

W1-1-5

几点呀-约见系统

几点呀

01 Task1: 组建项目团队

02 Task2: Web项目建议书

03 Task3: Web项目需求文档

04 Task4: Web应用建模

05 Task5: Web应用架构设计

06 Task6: Web应用设计



Task

1

组建项目团队

Wweeeeb / WebProject

<> Code

Issues 0

Pull requests 0

Projects 0

WebProject /

README.md



or cancel

<> Edit file

Preview changes

```
1 # WebProject
2 Web Engineer Project
3 学号      姓名      Github账户
4 15130110052 汤建波      Vvvvvvia
5 15130110051 胡翔      hu3266569
6 15130110108 齐琪      Lordrian
7 15130110098 汪丹      wd18292002413
8 15130110100 杜晔晔      duyeye
9 15130110097 孙仁      SRcheese
10 15130110099 张欣琪      StephanieFlowerZhang
11 15310110039 王昊      hao-naruto
12
```



Task 2

Web项目建议书

Task2: 项目建议书

01 项目背景

02 项目简介

03 组成部分

04 需求说明

05 项目建议

06 项目总结

Task2: 项目建议书

☀项目背景

当今社会，每个人都有自己塞满的行程，在一个团队中快速寻找到一个合适的会议或者是工作时间是一件相当提高团队效率的事情，同时，团队中很多大家都有的易碎时间也是极其难以在口头讨论中发现。因此我们致力于做这样一个网站为不同的团队去节省时间，去做大家工作中的统筹和总结的平台。

☀项目简介

这个网站针对于任何需要制定约见时间的组织或者个人。不需要单独注册也不需要第三方网站，使用者只需要进入网站，新建见面，然后在一个时间段（可以是一周），选择出所有自己空闲的时间，然后分享链接给自己的组员，每一个收到链接的人都可以在同一个时间段涂画出自己合适的时间，网站接受到这些数据会进行计算，给浏览者呈现出哪些时间是大部份人都有空。网站的特点可以是高效，便利，分享性高

Task2: 项目建议书

组成部分

- ☀ 创建 Event
- ☀ 团队成员登录
- ☀ 成员选定自己的合适时间
- ☀ 成员二次修改

Task2:项目建议书

☀需求说明

我们将致力于为用户提供一个安全、方便、免费的时间管理平台。所以希望能为用户提供最简单的时间统筹管理的平台，并预期提供以下的性能目标：

- 1.功能性：网页所提供的各项功能必须具备准确性和实用性，确保用户可以获取需要的信息。
- 2.可靠性：在用户进行非法操作时，进行提醒报错，维护数据库，避免被恶意修改。
- 3.易使用性：功能明确，界面清晰，保证用户易上手使用。
- 4.高效性：在完成功能时，尽量快速。
- 5.可维护性：网页要易于修改和测试，遇到故障时要能够快速的发现并修改错误。

Task2:项目建议书

☀项目建议

项目主要做一个团队时间管理的系统，考虑到分享链接可能会造成一定的不安全性，或许会因为链接的错误或其他原因而导致团队成员无法加入团队或是加入不正确的团队，因此建议添加一个暗号进入的功能，以确认加入正确的团队。

☀项目总结

项目比较实用而且开发成本中等，所以我们认为，通过我们不断的学习和努力，一定可以成功构建这个网站。



Task 3

Web项目需求文档

Task3: Web项目需求文档

01

引言

02

任务概述

03

需求分析

04

需求前端草图

05

项目交付须知

06

维护须知

Web项目需求文档#引言

☀ 编写目的

写该文档的目的，是对将要进行的工作进行一个系统的分析，对要完成的功能进行规划设计，使网站能在人为控制下完成相应的功能.

☀ 背景

随着社会的不断发展，人们的生活节奏变得越来越快。对于时间的利用率越来越高，对于效率的要求也是越来越高.在日常工作当中，每个人行程也是安排的满满的。因此需要与人帮助他们进行统筹，节约时间.

Web项目需求文档#任务概述

☀ 服务

我们组构建的网站主要是帮助团队进行约见统筹，反馈最佳的时段.

☀ 目标对象特点

对象：组员空闲时间不同步的团队.

规模：初步估计为十个左右.

用户要求：能看懂汉字，并且能按照提示操作计算机.

维护人员要求：懂得网站的运行和维护的原理.

☀ 系统开发环境

硬件：计算机 Windows x64/ Linux 系统

软件：Html，css，JavaScript 以及 Java 后端和 Mysql 等相关开发环境

☀ 系统运行环境

硬件：PC 端，移动端

软件：Internet 浏览器

Web项目需求文档#需求分析

☀对功能的需求

用户的功能需求

系统需要进行的工作

功能的实现

☀对系统性能的需求

Web项目需求文档#需求分析

☀对功能的需求

用户的功能需求

一. 创建 event.

角色	用户 (创建者)
前提条件	登陆网站
事件	<div>1. 登陆网站</div> <div>2. 点击网站上的创建 event 按钮</div> <div>3. 设定 event 的生存周期.</div> <div>4. 填写想要创建的 event 的信息(包括时区选择 , 每日起止时间) 并点击完成按钮.</div> <div>5. 将系统生成的链接分享给组员.</div> <div>6. 事件流结束.</div>

Web项目需求文档#需求分析

☀对功能的需求

用户的功能需求

二. 参与 event

角色	用户（参与者）
前提条件	通过分享链接进入 event
事件	1.确定自己的用户名和密码. 2.选择自己有空余的时间段.. 3.事件流结束.

Web项目需求文档#需求分析

☀对功能的需求

用户的功能需求

三、二次修改个人信息

角色	用户（参与者）
前提条件	通过分享链接进入 event 且已有用户名和密码
事件	<ol style="list-style-type: none">1. 修改自己有空余的时间段.2. 事件流结束.

Web项目需求文档#需求分析

☀对功能的需求

系统需要进行的工作

对 event 的时间段进行统筹得出最优解.

角色	系统
前提条件	用户参与了 event
事件	<ol style="list-style-type: none">1. 接收并保存用户填写的空余时间段.2. 计算出最优约见时间, 反馈到团队时间表中 (表中方格颜色越深, 说明成员空余时间的重复程度越高).3. 事件流结束.

Web项目需求文档#需求分析

☀对功能的需求

功能的实现

1. 前端：html css JavaScript（Jquery，moment.js 库）
2. 后端：JavaEE 三层架构 TomCat 服务器 Mysql 数据库
3. 测试：单元测试 功能测试 性能测试等
4. 维护：内容维护 服务器维护等

Web项目需求文档#需求分析

☀对功能的需求

☀对系统性能的需求

1.精度

我们组的项目对于用户的反馈为时间，因此对于输出的精度控制为以小时为单位.

2.时间特性需求

对于系统的效率，初步设定的系统响应时间为 500ms.



