

[一休哥]Blockstack课程作业lesson4

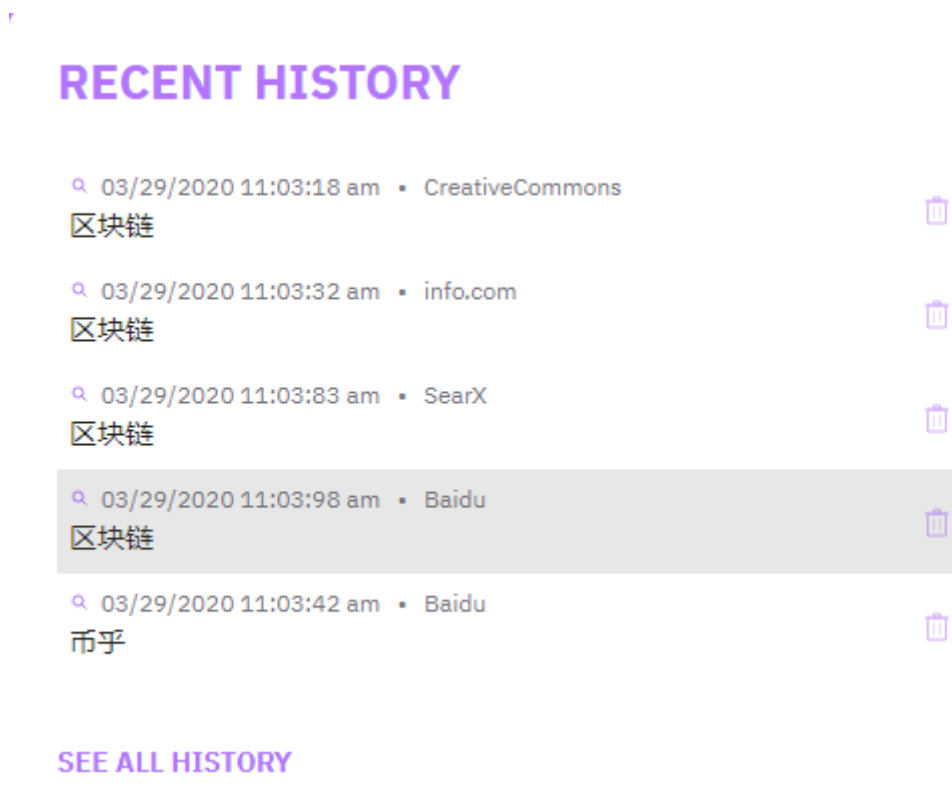
作业1、去中心化搜索引擎 landho 分析

1.1 项目功能简介

landho 是一个去中心化的搜索引擎，用户使用 Blockstack ID 登录应用之后，可以使用其搜索功能。其本质是将常用的搜索引擎聚合起来为用户所用——默认使用谷歌搜索，用户也可以选择不同的搜索引擎进行搜索，它还提供了一个搜索历史纪录的功能，它记录了以下信息：

