**武汉大学计算机学院**

**本科生课程设计报告**

**ChatForReading**

专 业 名 称 ：计算机科学与技术

课 程 名 称 ：软件构造基础

指 导 教 师 ：贾向阳

学 生 学 号 ：2021302111506

学 生 姓 名 ：蒋硕

二○二三年六月

摘 要

对于学习者而言，他们经常会进行学习方面的讨论，除了线下面对面的交流，当今得益于网络的迅速发展，我们越来越多地采取线上讨论的模式。一是可以将讨论的内容保存下来，二是可以跨越时空的限制，让讨论更加自由。

在我们的日常学习生活中，我们往往会针对某一本书，抑或是某些资料进行针对性的讨论，也就是说整个交流相比较于日常聊天，背景更加明确，即基于某些特定的资料进行专题性的讨论。同时还不得不考虑一种情况，即交流的群体对于某些知识内容并不是特别清楚，往往需要借助网络进行相关搜索。在讨论结束后，我们往往还需要进行会议总结，如此能更好地对讨论内容进行回顾。

针对以上需求，我们小组利用了C# web开发，SignalR通信技术，Mysql数据库，EF框架，ChatGPT接口与功能实现，对文本内容的自然语言处理等技术，开发了ChatForReading软件，场景限制在专题资料背景下的讨论，支持上传相关讨论的资料并提供语义化的内容搜索匹配。同时，考虑到如今非常火热的大语言模型的相关应用，我们利用ChatGPT的Web接口对整个聊天讨论提供更智能化的功能，如回答某些问题，让ChatGPT给出某些参考的资料，观点，同时通过限定会议讨论的开始与结束，将整个会议的内容进行总结概况，方便后续回顾。

本人在此次项目中除了共同实现基本的聊天功能外，主要负责搜索模块，ChatGPT模块，文件模块等方面的内容。在实现相关软件功能的过程中进一步加强提高了自己的软件开发能力，收获颇丰。

**关键词：**专题化讨论，大语言模型，C# Web开发，SignalR，Mysql，EF，ChatGPT，自然语言处理。

1. **选题动机**
   1. **选题背景**

对于学习者而言，他们经常会进行学习方面的讨论，除了线下面对面的交流，当今得益于网络的迅速发展，我们越来越多地采取线上讨论的模式。一是可以将讨论的内容保存下来，二是可以跨越时空的限制，让讨论更加自由。

在我们的日常学习生活中，我们往往会针对某一本书，抑或是某些资料进行针对性的讨论，也就是说整个交流相比较于日常聊天，背景更加明确，即基于某些特定的资料进行专题性的讨论。同时还不得不考虑一种情况，即交流的群体对于某些知识内容并不是特别清楚，往往需要借助网络进行相关搜索。在讨论结束后，我们往往还需要进行会议总结，如此能更好地对讨论内容进行回顾。

* 1. **选题意义**

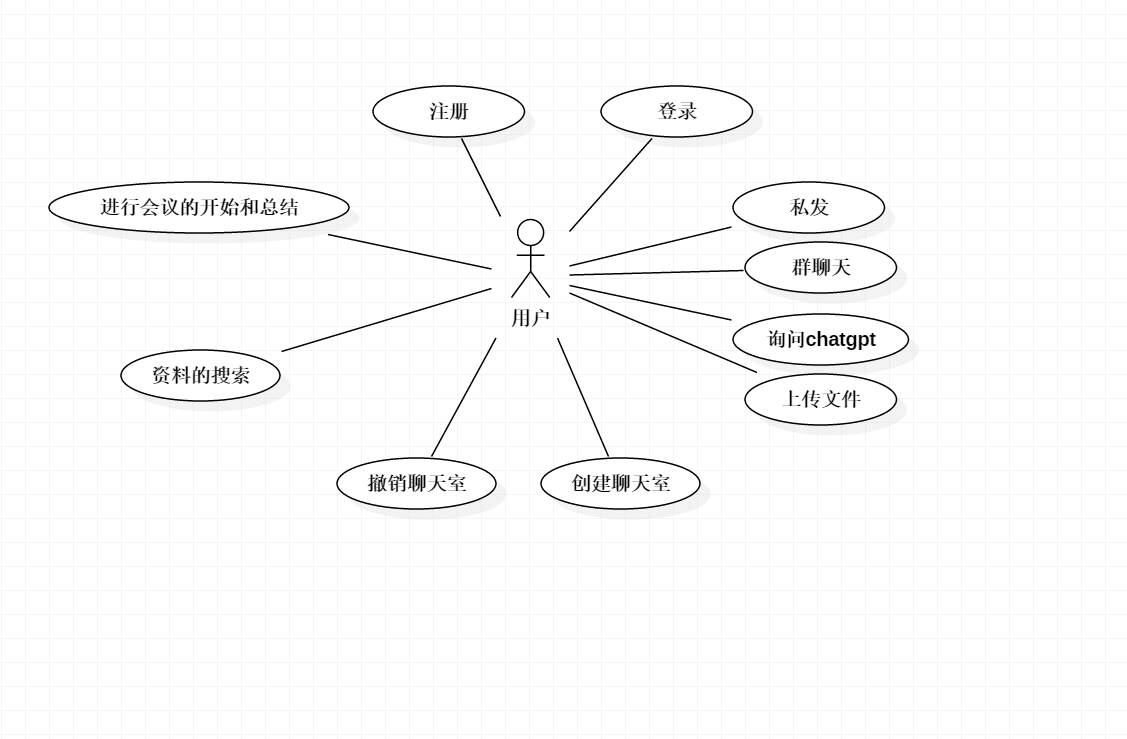
本项目瞄定学习讨论方面的交流聊天需求，考虑到该背景下的一些特殊限定和具体需求，如基于某些特定的专题资料，如会议总结，如相关问题的查询搜索等。我们进行了专项的优化契合，保障了相关功能的实现，在学习背景下很好了满足了相应的需求，能够很好地解决这个方面的难题。

* 1. **我承担的工作**

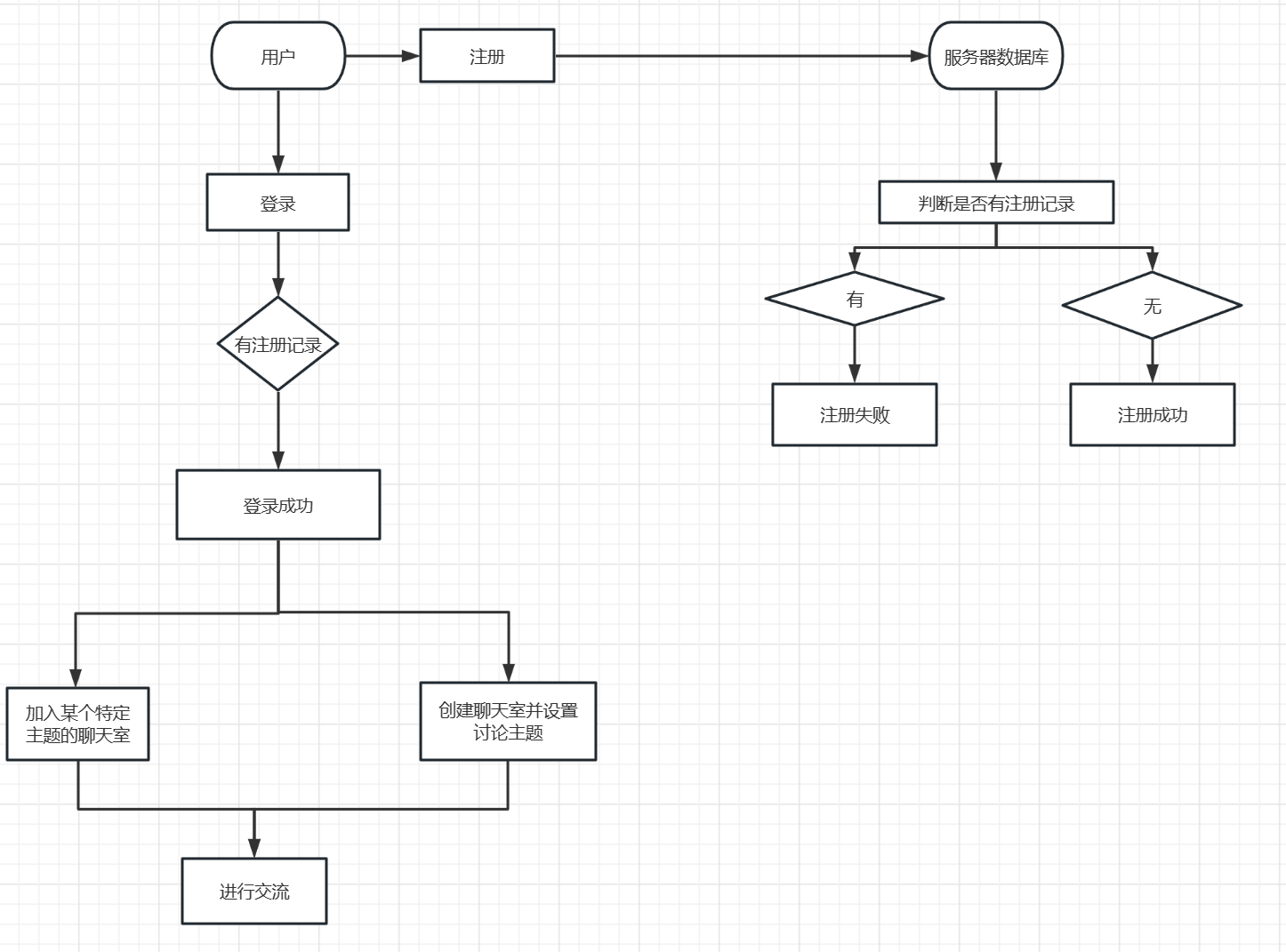
在此次项目开发的过程中，除了共同出力搭建基本的聊天框架外，我主要负责：基于特定文本内容的语义匹配搜索功能；文件存储位置的规划，以及在此基础上对ChatGPT模块的改进；最后就是前端页面中，针对每个房间上传的文档或者生成的会议总结的总览区域的设计，以及相对应的数据交互的部分，即相应JS的更改。

