**整体业务需求**

回答者

QQ群

QQ提问者

机器人

Web端

数据库

**QQ用户向机器人发问：**

QQ用户私聊机器人发送问题，由机器人将信息存入数据库，同时向群广播这条消息。web页面被打开时，在page\_load事件中，从数据库按时间排序读出消息并展示。

**QQ用户向机器人回答问题：**

**Qlite插件业务需求**

通过积分驱动积极性：

1. 首次入群每人默认10积分，一条问题耗费[A]积分；
2. 邀请同学入群奖励1积分；
3. 回答问题奖励[A]积分，由提问者结束问题时输入采用者的QQ号来指定奖励一人。
4. 设置问题举报机制，举报成功获取1积分；
5. 业务描述
6. QQ用户提问：
7. 用户向机器人发出提问指令(若非指令则回复引导)
8. 用户发出问题；
9. 询问价值积分
10. 用户出价后，判断剩余积分是否足够
11. 满足后，机器人为问题生成唯一ID，将问题ID，群名，用户QQ号码，问题发送时间，发送内容记录并发送至web端。(这里web接口地址不要写进代码，方便以后随时更改)///(这里还有第二种方案，将这些数据全部存到数据库里，不用请求，由web端自行根据时间读取数据库的信息展示)
12. 发送web请求成功后，将提问者所在的群记录下来，以 ”

[问题内容],当前问题价值[A]积分

--来自xx群;

--回答问题请回复[问题ID];

--当前用户积分[积分值];

--[web端短地址]

答疑成功赚积分

”类似这种的形式向所有群进行广播。

1. QQ用户回答问题：
2. 用户向机器人发送问题ID
3. 机器人检测ID是否合法:包括问题是否已经解决，问题是否存在。
4. 若合法，向回答问题的人展示提问者QQ，并展示当前是第几位前来回答的用户；

基于事件触发的动作：

1. 向机器人结束提问：(主动结束，设计策略自动结束)
2. 判断问题提问者是否是当前结束问题者
3. 若合法，则奖赏回答者(这里很重要，具体的逻辑需要仔细考虑)

**Web端业务需求**

1. 所有业务
2. 展示机器人的消息
3. 根据姓名查电话
4. 微社区(独立于机器人的线上富文本问答)
5. （帮老师开发的期末总分的项目）
6. 模块