- 搜索的时间
- 搜索的关键字
- 使用的搜索引擎

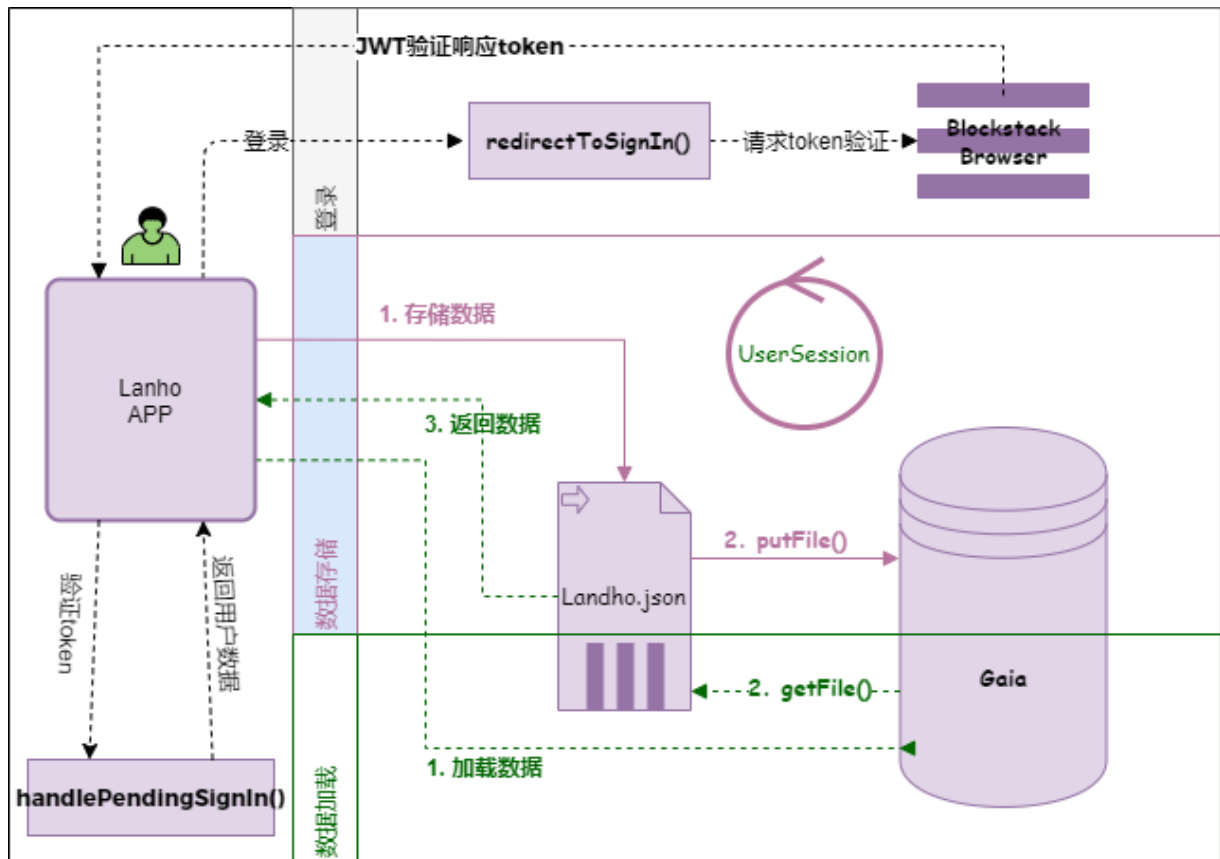
用户进入之后，默认加载5条历史记录，可以选择“**SEE ALL HISTORY**”加载所有的记录。用户可以删除某一条记录，也可以清空搜索记录。



1.2 landho 存在如下bug

- 搜索的时间显示有问题，秒数超过了 60
- 历史记录的排序有问题
- 搜索引擎的图标总是重新加载，影响性能和体验

1.3 landho 的实现方案



1.3.1 授权登录

这部分和所有的 Blockstack 应用一样，获取账户列表，申请授权，返回用户信息，在此不做赘述。

1.3.2 加载数据

登录完成之后，会去尝试到用户的存储下获取应用的数据，也即获取Landho.json，若用户第一次使用应用，则返回404。如果用户之前使用 landho 搜索过，则会通过userSession.getFile()方法，将Landho.json请求到本地，这是一个完整的记录列表，首次只加载最新的5条记录。当用户点击 SHOW ALL HISTORY时，将数据全部渲染出来。

1.3.3 保存数据

首先定义数据结构：

Landho.json：

```
{
  "search_history": [
    {
      "id": 1,
      "engine": "Google",
      "search_text": "区块链",
      "time": 1585007420154
    },
    {
      "id": 2,
```

```
        "engine": "Baidu",
        "search_text": "Blockstack",
        "time": 1585007420254
    }
]
}
```