1. **软件需求与设计**
   1. **软件需求**

用户可以登录，注册，创建和管理群聊房间，上传文件和聊天内容，私发和群发，调用ChatGPT回答问题，会议开始与总结，资料的内容搜索匹配。



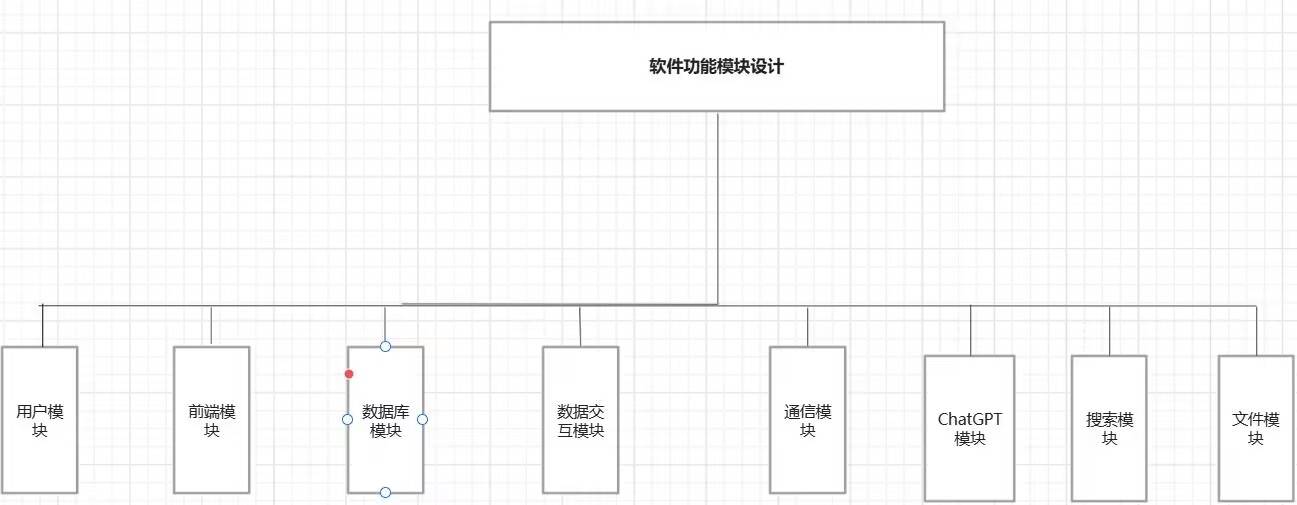
基本业务流程如下：



* 1. **功能组成**

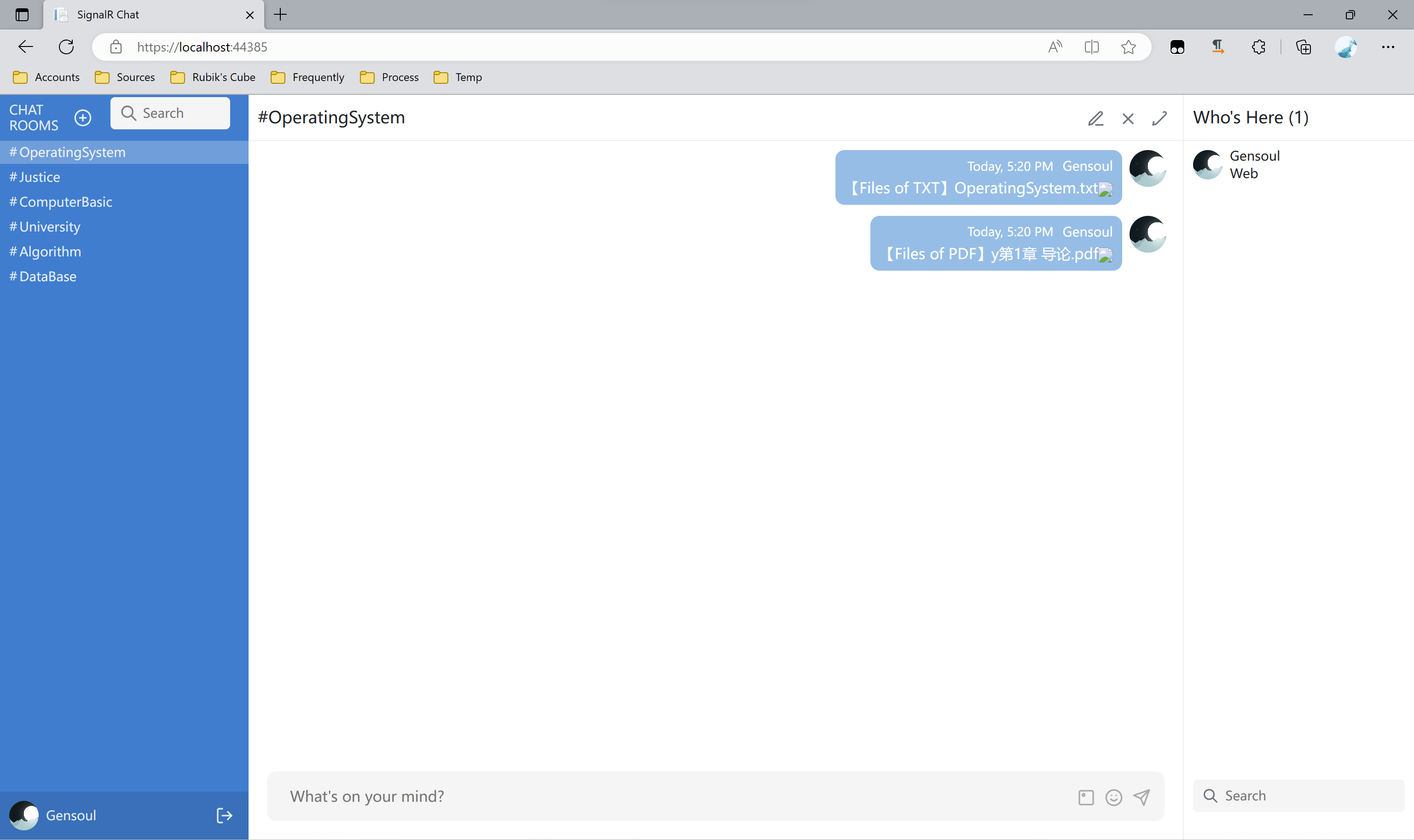
该项目主要分为如下模块：用户模块（登陆注册），数据库模块，数据交互模块，前端（界面设计+js），通信模块（signalr），ChatGPT模块，搜索模块（NLP），文件转化模块。