Task **4**

Web应用建模



Task4: Web应用建模

01 功能需求

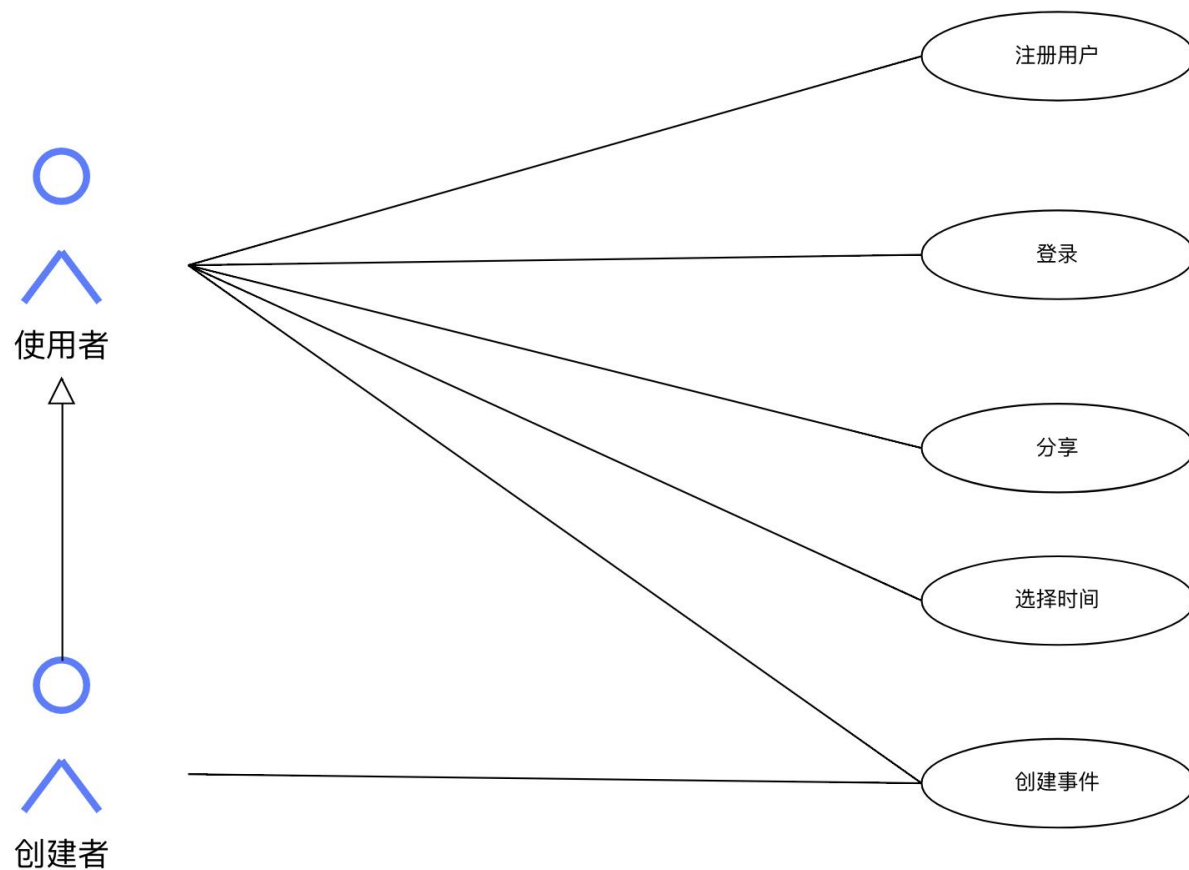
02 内容

03 超文本

04 适应性

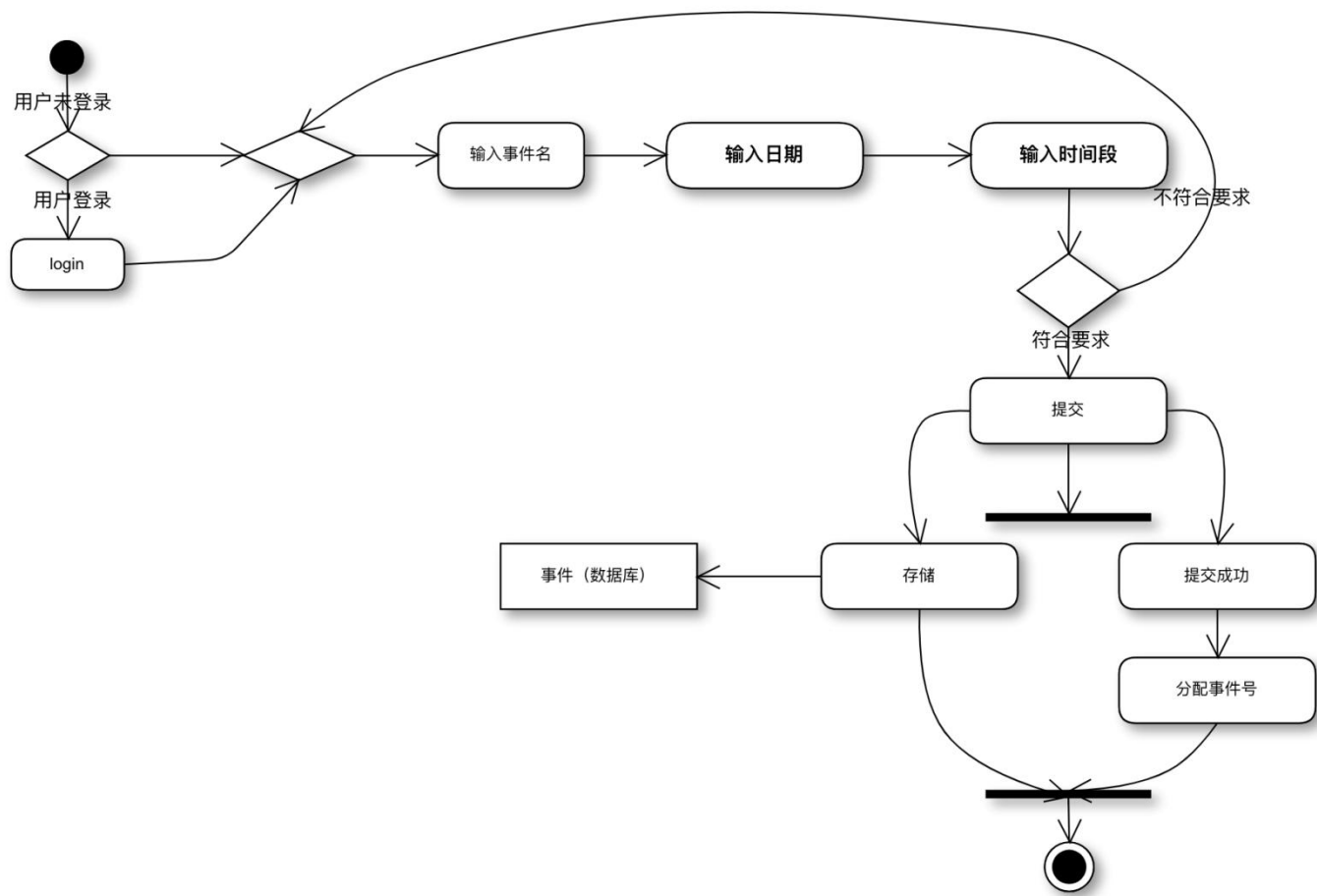
Task4:Web应用建模#功能需求

用例图



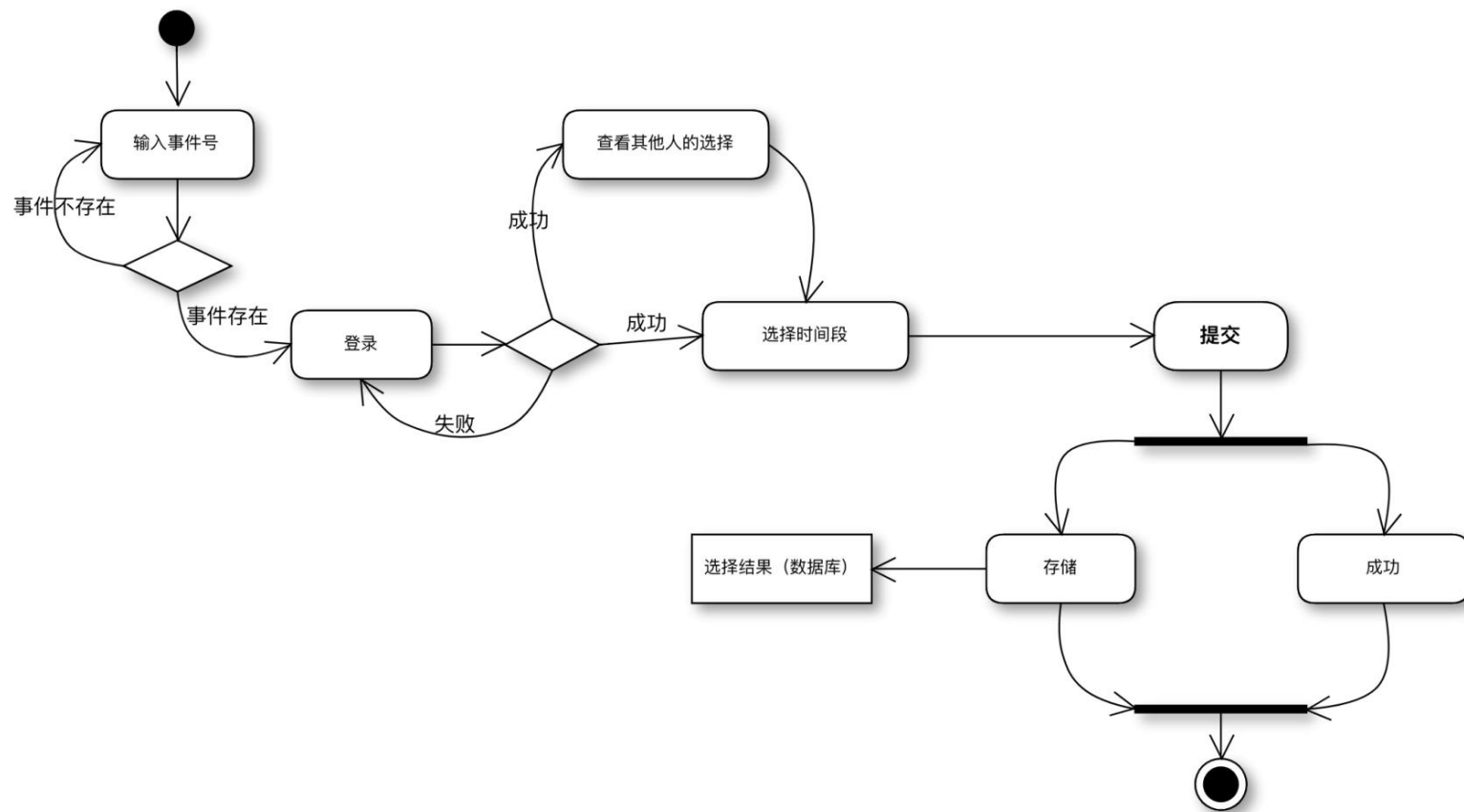
Task4:Web应用建模#功能需求

创建事件活动图



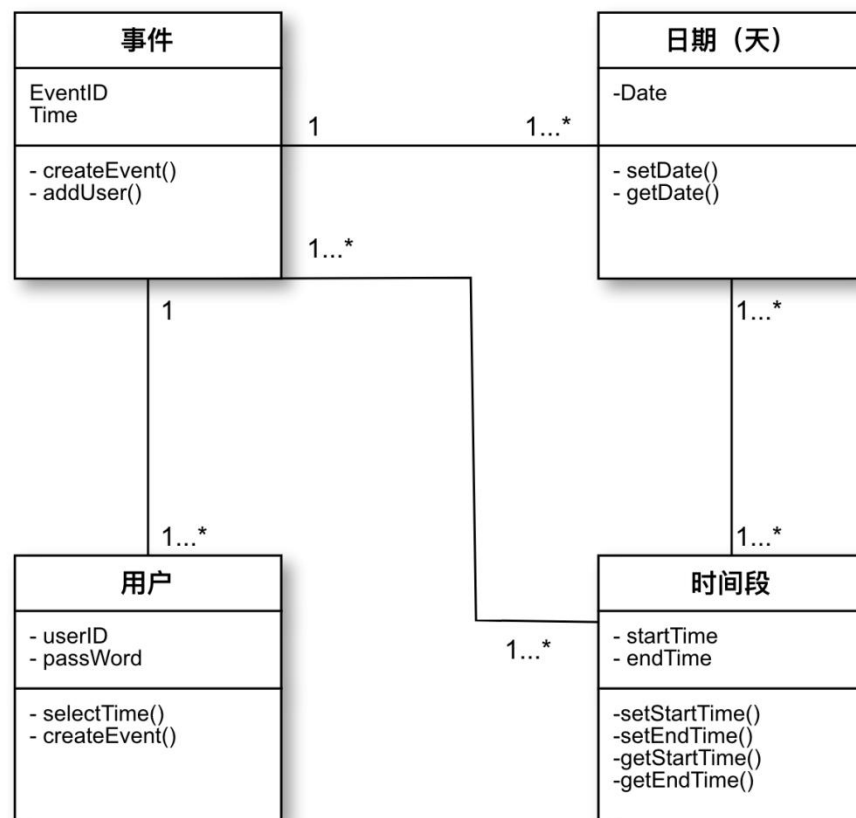
Task4:Web应用建模#功能需求

选择时间活动图



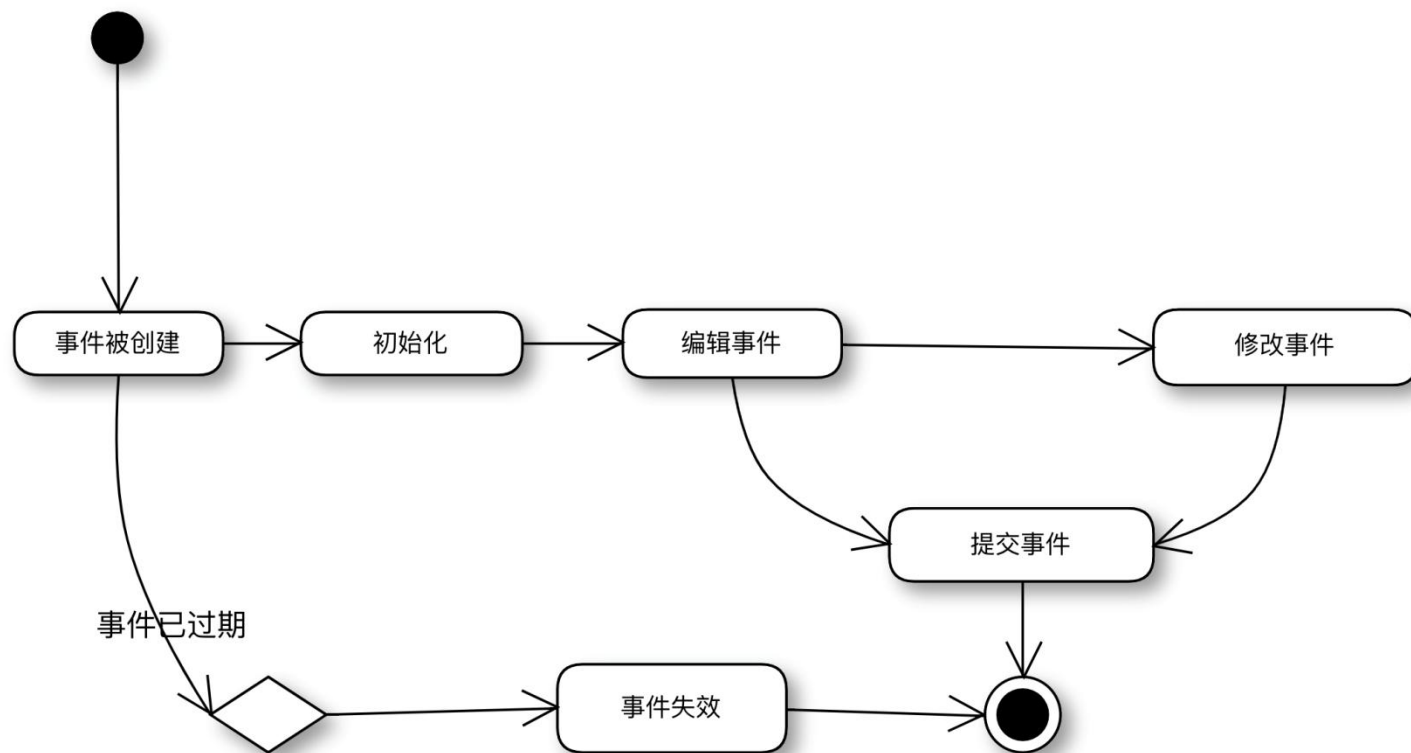