在这个项目中，存储数据涉及到的操作有：**增、删操作**

- 当用户新增一条搜索记录时，追加记录到search_history数组中。同时将数据pushFile到Gaia存储中。
- 当用户从记录中删除某一条记录时，先从内存中删掉这条记录，然后将完整数据pushFile到Gaia存储中。
- 当用户清空记录时，清空页面内存中的记录，然后将空数据pushFile到Gaia存储中。

1.4 如果我来做？

通过上面的分析，如果我来做，可能会做以下工作：

- 修复前面提到的bug
- 增加一个Recent.json文件来存储最近的几条搜索记录，用户进入时，固定加载这个文件。避免随着用户使用增多，记录增加时文件过大，下载慢。
- 显示所有时，再去加载Landho.json文件。

当然，还有一种方法是去掉显示所有的功能，改为显示更多，使用分页加载的方式，每一页是一个文件，但由于有删除功能，当删除前面的某一条记录时，要改变几乎所有分页的文件（除非不用考虑分页是否必须固定条数）。

作业2. 自己项目的思考题

1. 你选择的产品如何解决了行业痛点？（行业痛点是什么，它是如何解决它的？）

A：目前企业办公存在如下痛点：

- **高昂的人力成本**
当没有业务时，人员都无事可做，员工面临被裁；当业务忙时，急需用人，而人才并非立能就位；
- **员工很辛苦，但大部分利润被企业拿走**
- **激励不足，团队的工作效率不高**

我的产品用于未来分布式办公场景，它无需有一个实体企业，用户通过点对点加入某个虚拟的组织，来协作完成一项工作，在这期间，用户协作时产生的记录留档。该虚拟组织有

牵头人，用户完成的工作事项，可以 publish 给牵头人确认，在目标达成后，参与协作的人，可以获得一定的 TOKEN 奖励。

2. 如果你是这个产品的开发团队，你认为这个产品在中国的商业模式是什么？你认为应该如何改变或添加产品的模块来扩大整个商业模式呢？

A：这个产品的商业模式是让创意者能打破现有的规则，创建一种分布式办公的模式，目标驱动，实时激励，更高效地完成目标。

为了达成该目标，产品需要有如下功能：

- 项目发布
- 项目成员管理
- 项目工作事项列表
- 协作者加入项目组审批流程
- 协作者可查看自己当前的工作列表
- 协作者可以提交工作日志
- 沟通功能
- 对协作者的工作确认以及奖励的发放