1. 用户模块：登录注册，验证匹配。
2. 前端模块：利用html和css对聊天界面进行基础的布局设计（同时后面拓展功能时需要添加了一些特定的布局和控件）以及一些动态数据的交互，利用JavaScript进行实时内容的更新。
3. 数据库模块：利用EF框架保存用户，房间，消息等数据（高级功能，如模型与视图的映射，数据库迁移技术）。
4. 数据交互模块：在WebController处对相关数据进行更改。
5. 通信模块： SignalR处理网络连接，并及时推送更新相关信息。
6. ChatGPT模块：利用web接口调用ChatGPT，实现相应的功能，如询问，会议总结，（包括了功能的改进）会议内容的保存。
7. 搜索模块：将上传的资料按照相应的规模分割，然后进行自然语言处理（借用ChatGPT生成词向量，Embedding）。在搜索时先在本地根据设置的阈值进行初步筛选，然后将相匹配的内容结合搜索请求传递给ChatGPT，生成基于语义匹配的搜索结果。
8. 文件转化模块：将上传的pdf转换为ChatGPT支持的文本文件。

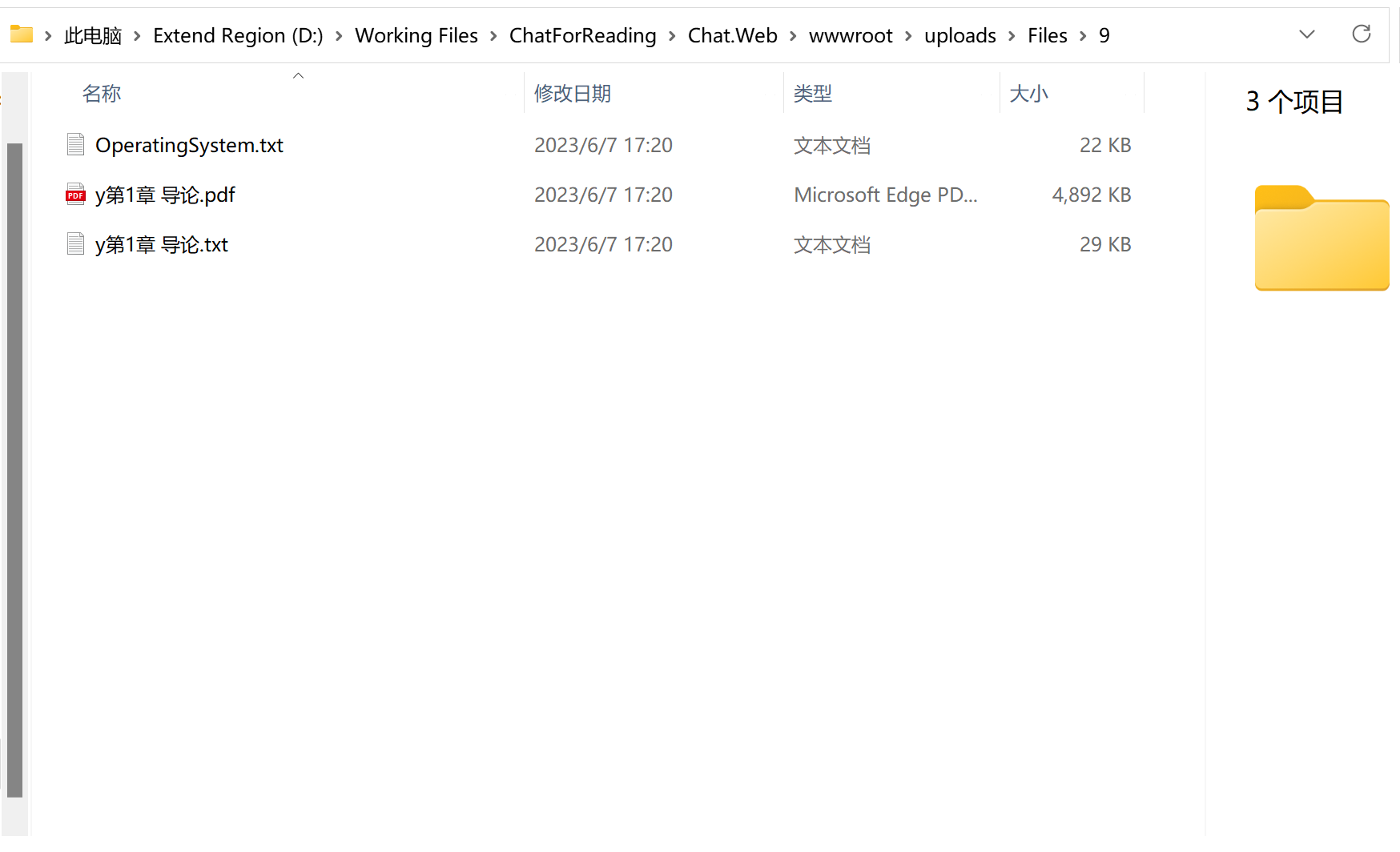


1. **关键技术**
   1. **基于特定文本内容的语义匹配搜索功能**

聊天室内可以通过聊天框传送需要的资料，目前支持pdf和txt两种类型。

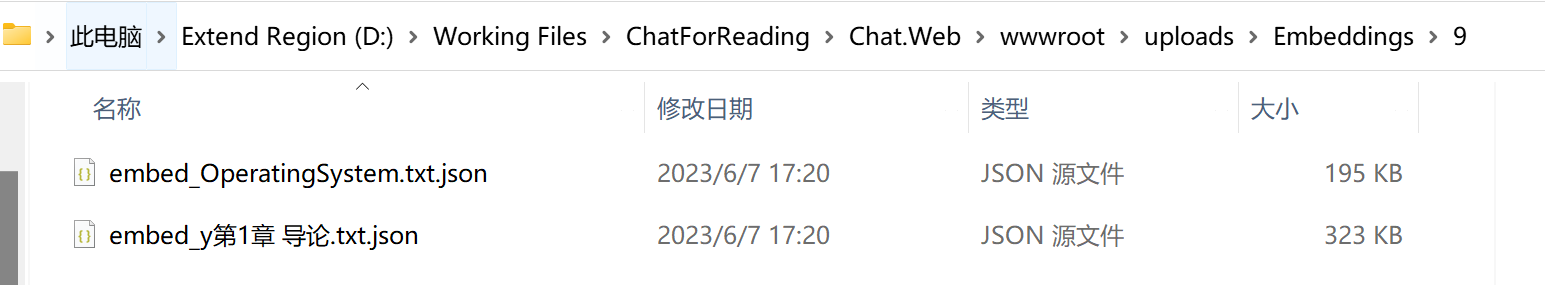


在设置的特定位置，即../Chat.Web/wwwroot/uploads/Files/(room\_id)处实现分房间的文件存储。对于pdf，我们在项目中通过“文件转换模块”将其文本内容抽出，形成可以进行下一步使用的txt文档。每次传输传输文档后，将会利用ChatGPT自动进行下一步Embedding文件的生成。



在../Chat.Web/wwwroot/uploads/Embedding/(room\_id)处存放经过自然语言处理后的词向量以及对应的文本，即json文件。在此项目中，我们是通过ChatGPT生成的Embedding，但是考虑到ChatGPT的api有token限制，所以我先是设置了maxSectionLength，用于将txt文档分割，分批次地通过ChatGPT的api进行处理，最后再整合起来放入同一个json文件中，如此便可以实现大文本的处理。



