|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 技术名称 | 作用 | 是否学习过 |
| 1 | PlayFrameWork | Play！是一个full-stack（全栈的）Java Web应用框架，包括一个简单的无  [Logo](http://baike.baidu.com/pic/play%20framework/6726369/0/6d81800a19d8bc3ee7356449828ba61ea8d34577?fr=lemma&ct=single)Logo  状态MVC模型，具有Hibernate的对象持续，一个基于Groovy的[模板引擎](http://baike.baidu.com/view/4258079.htm)，以及建立一个现代Web应用所需的所有东西。  Play 是一个纯 Java 的框架，它让你保持使用你喜欢的开发工具和类库。如果你已经是一个使用 Java 平台的开发者，那么你不需要切换到另一种语言，其他 IDE 或者其他类库， 而仅仅是切换到一个效率更高的 Java 环境。[1] |  |
| 2 | Echarts | 图标工具 | 了解过 |
| 3 | JBPM | JBPM，全称是Java Business Process Management（业务流程管理），它是覆盖了业务流程管理、[工作流](http://baike.baidu.com/view/60285.htm)、服务协作等领域的一个开源的、灵活的、易扩展的可执行流程语言框架。jBPM是公开源代码项目，使用它要遵循 ASL(Apache License Version 2.0)和EULA(JBoss End User License Agreement)协议。 |  |
| 4 | **WebSocket** | WebSocket protocol 是HTML5一种新的协议。它实现了浏览器与服务器全双工通信(full-duplex)。一开始的握手需要借助HTTP请求完成。现很多网站为了实现即时通讯，所用的技术都是轮询(polling)。轮询是在特定的的时间间隔（如每1秒），由浏览器对服务器发出HTTP request，然后由服务器返回最新的数据给客户端的浏览器。这种传统的HTTP request 的模式带来很明显的缺点 – 浏览器需要不断的向服务器发出请求，然而HTTP request 的header是非常长的，里面包含的有用数据可能只是一个很小的值，这样会占用很多的带宽。  而比较新的技术去做轮询的效果是Comet – 用了AJAX。但这种技术虽然可达到全双工通信，但依然需要发出请求。  在 WebSocket API，浏览器和服务器只需要做一个握手的动作，然后，浏览器和服务器之间就形成了一条快速通道。两者之间就直接可以数据互相传送。在此WebSocket 协议中，为我们实现即时服务带来了两大好处：  1. Header  互相沟通的Header是很小的-大概只有 2 Bytes  2. Server Push  服务器的推送，服务器不再被动的接收到浏览器的request之后才返回数据，而是在有新数据时就主动推送给浏览器。 |  |
| 5 | Html5 |  |  |