Task4:Web应用建模#内容

系统类图



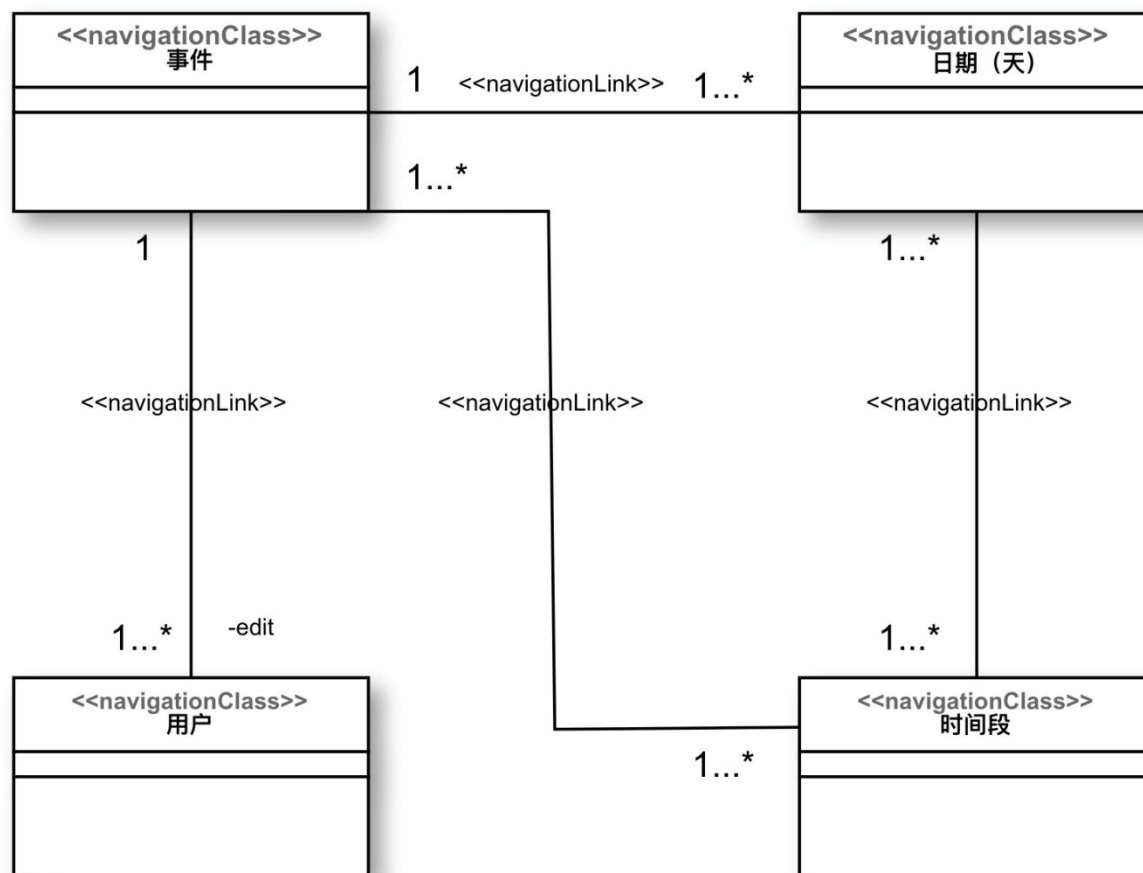
Task4:Web应用建模#内容

事件类状态图



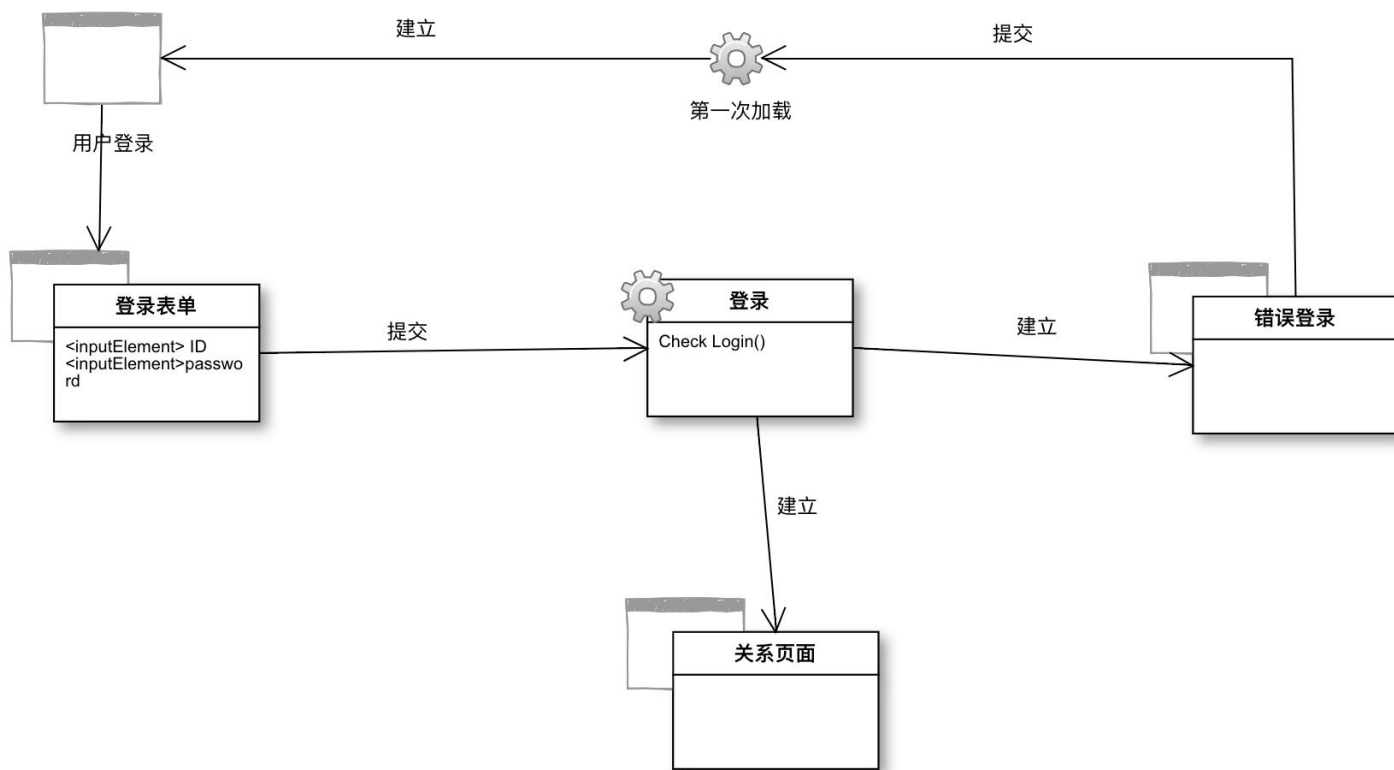
Task4:Web应用建模#超文本

参与者视图的超文本结构模型



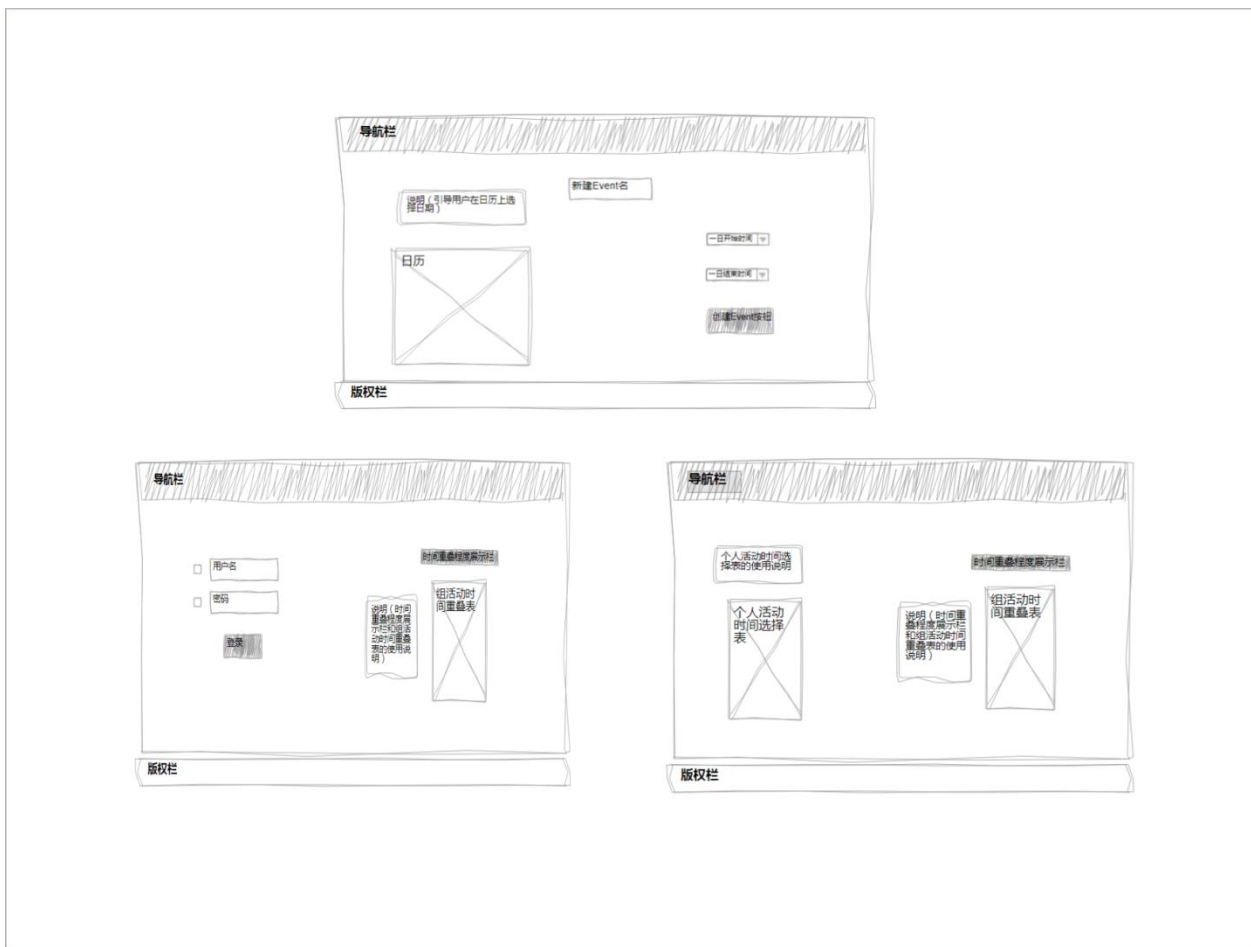
Task4:Web应用建模#超文本

用户登录的WEB模型



Task4:Web应用建模#适应性

页面的动态适应





Task 5

Web应用架构设计



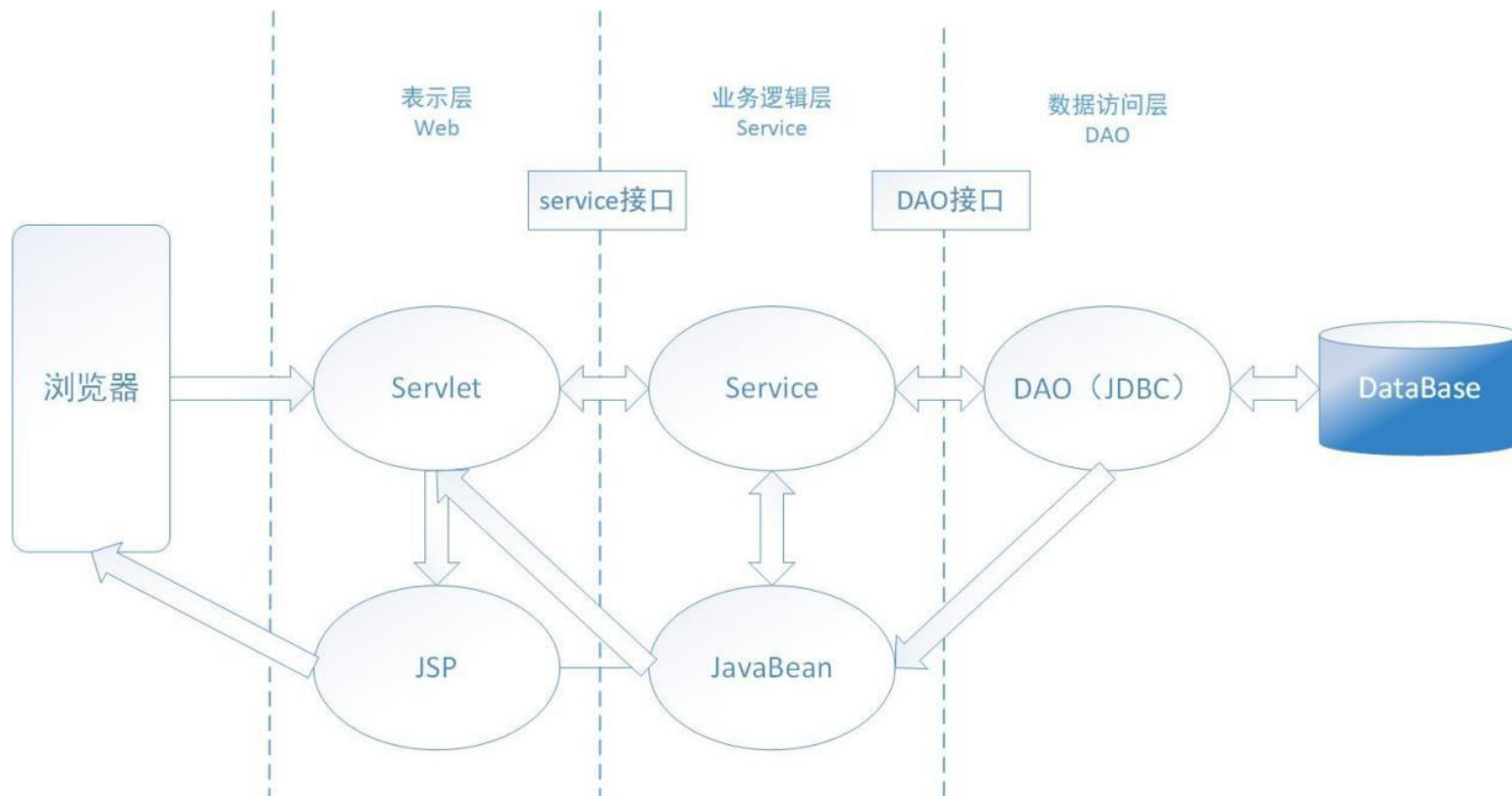