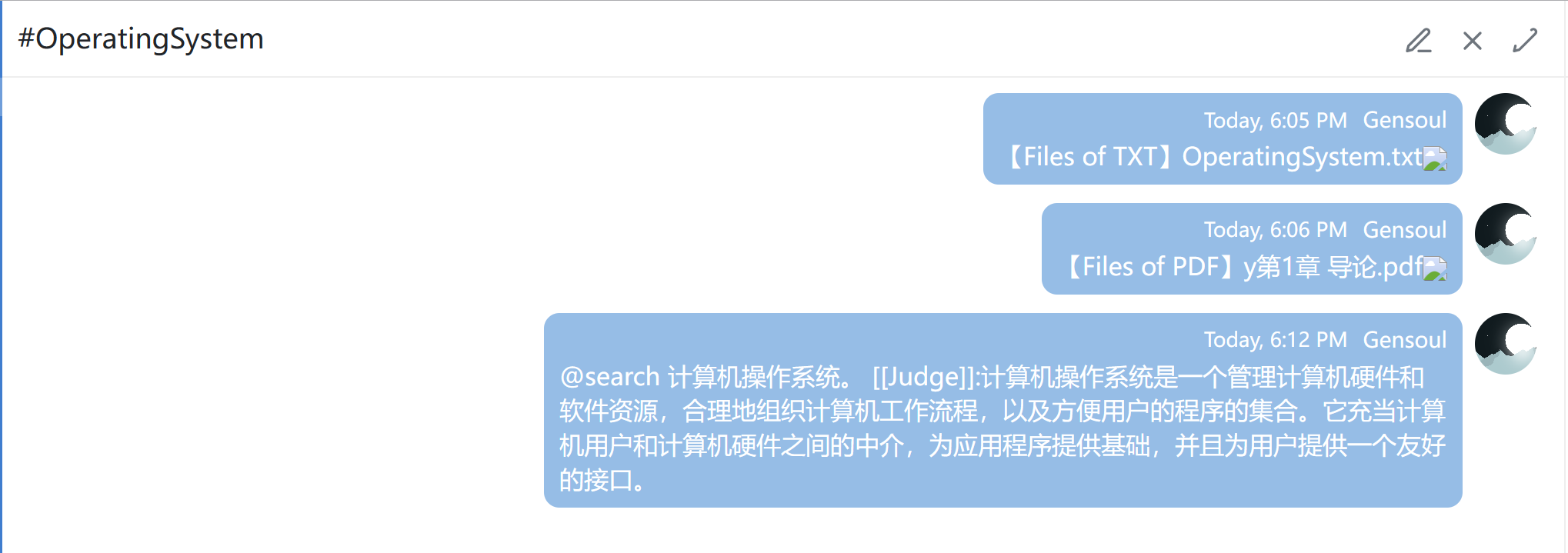


当输入关键字@search的时候，程序会自动识别，并在Embedding中先本地匹配（根据词向量的特性），初步筛选出相似程度较高的一些内容，并根据相似程度排序，通过本地初步筛选可以进一步减少调用ChatGPT的运行负担。



然后设定一个限制，将相似程度靠前的内容选择一部分，这里为8192，然后设置提示词，通过ChatGPT进行语义匹配并总结搜索的内容，最后只将ChatGPT的Judge内容显示出来，如此便实现了基于特定文本内容的语义匹配搜索功能。





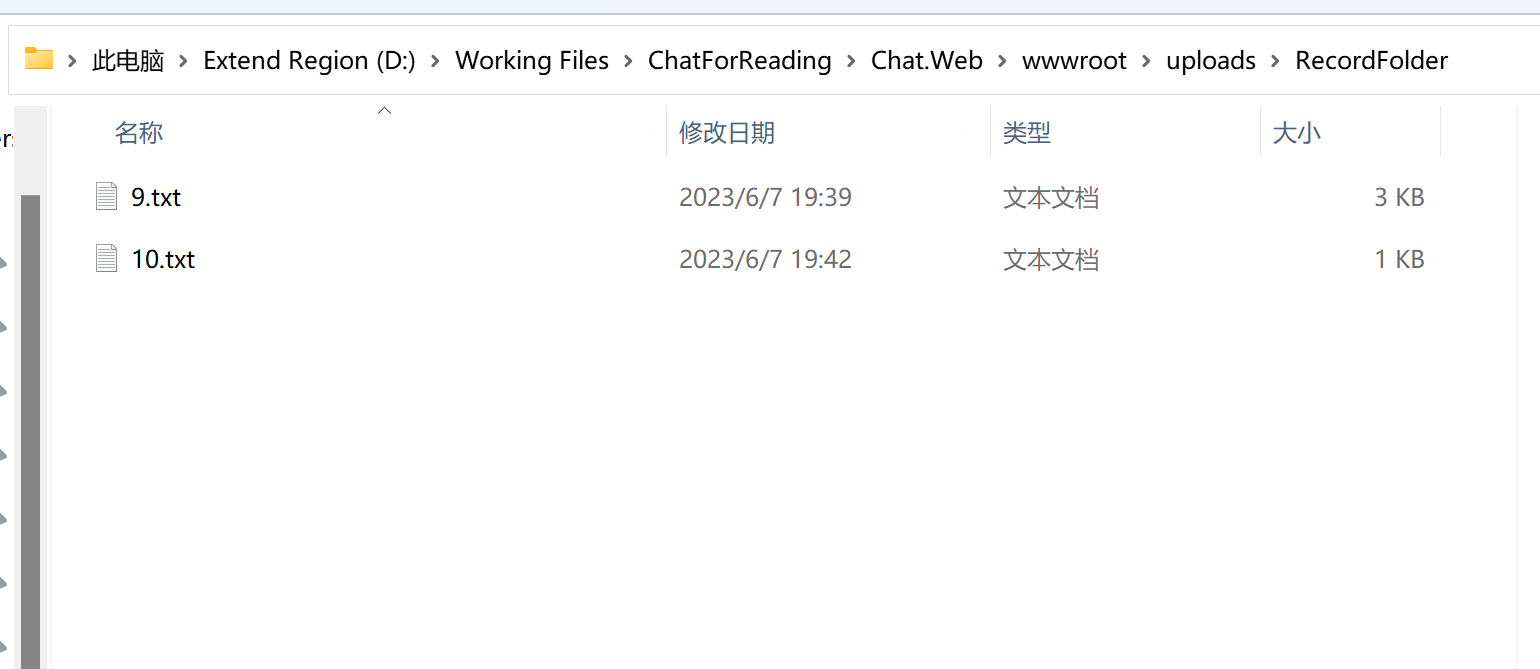
* 1. **基于ChatGPT的会议总结功能及其改进**

通过关键词@sum\_begin和@sum\_end，来指示会议的开始与结束，限定要进行会议总结的内容范围。

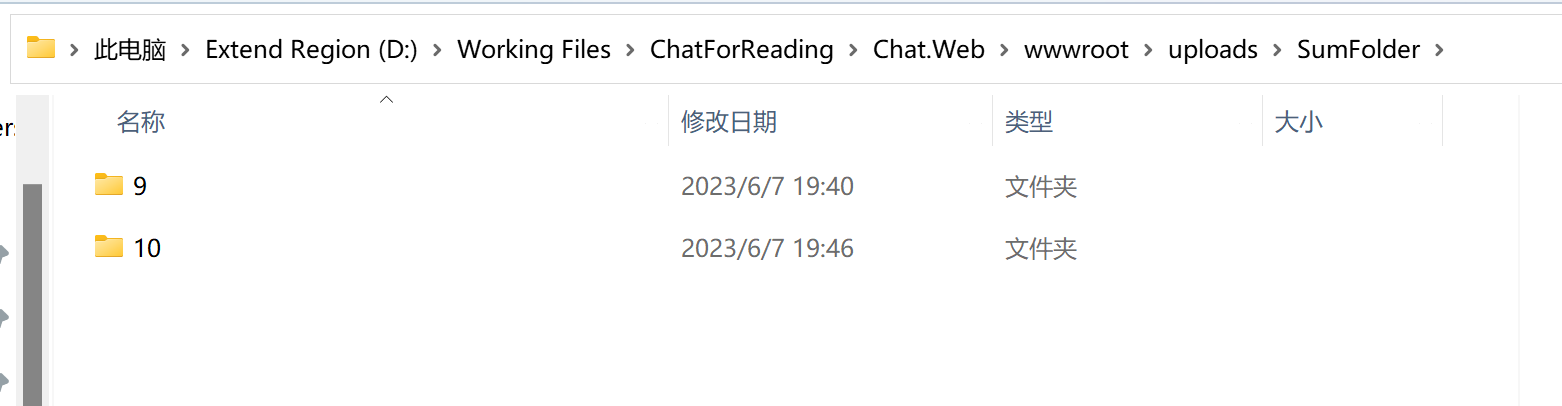
原本的实现方法是：利用一个全局的字符串，@sum\_begin后打开，将后续输入的所有内容通过一定的格式连接起来，在@sum\_end过后再将整个会议的内容集中发送给ChatGPT进行处理，返回会议记录。

