技术选型理由

作为一款匿名聊天软件，从便携性角度考虑，显然选择移动端作为产品的客户端。从用户基数来说，

Android用户基数大于IOS、windows等其他OS系统；从开发条件来看，团队不具备开发IOS APP的基础。所以，客户端最终选择Android作为客户端的环境。

//Android框架选择理由

服务器方面，作为一款聊天软件，服务器请求高并发程度较高，选择nodejs作为后台开发语言有效地处理了高并发问题，并降低了软件开发难度。框架方面，选择了著名的Express框架。Express是一个基于Node.js平台的极简、灵活的web应用开发框架，它提供一系列的特性，帮助你创建各种Web和移动设备应用。Express的API提供了丰富的HTTP快捷方法和任意排列组合的Connect中间间，让我们创建健壮、友好的API变得即快速又简单。同时Express不对Node.js已有的特性进行二次抽象，只是在Node.js之上扩展了Web应用所需的基本功能。

后台测试框架使用了Node.js应用中最流行的Mocha。使用Mocha，开发者只需要专注于编写单元测试本身，让Mocha自动运行所有测试，并生成高效、准确的测试报告。同时，基于BDD的 单元测试编写模式，使测试用例更像一份说明书，详细描述软件的每一个功能。最后，使用对Mocha友好的Istanbul，高效快捷地统计测试覆盖率，并自动化生成测试报告。

//持续集成框架选择理由

架构设计

//UML图（部署图、用例图、泳道图）

模块划分

//UML图（包图）

软件设计技术

//Structure Programming

//Object-Oriented Programming

//MVC Pattern