Web应用架构设计

本项目采用了JAVA EE的层次架构，分为以下三层：

- ☀ 表现层（Web）：主要对用户的请求接受，以及数据的返回，为客户端提供应用程序的访问。
- ☀ 业务逻辑层（Service）：针对具体问题的操作，也可以说是对数据层的操作，对数据业务逻辑处理。
- ☀ 数据访问层（DAO）：该层执行事务直接操作数据库，针对数据的增添、删除、修改、查找等。数据库用于存储不同用户记录的空闲时间段，用户可以通过操作访问数据库，进行增删改查等。

Web应用架构设计

- ☀ 前端用了jQuery框架和moment.js库。
- ☀ 后端用了JavaEE， MySql 数据库， Tomcat web应用服务器。





Task 6

Web应用设计



Task6: Web应用设计

01

交互设计

02

展示设计

03

内容设计

04

功能设计

Task6:Web应用设计#交互设计

☀ 用户交互

☀ 用户页面组织

☀ 导航设计

☀ 复杂活动的会话设计

Task6:Web应用设计#交互设计

☀ 用户交互

(1) 创建Event页面

Event命名文本框：根据框中默认文字提示，在框中输入Event名。

日期选择按钮：给出当月日历，根据文字引导用户点击日历中要选的日期的按钮。

一日时间开始/结束下拉选择框：点击下拉框，选择开始/结束时间。

创建Event按钮：将页面中填写的信息存储成一个Event，导航到第二个页面。

Task6:Web应用设计#交互设计

☀ 用户交互

- (1) 创建Event页面
- (2) 登录页面