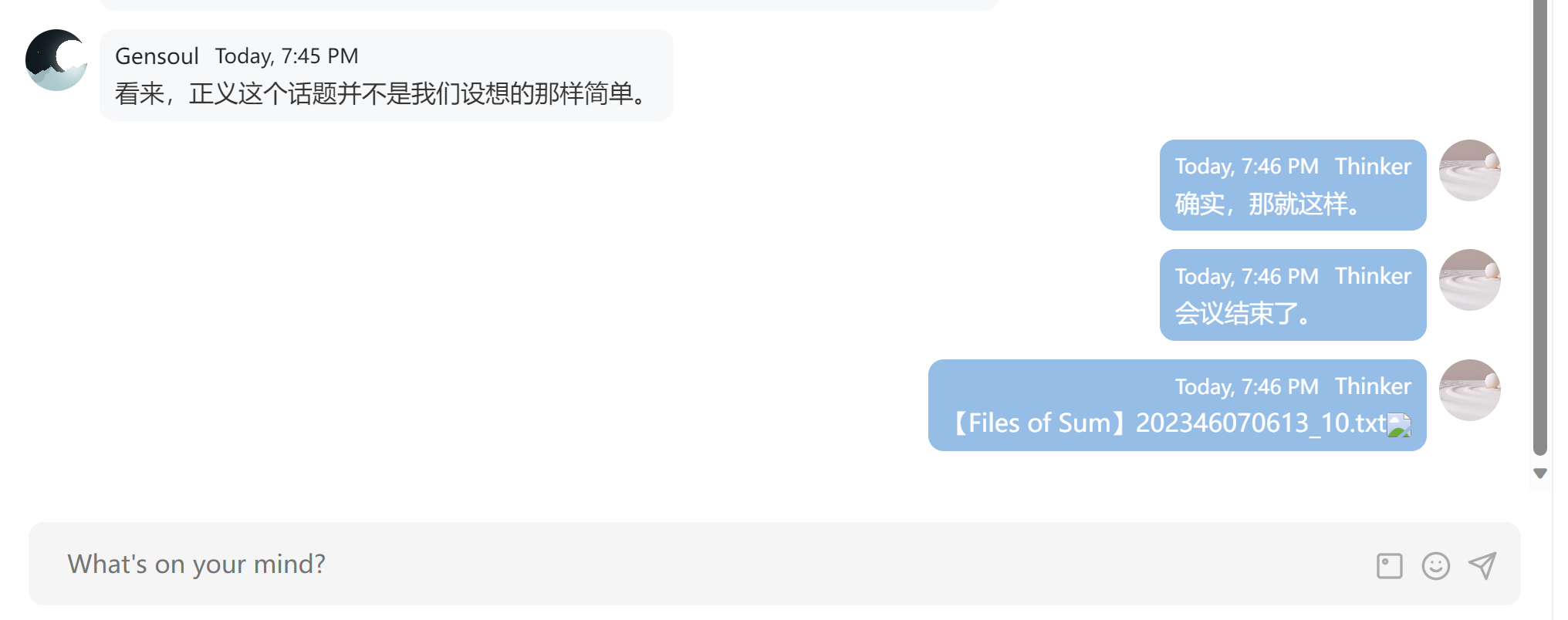
该方法存在的问题是，当多个房间同时有会议记录的请求时，该方法无法满足要求，在原本的基础上，我通过将会议的内容根据房间号存储在不同的文件中，实现不同房间会议记录的分离，彼此互不干扰，但每个房间每次至多只能有一次讨论，具体参考：

../Chat.Web/wwwroot/uploads/RecordFolder处，如图所示：



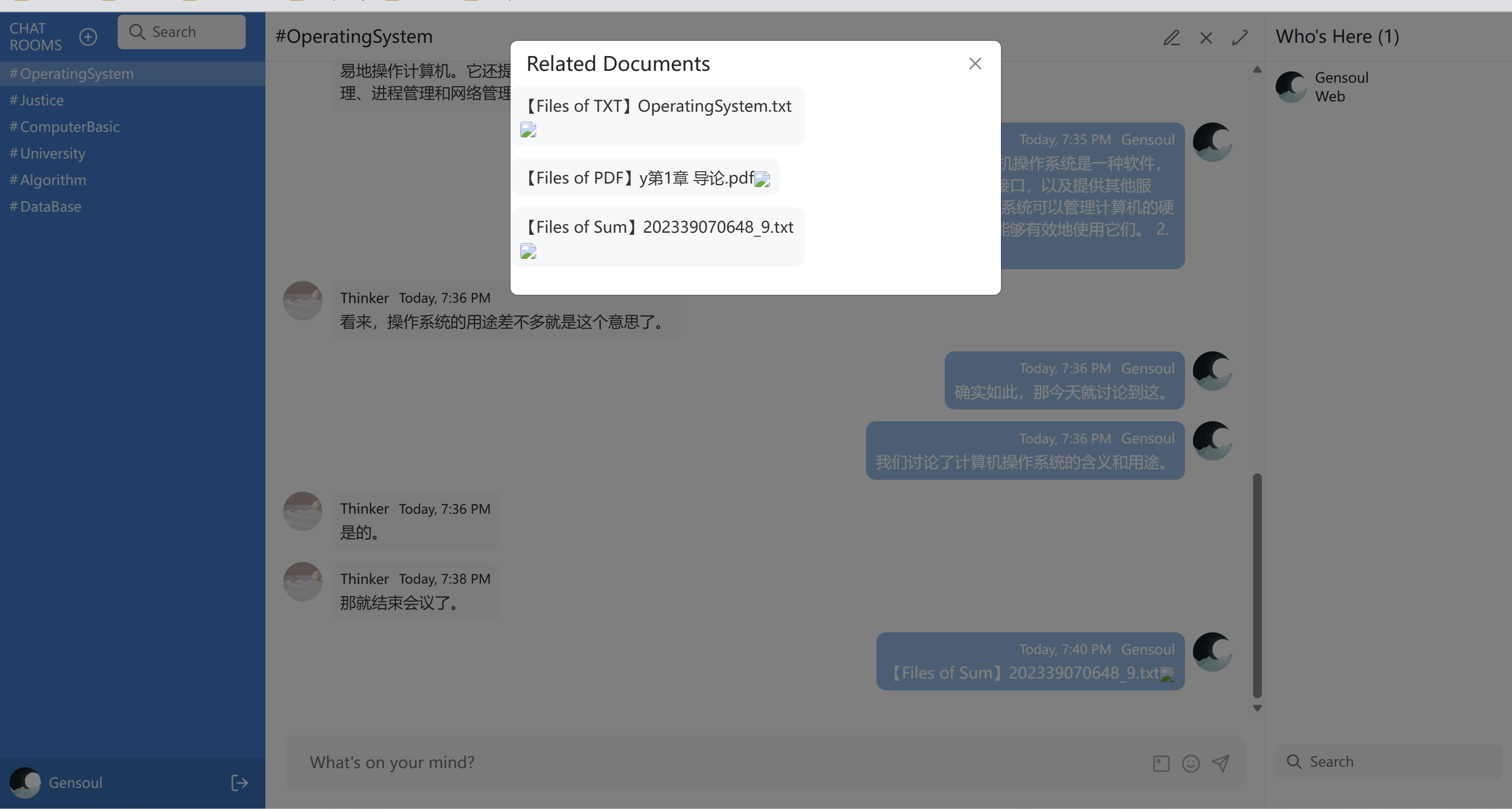
然后在../Chat.Web/wwwroot/uploads/SumFolde处保存了每一次的会议记录，格式为时间+房间号，同时在聊天框中会出现会议总结的文档，点击即可查看。



