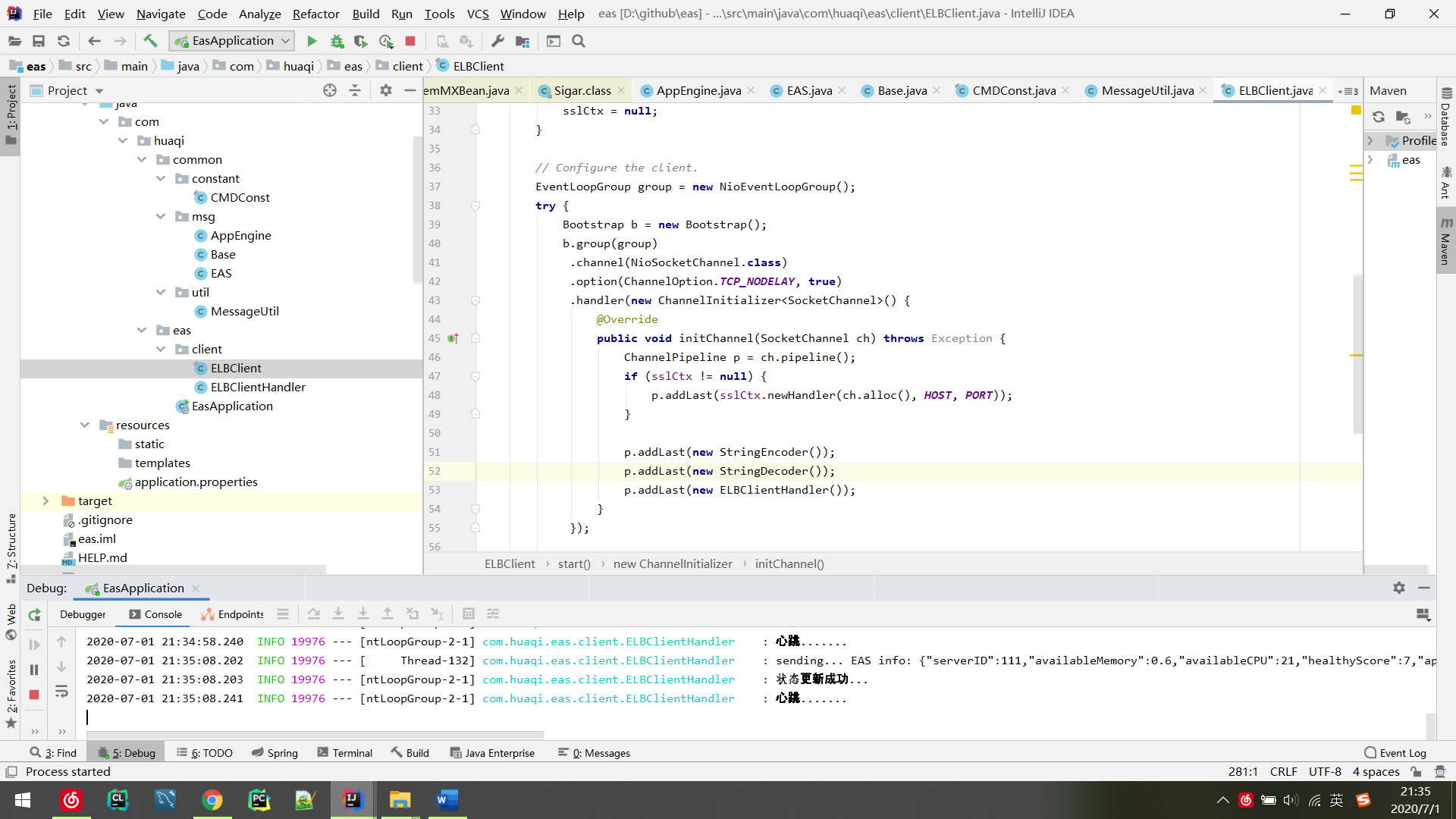
1.elb实现Filter接口，在doFilter方法里调用负载均衡算法（代码实现了随机和轮训两种）做请求转发，如下图：

