

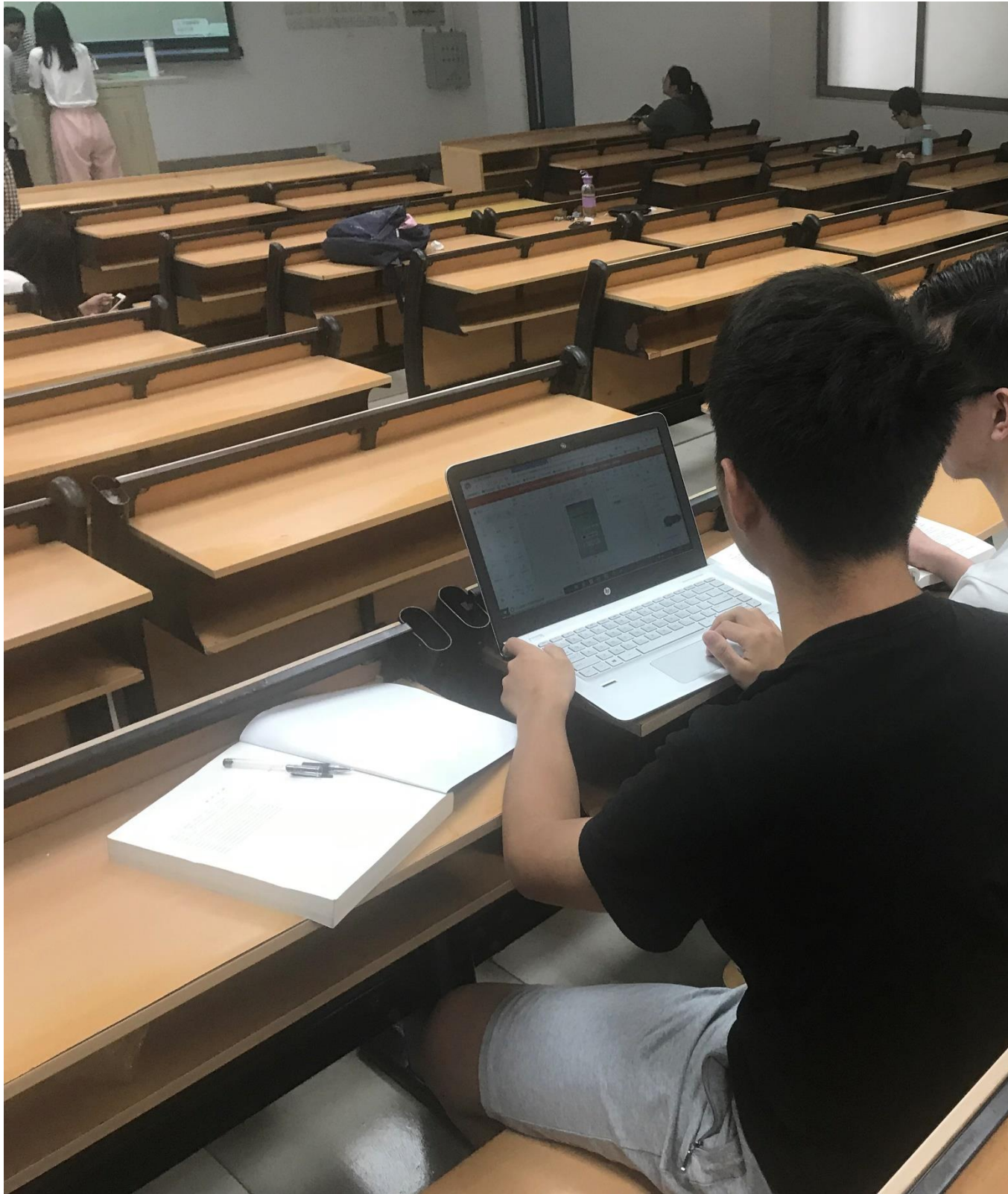
Link:

搭档链接: <https://www.cnblogs.com/littlenorthwest/p/9682472.html>

原型链接:

<https://modao.cc/app/QI7JghX5xX4xYvURiGTknu1InsS4J0R#screen=sB34D6855CD1537010429891>

Photo of Discussion:



Book Report:

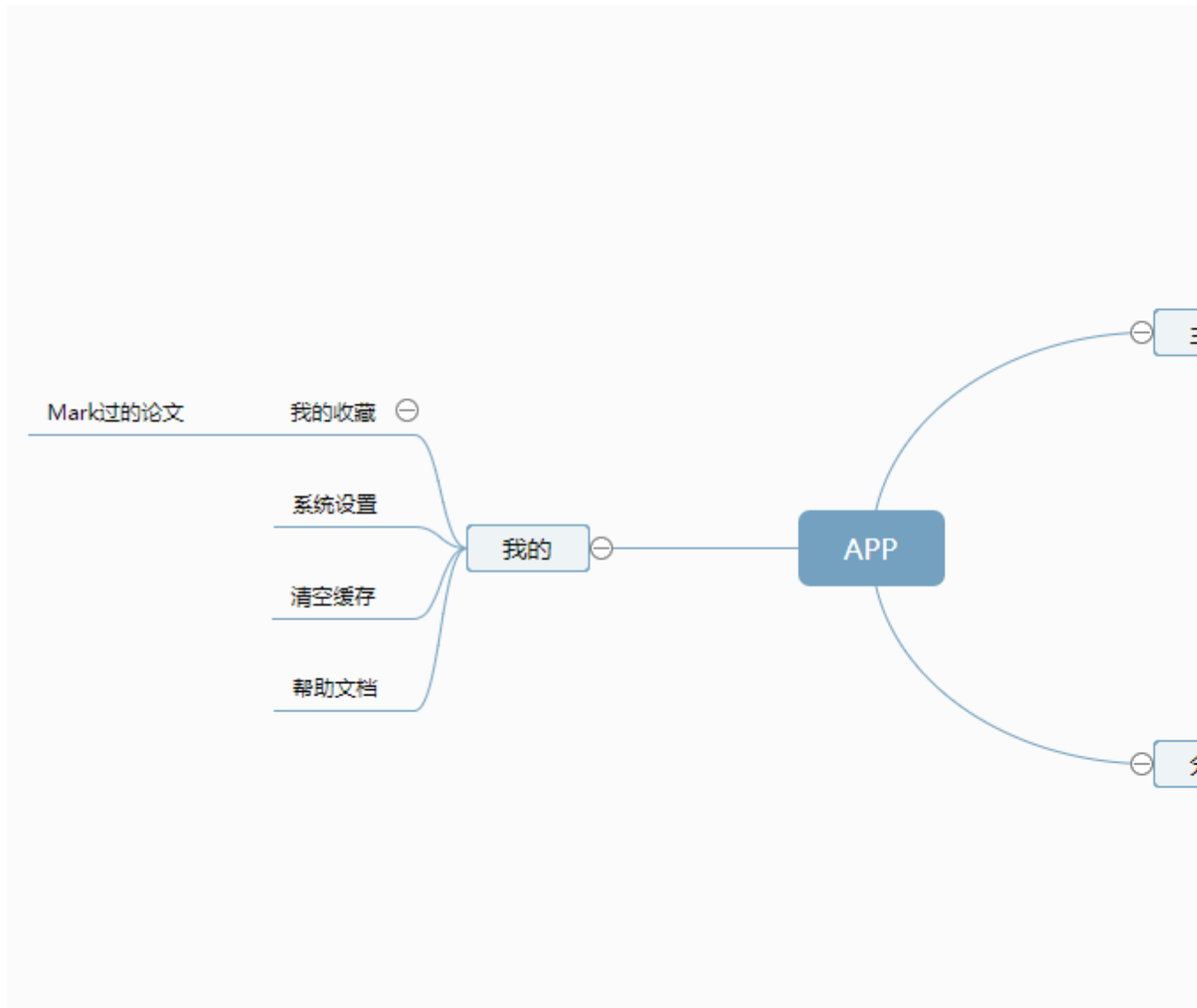
聚沙成塔的道理谁都懂，但是把小问题真正解决好，也是相当不容易的。作为一个软件工程师，不仅需要严格的对自己把关，不断地学习、完善自身的知识和能力、对自身的发展有一个明朗的认知、对自身能力有一个科学的认知，而且更应该学会如何融入团体，这一点是完成一个成功的软件必须的重要前提条件，一味的追寻待在自身所认为的“舒适区”，不愿意融入集体，所得来的后果往往是致命的，对于软件的研发而言是有害的，对于自身的职业发展也是不利的！知道作为一个软件工程师应该学些什么、怎么去学，以及自身的核心竞争力是什么也是十分重要的。