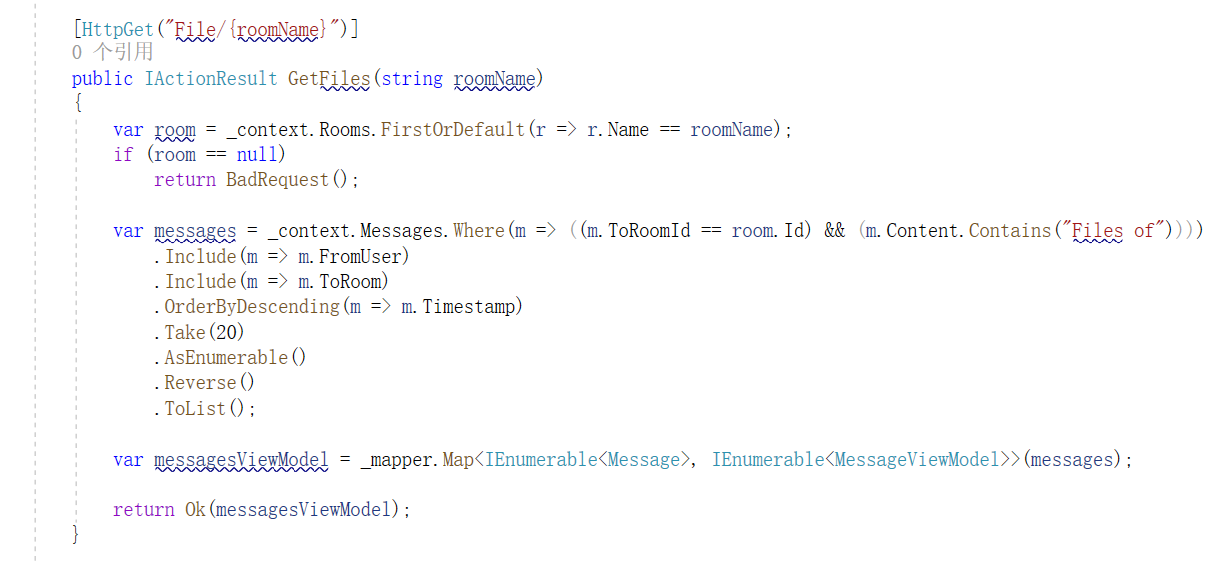


* 1. **基于JS技术的文件显示功能**

如图所示如图所示，在右上角的按键中是Related Document，即本房间中出现过的txt或者pdf文档，不仅有用户上传的文档，也有会议总结的文档，用户可以在这里迅速找到自己想要的资料。



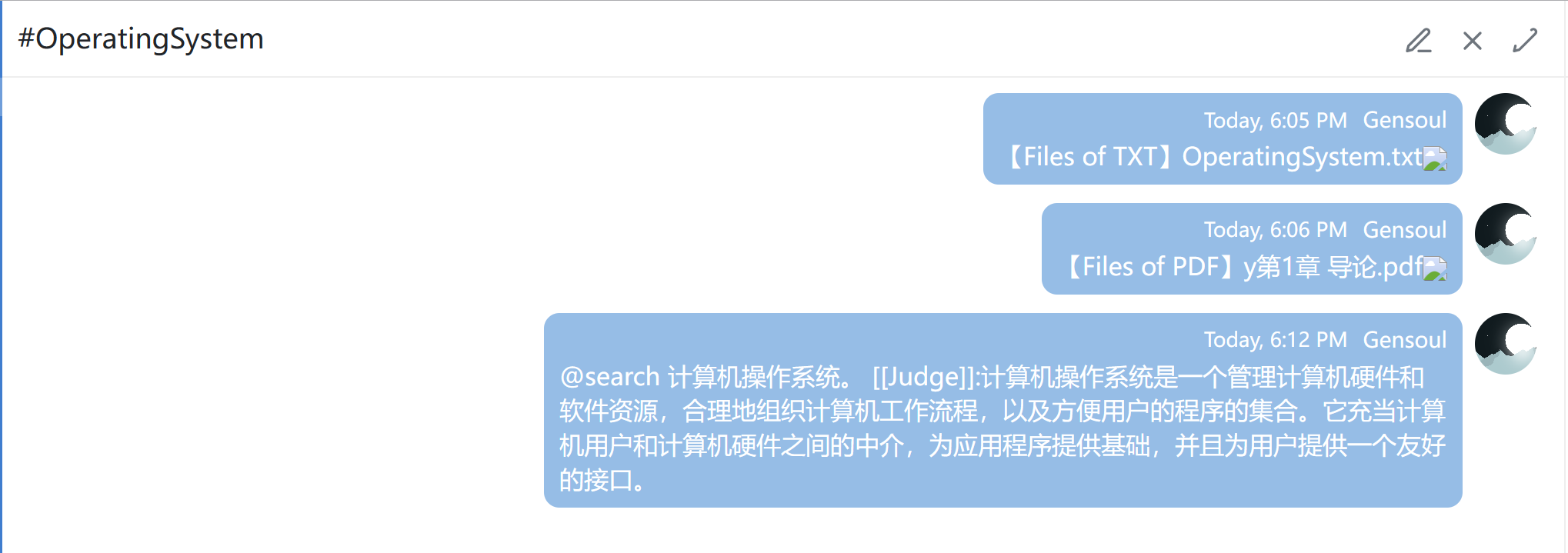
具体的实现为，在WebController中增加相对应的HttpPost方法，用于筛选出聊天中出现的文件（具体实现方法为，根据生成的文档信息的特点，即包含Files of，进行模式匹配），同时在chat.js中用相应的变量保存每次进入房间时生成的文件列表，对内容进行实时刷新，最后在html中将其显示出来。