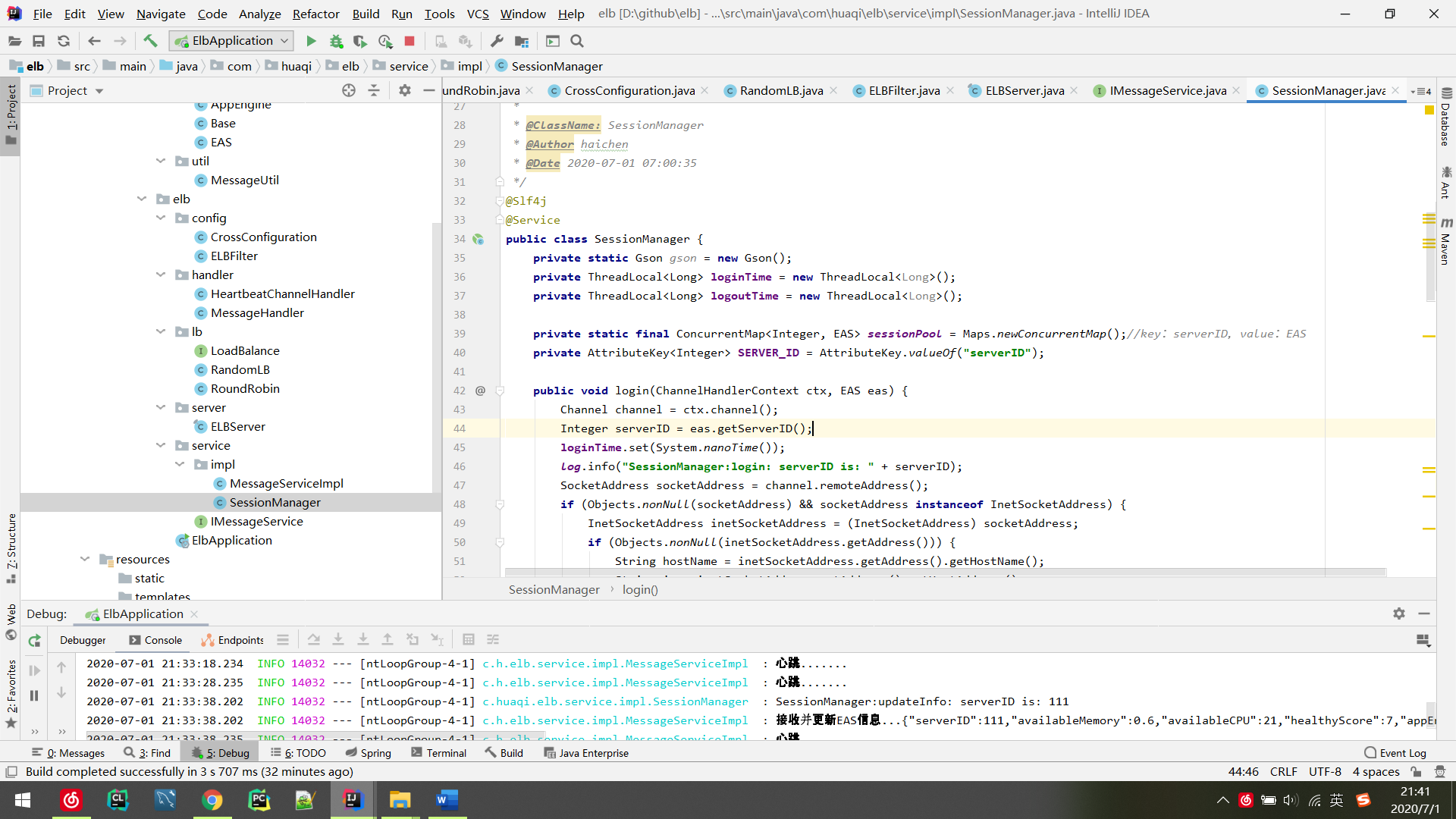


2.elb和eas使用netty通信（elb为netty server端、eas为netty客户端），数据格式为json，数据编解码和心跳机制均**已实现**，使用ConcurrentMap