第八章：

这一章节，主要点在于 NABCD 模型，从五个方面去分析软件设计的各种层次要求，当我们接手一个项目时，我们可以按照此模型，分析用户的需求，依次列出我们的想法，从而能够十分科学地去说服客户，同时也能够做到有理有据，能够让客户理解我们的初步策划，十分的直观和高效！能够展现给客户我们十分有条理，会显得我们十分的专业！此外，我们还要考虑到这款软件给用户带来的便利与快捷，它能实质性地带给用户何种好处。同时，我们也应当着重考虑我们设计的产品竞争力在何处，用户为什么要选择他们的产品，此外，如何进行产品的推广与宣传也是必不可少的一点。从这五个方面分析考虑，做出来的产品才会有价值。

NABCD Model & Model Pictures:

APP 总体框架图



主页面：包含论文批量导入（通过文本文档文件），单个导入（输入论文题目）
查看已导入的论文以及本地论文检索。如下图

» 请选择您要导入的论文列表文本文件



Choose File

no file selected

请输入论文题目

添加



查看已导入的论文



本地论文检索





筛选



All



All



All



筛选结果

高等数学

详细信息 >



分析页面：包含对导入论文的属性分析，热门领域及热词分析以及相关数据统计。

More

近三年各顶会热门领域分析图

CVPR



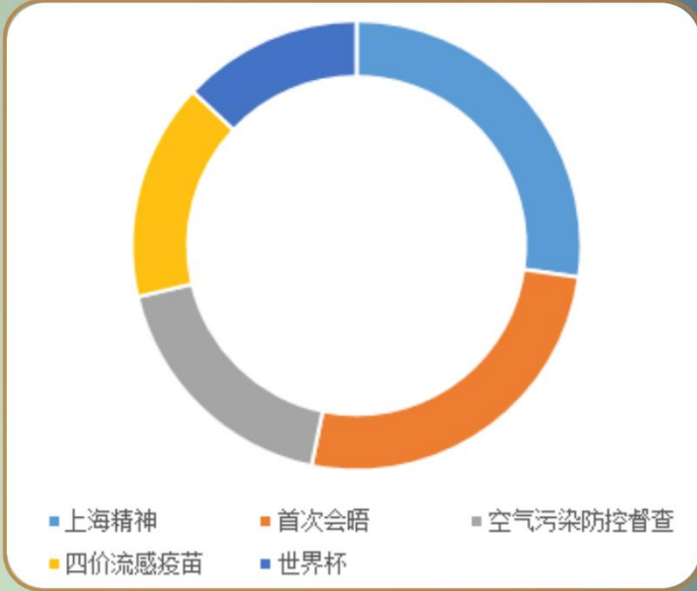
- QQ空间
- 梨视频
- QQ空间
- 百家号
- 网易号
- QQ公众号
- 抖音
- 一点资讯
- 腾讯号
- 飞信公众平台
- 微视