- (3) 时间选择页面
- (4) 关于我们页面

Task6:Web应用设计#交互设计

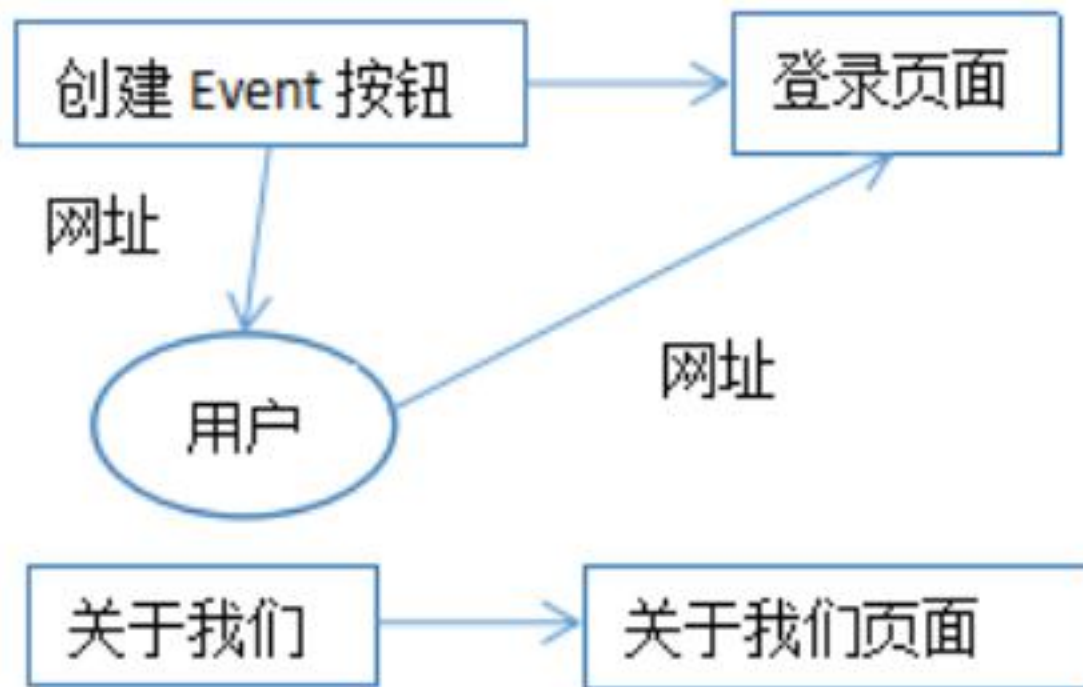
☀ 用户页面组织

主要分为三个页面：Event页面、登录页面、时间选择页面。

屏幕大小：一个页面内容容量较小，只使用滚动条。

Task6:Web应用设计#交互设计

☀ 导航设计



Task6:Web应用设计#交互设计

☀ 复杂活动的会话设计

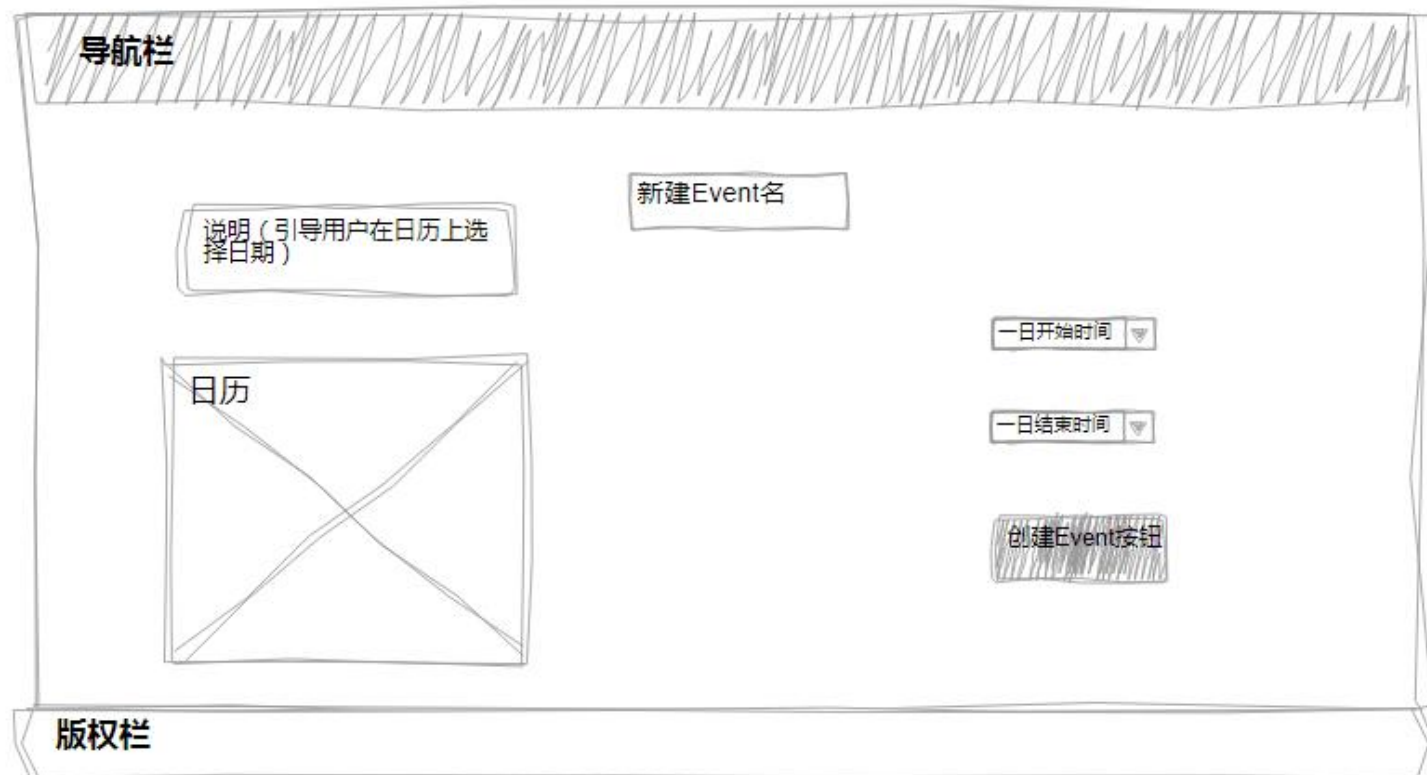
回退：从Event页面创建Event后链接到登录页面，未按登录按钮前回退，页面从登录页面回退到Event页面，用相同的信息创建Event，获得与回退前不同链接的登录页面网址。