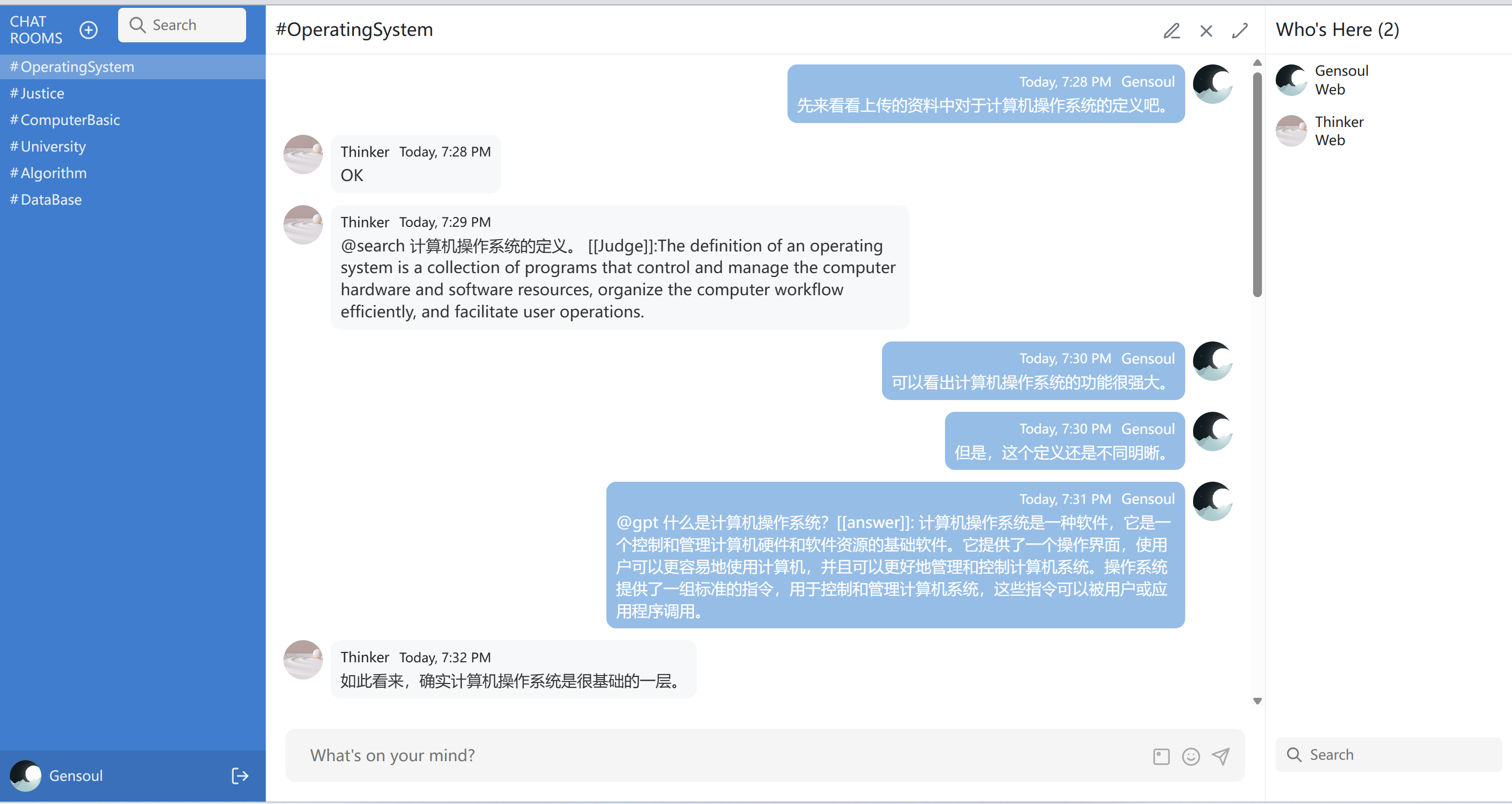


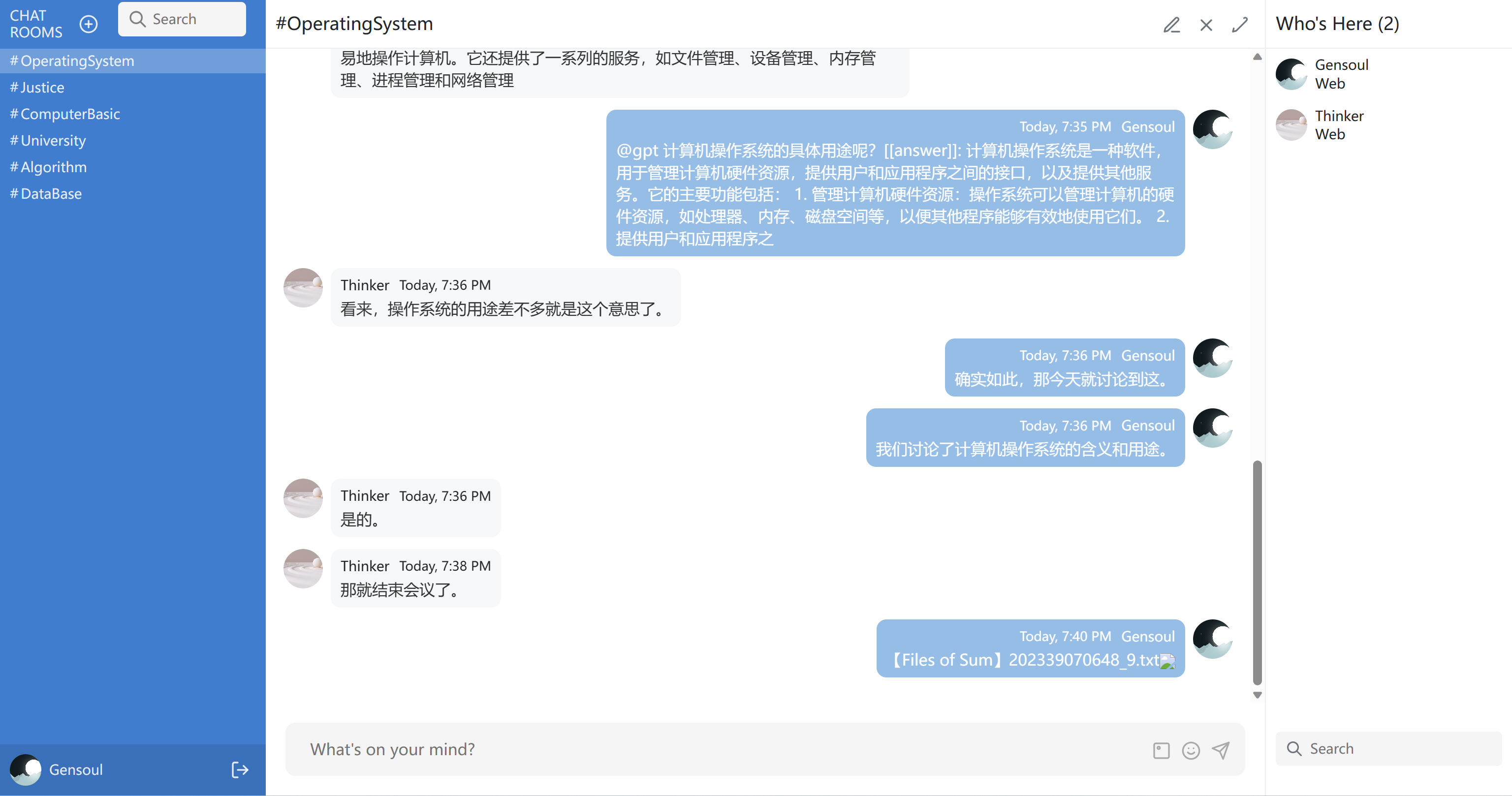
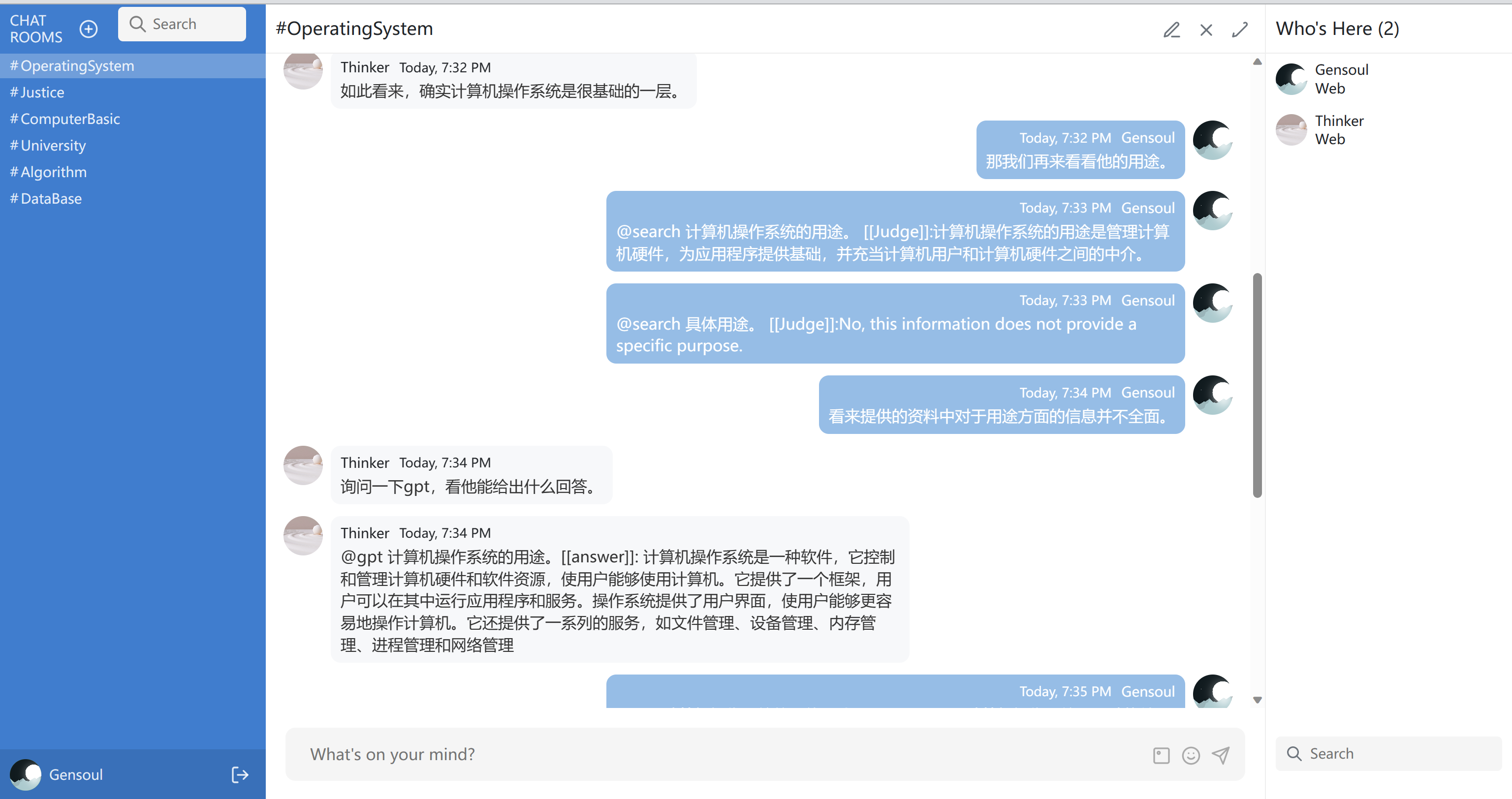