以及其它辅助功能。

3. 你认为这个产品相比较中心化产品有哪些不足？不足的点应该怎样改进，可以用技术流程图补充的方式来说明你的观点。

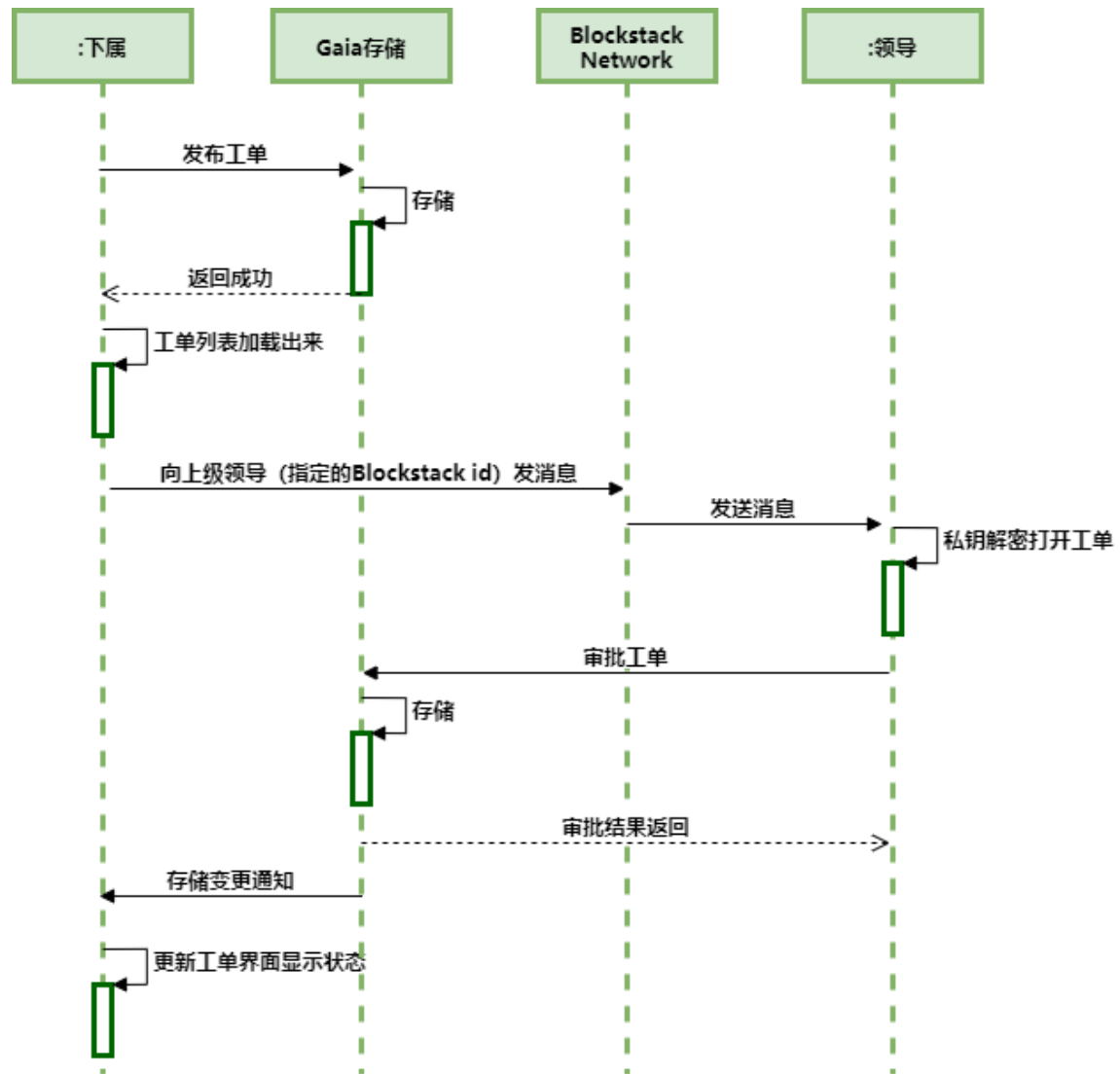
A：相较于中心化产品，这个产品的难点有：

- 很难审核项目发布者的可信程度；
- 奖励的发放；
- 强中心化的组织可能人员都在一起，沟通会更加顺畅。

针对这些问题，可能在产品中要进行如下改进：

- 发布者需要 KYC 认证，（或者其它方式，目前还没想清楚）
- 奖励的发放使用智能合约来完成，
- 产品中自带沟通功能，以便随时交流项目中的问题。

以及其它方面的改进。下图的时序图，是其中某一个业务的交互图，为了方便理解，我们把协作者称为员工，把项目管理员称为领导：