其它情况下回退，关闭页面，其中时间选择页面，如果在回退前有进行时间选择，结果会保存。

Task6:Web应用设计#展示设计

☀ 页面布局设计（线框）

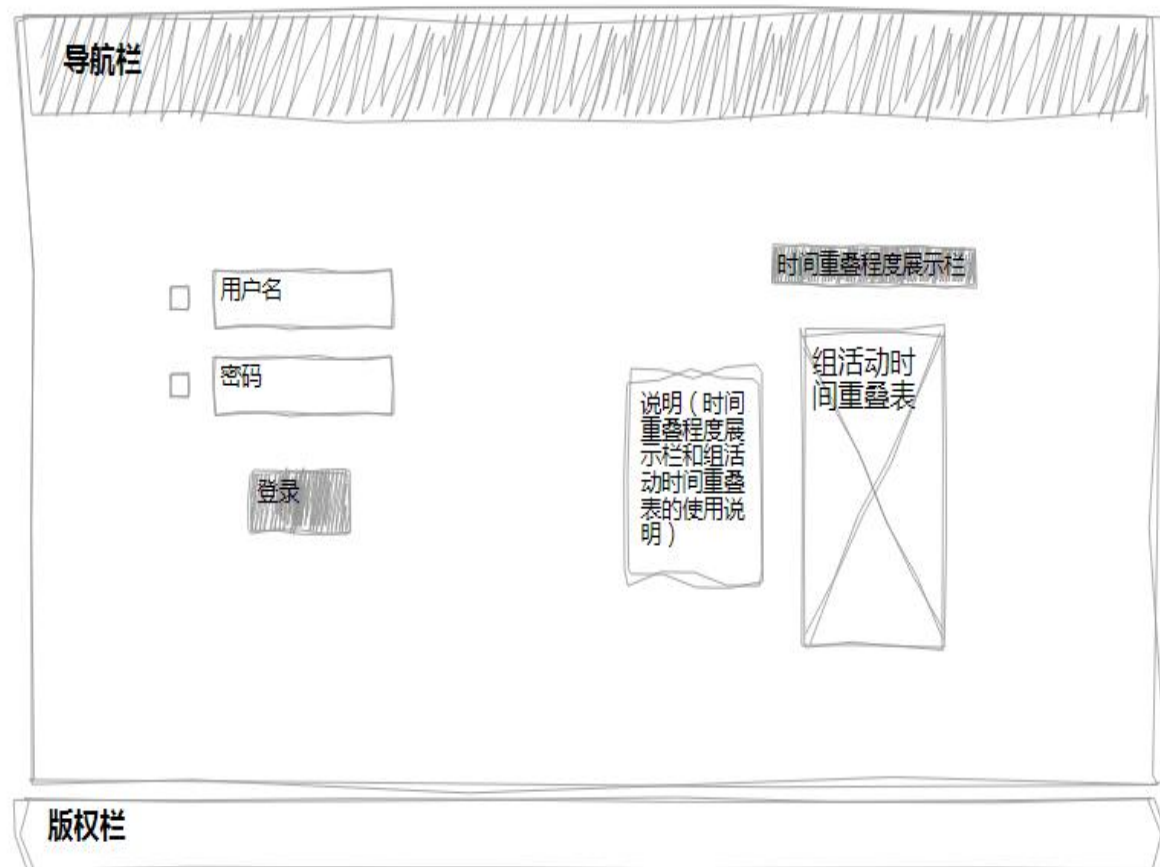
☀ 创建事件页面：上下框架型



Task6:Web应用设计#展示设计

☀ 页面布局设计（线框）

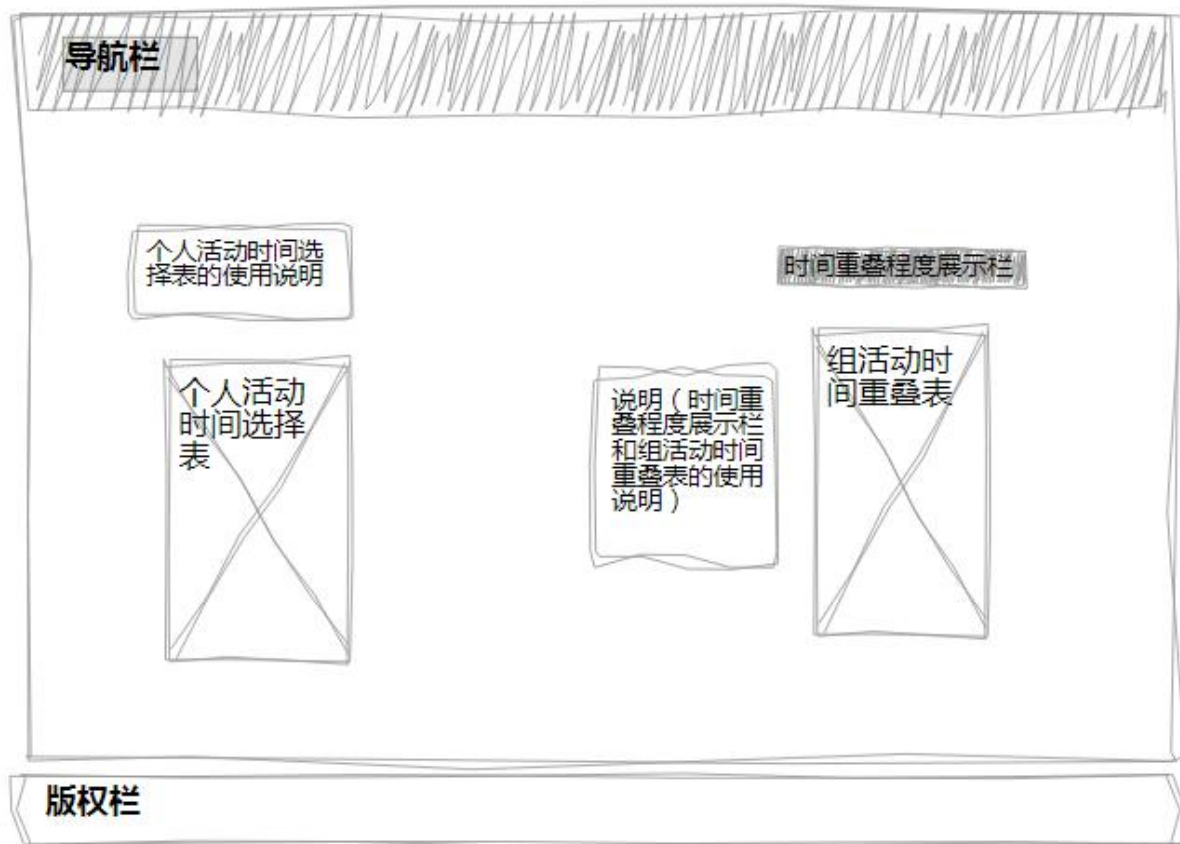
☀ 登录页面：上下框架型