维护所用eas的信息和app-engine的信息， 如下图，具体逻辑参考代码：



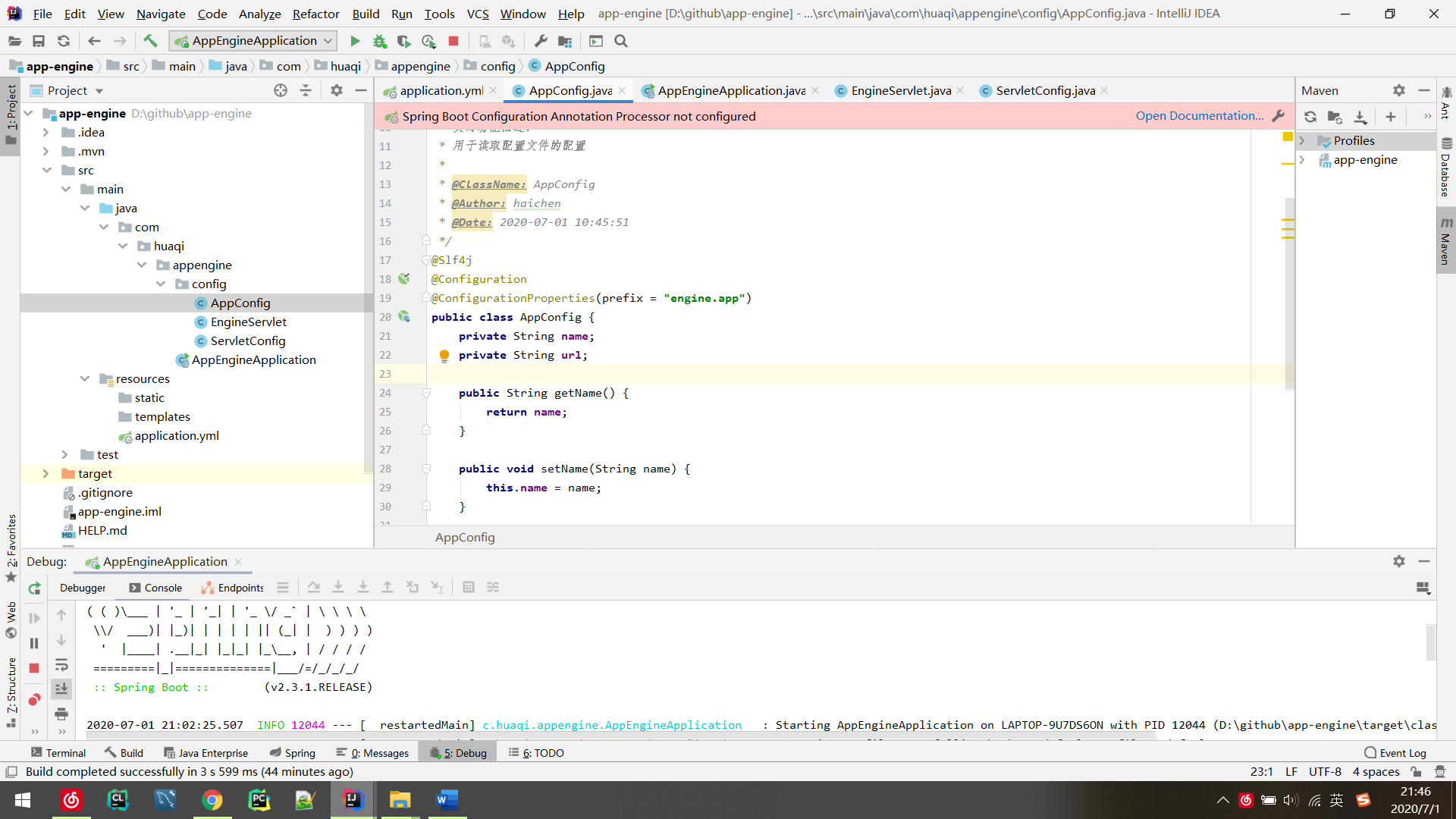