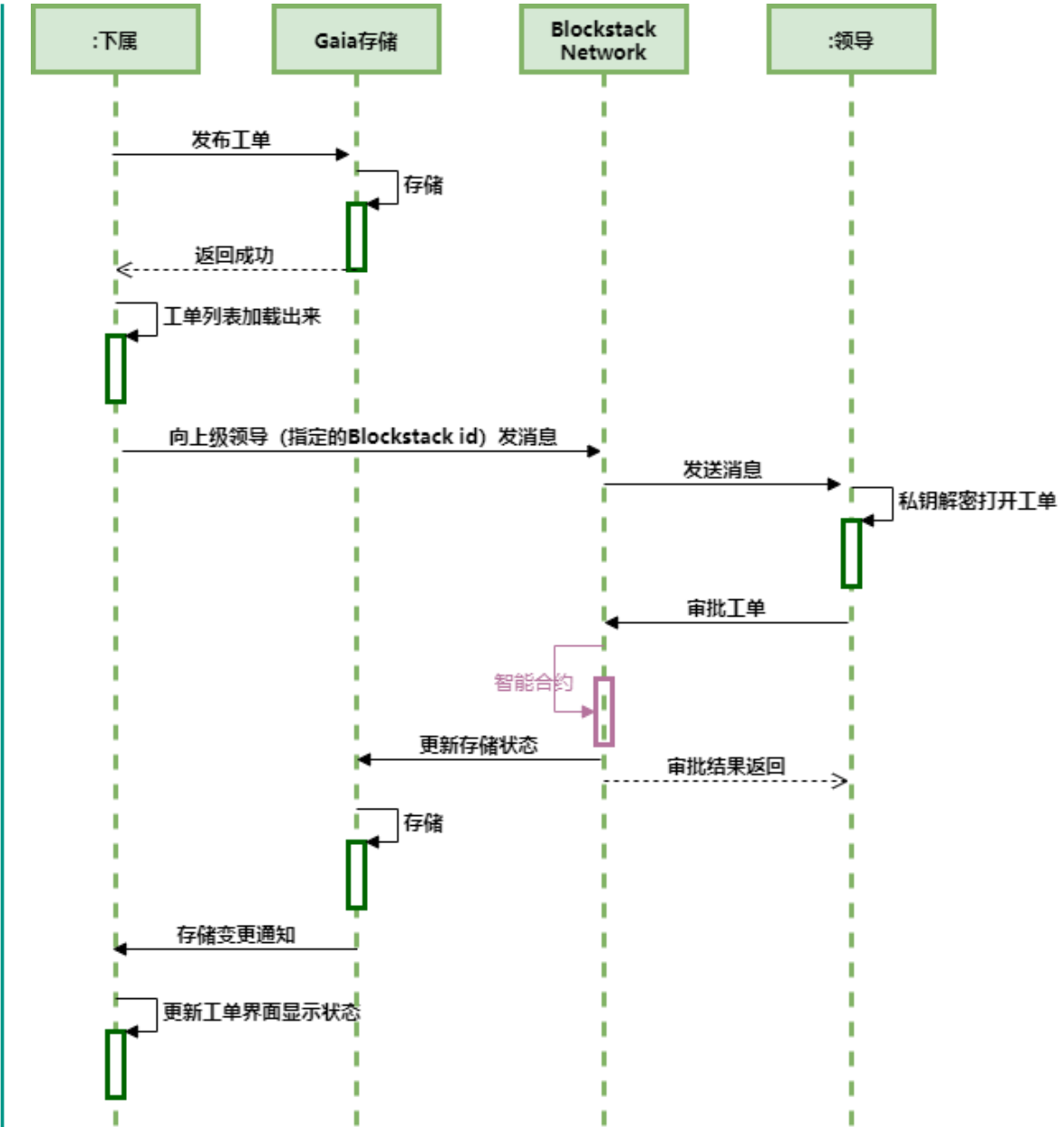


协作者可以发布工单，工单是一个统称，比如工作日志、请假、Milestone确认等。

4. 未来如果推出智能合约，整个产品的技术流程图你认为会有怎样的变化？（使用流程图补充你的观点）

A：如果未来推出智能合约，涉及到奖励的部分，用智能合约来实现。举个例子：项目管理员小王发布了一个项目，项目赏金为 20w 美元。协作者张三加入项目组，然后其拆分工作量，发给项目管理员审核，审核通过后，生成一份合约（赏金抵押）。当张三完成后，交给小王审核验证，验证通过，确认完成，触发智能合约将小王抵押的赏金发放到张三的账户。

还是结合上面的业务场景，如果员工发布的是工作完成的工单，结合上图以及智能合约，可能实现的时序如下：



当然，在实操的过程中，肯定还有很多细节需要考虑。比如，纠纷如何处理等等？