Task6:Web应用设计#展示设计

☀ 页面布局设计（线框）

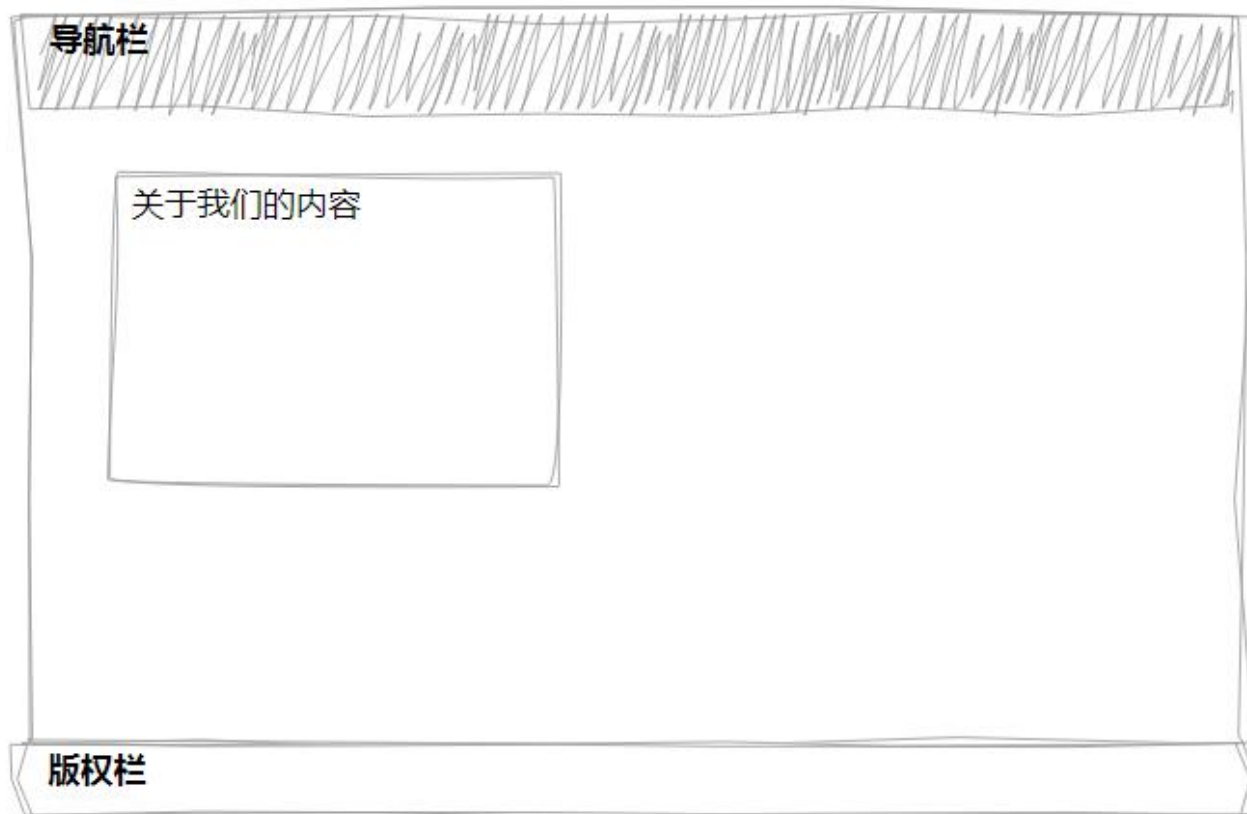
☀ 时间选择页面：上下框架型



Task6:Web应用设计#展示设计

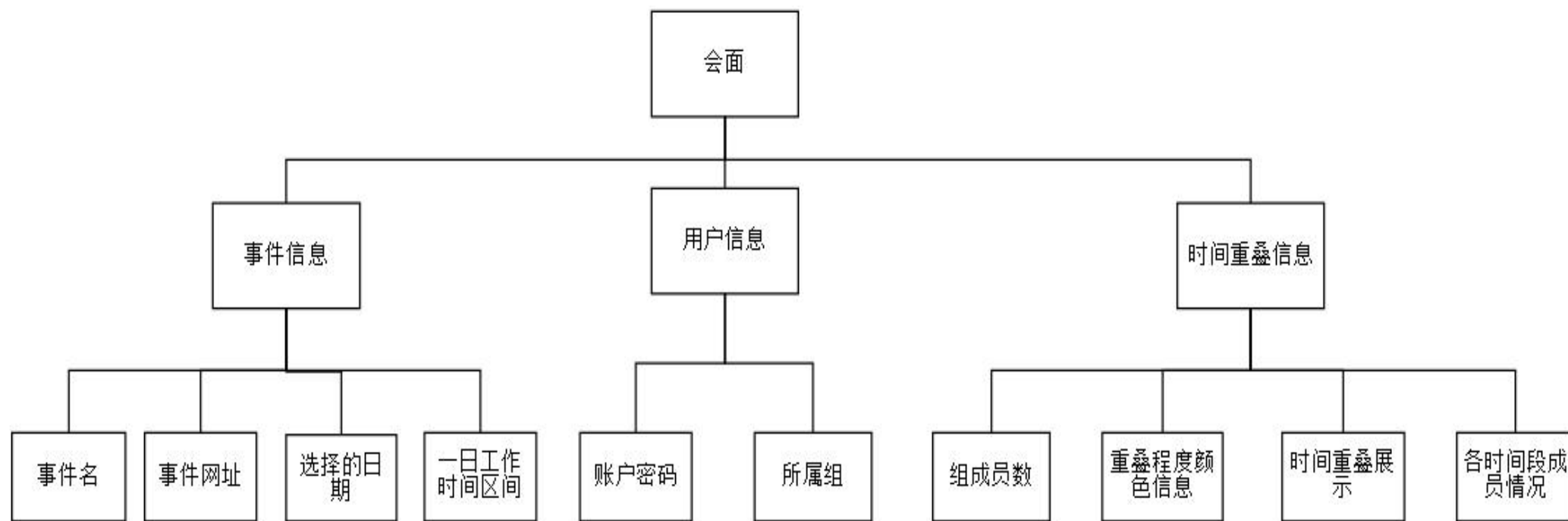
☀ 页面布局设计（线框）

☀ 关于我们页面：上下框架型



Task6:Web应用设计#内容设计

☀ 信息架构图



THANKS