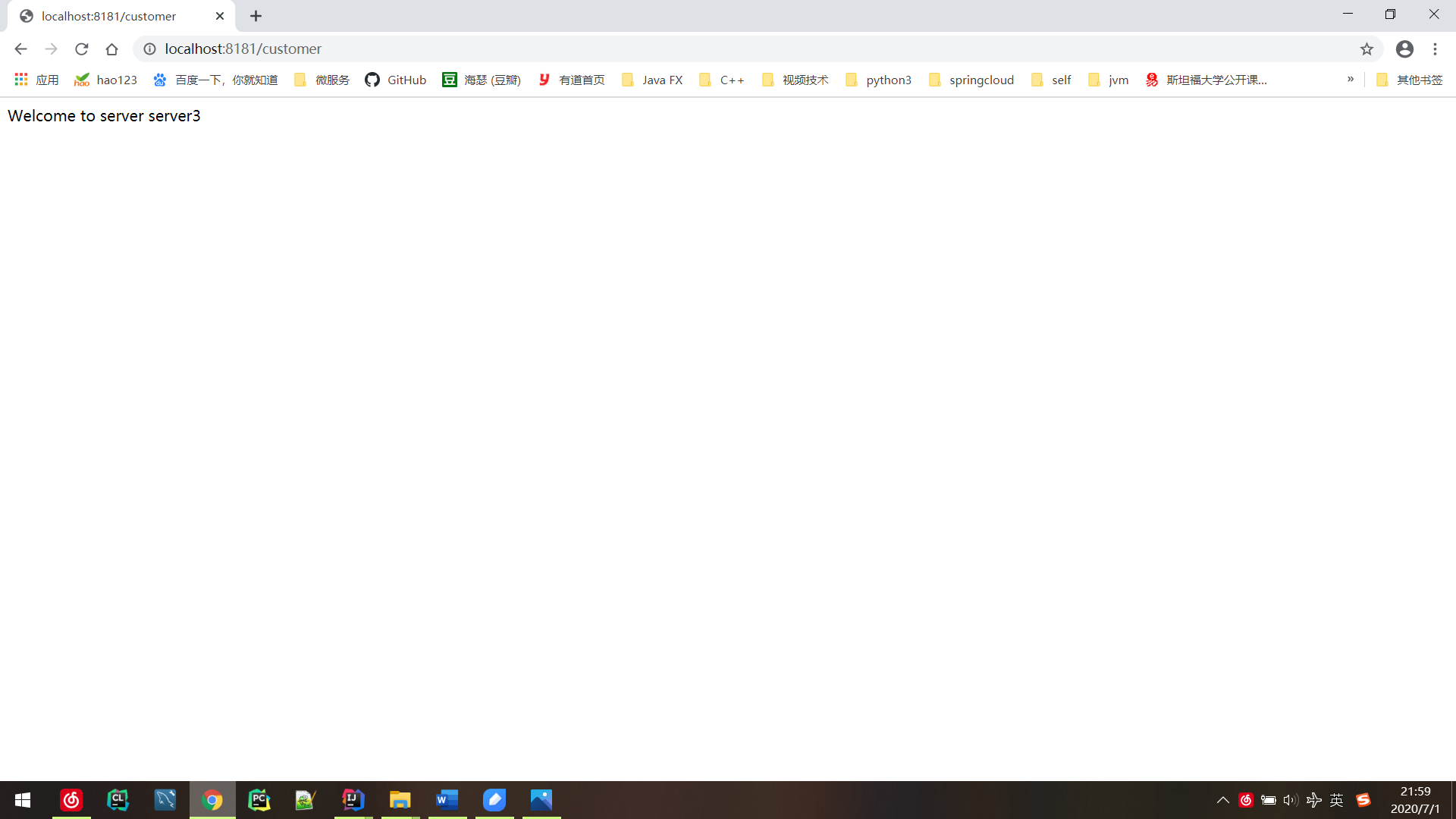
3. eas和app-engine也可以使用netty通信（eas为netty server端、app-engine为netty客户端）. ( 同2**未实现**)