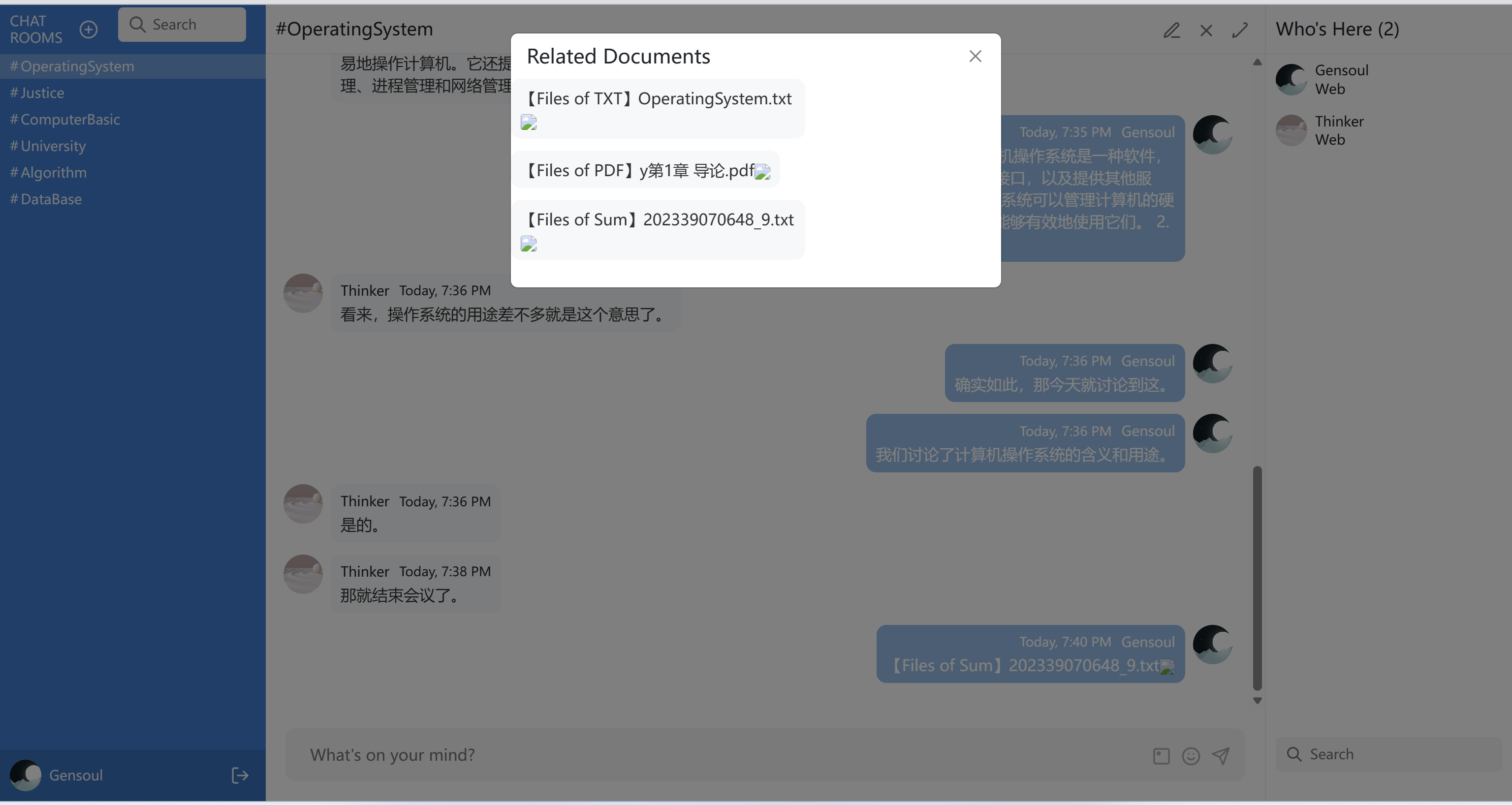
1. **案例分析**

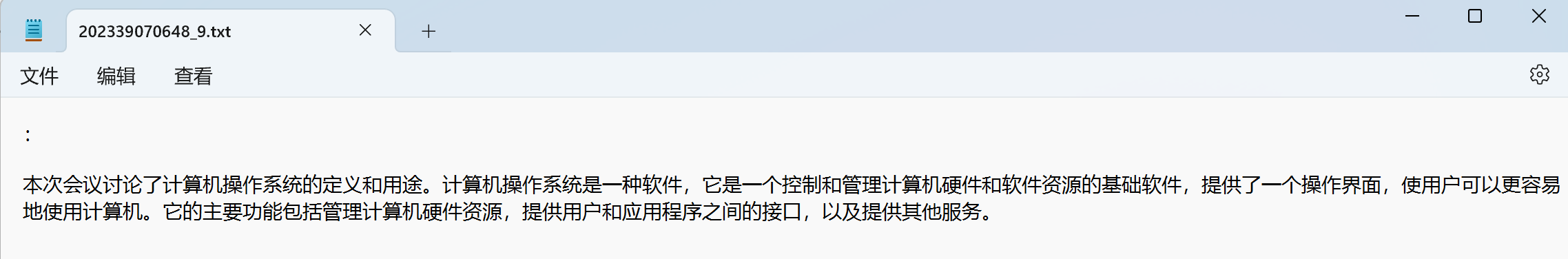
如下述案例，两位聊天者通过上传计算机操作系统的相关文档，借助ChatGPT的问答功能，和房间的会议总结功能讨论了计算机操作系统的定义和用途。











不难发现，在我们的测试案例中，该聊天软件很好地实现了本应该具备的功能，其中包括基本的聊天功能，上传文件，对文件进行语义查询，询问ChatGPT，利用ChatGPT生成会议记录并导出到txt文档中，罗列当前房间的文档等等。

在这个项目中，除了共同出力搭建聊天框架外，我主要负责内容查询方面的功能，会议总结功能的优化，以及房间文件的罗列等功能。

1. **总结与反思**

在此次项目实践中，我们从具体的应用需求分析开始，明晰了自己软件的具体定位，在开发过程中努力实现最初制定的功能标准，取得了阶段性的成就，收获颇丰。

此次项目，我在实现我所负责的一些主要功能的时候，深入理解代码和相应的知识，比如在实现文本转换（即pdf转txt），文本分割（将大文本按照一定的规模进行分割），文本向量的相似匹配，ChatGPT会议记录的改善等功能的过程中，我参考了很多现有的项目与框架，学习到了很多具体开发时才会接触的知识和内容，这也算是课堂以外对自己知识体系的一个极大的补足。同时在整个项目的实现过程中，我也更加深入地理解了基于C#的web开发应用，虽然仍然很粗浅，但是不可否认，相比以往，我的理解更加深刻。而且在有了此次项目实践的基础上，我对于后续的学习方向与路线有了更为清晰的认识。

经过这次实践，我极大地锻炼了自己对于问题的分析能力，代码理解与编写的能力，让自己在计算机软件开发这一方面有了很大的进步。本次项目实践极大地提高了自己在计算机方面的实践能力，同时也初步感受到了小组合作的氛围。

当然，本次项目也还有很多值得改进和反思的地方。在软件方面，我们还是显得经验不足， 在实现过程中没有规范自己的设计，导致许多功能过度耦合，扩展性极差，程序的整体架构和风格也极其臃肿。

由于缺乏实践，导致课堂所学的很多准则，知识都未能考量与使用。除此之外，在相关类以及功能的设计方面，我们的处理也仍有所欠缺，缺乏缜密的考量，这是上面过度耦合情况出现的一个重要原因。

再从整体来看，虽然我们初步实现了自己预期的功能和目标，但是不可否认，这也仅仅只是演示层面的产物，距离发布，或者是公开使用还有很长的距离，本身也有很多方面需要从演示向实用方面改进，比如数据库的设计，数据安全性的考量，各个模块之间的解耦独立，独立模块的设计与封装，ChatGPT支持的不稳定性，文件保存相关策略的选择等等。

还有一点，我们在聊天的基础上实现的附加功能基本都依赖于ChatGPT，但是在实际使用的过程中，我们发现使用ChatGPT的接口存在使用条件苛刻，运行不稳定等问题，因此对于支撑该项目核心功能的ChatGPT（大语言模型），我认为需要考虑其替代者，或许可以通过部署本地的大语言模型来取代其相当一部分的职能，进而把ChatGPT的使用场景压缩，减少其使用频率。

值得一提的是，经过此次项目实践，我在有所收获的基础上，发现了自己存在的大量问题，为自己以后学习方向的选择提供了更详细的参考，有利于我后续进一步的规划。

**教师评语评分**

评语：

评分：

评阅人：

年 月 日

（备注：对该实验报告给予优点和不足的评价，并给出百分制评分。）