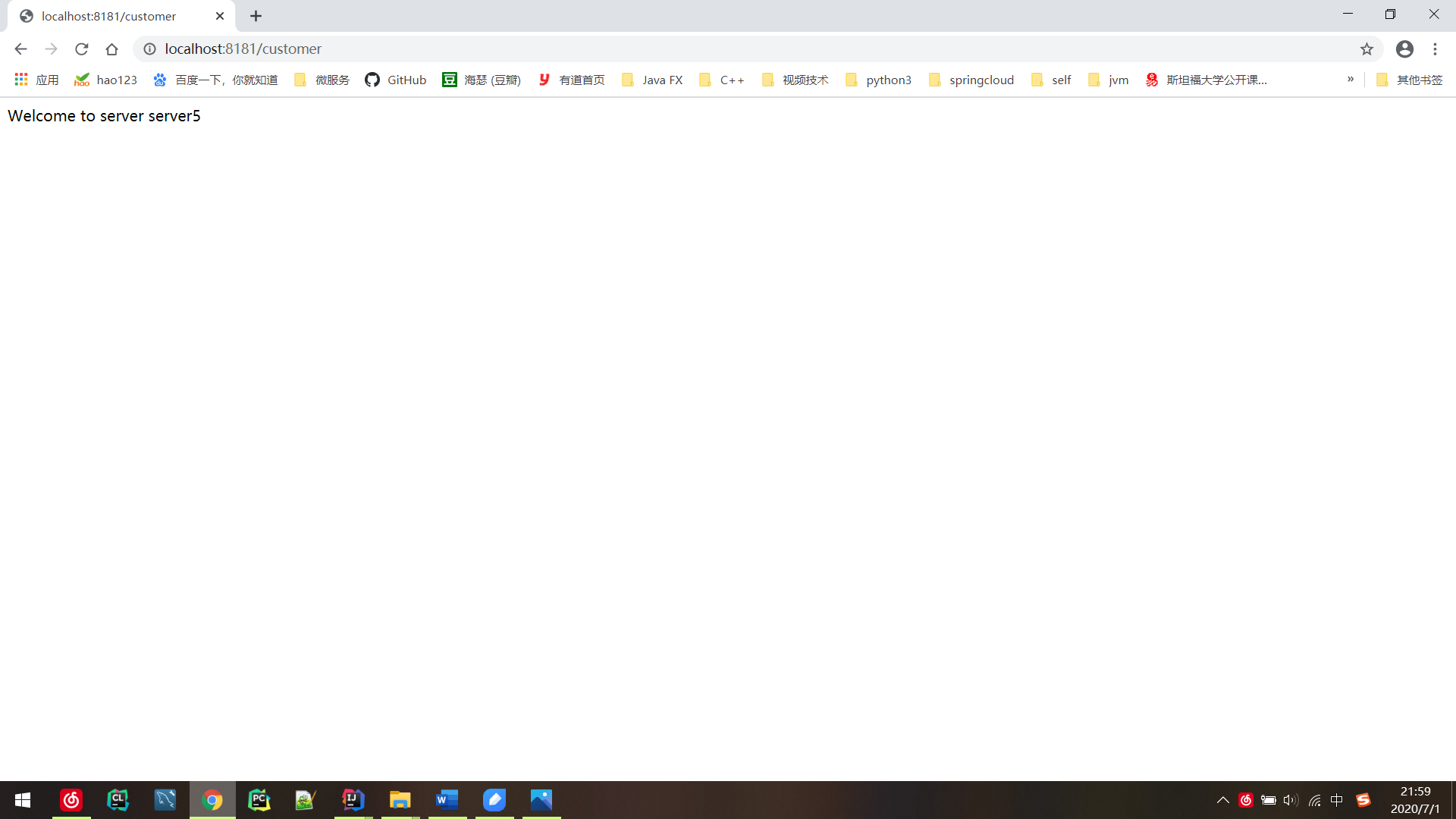
4. app-engine 自定义了EngineServlet，通过注解ConfigurationProperties

读取yml参数，具体请参考代码：



5.演示结果如图：





演示步骤：

1. 启动elb：可以用idea打开项目、或者mvn命令打包后，

命令行运行 **java -jar elb-1.0.0.jar &**

1. 启动eas：可以用idea打开项目、或者mvn命令打包后，

命令行运行 **java -jar** **eas-1.0.0.jar &**

1. 启动app-engine：可以用idea打开项目、或者mvn命令打包后，

命令行运行 **java -jar** **app-engine-1.0.0.jar &**

1. Elb访问地址：<http://localhost:8181/customer> 即可查看结果。

Note: target目录下有打包好的jar包。