分析

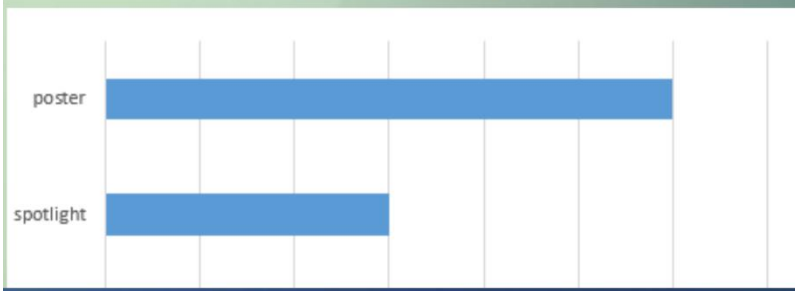
近三年各顶会热词分析图

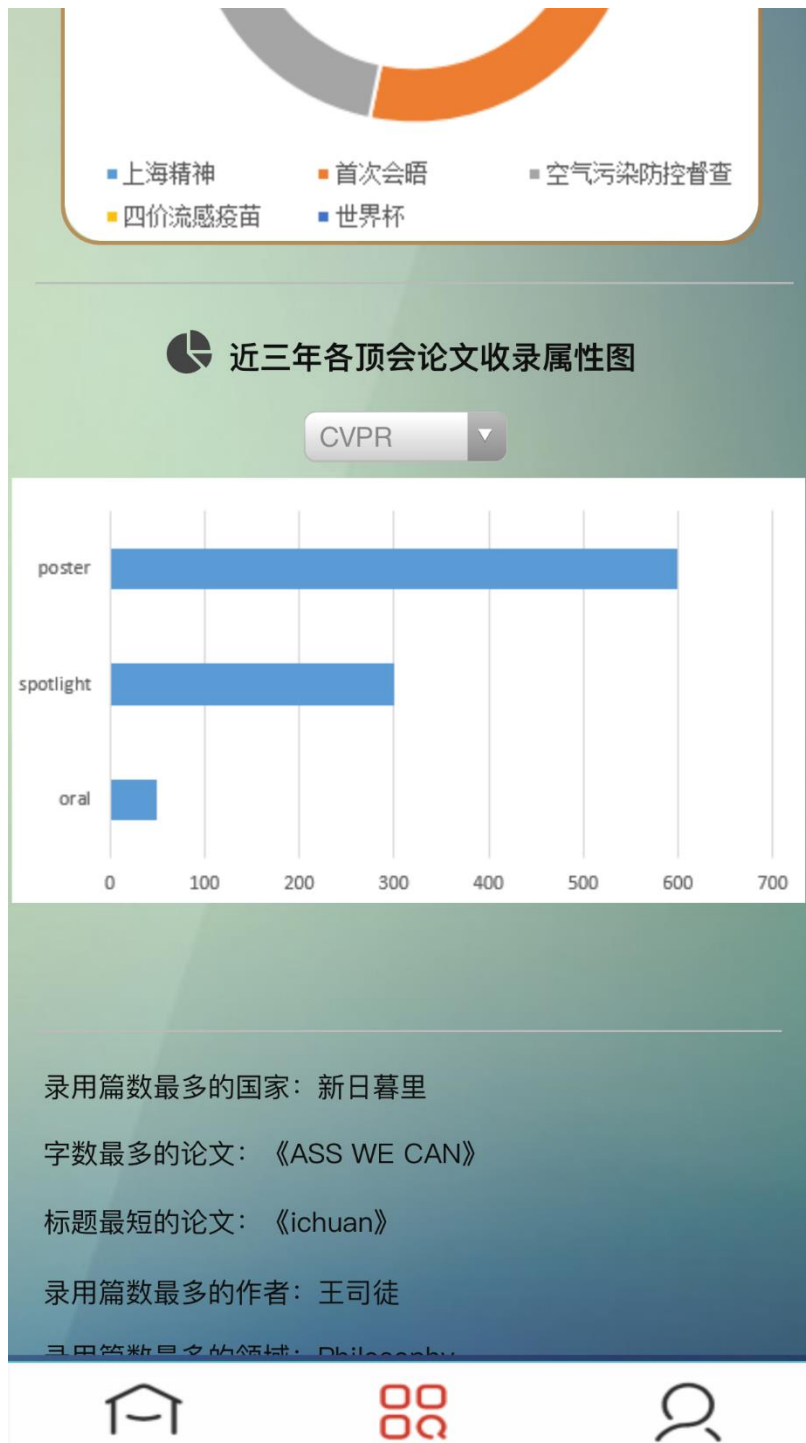
CVPR



近三年各顶会论文收录属性图

CVPR





我的页面：包含系统设置，我的收藏，清空缓存，帮助文档（其中主要功能是我的收藏）

我的

-  系统设置
-  我的收藏
-  清空缓存
-  帮助文档

More functions will be added.



我的收藏:



高等数学

详细信息 >



作者: mada...

编号: 875425

摘要:

在中国理工科各类专业的学生（数学专业除外，数学专业学数学分析），学的数学较难，课本常称“高等数学”；文史科各类专业的学生，学的数学稍微浅一些，课本常称“微积分”。理工科的不同专业，文史科的不同专业，深浅程度又各不相同。研究变量的是高等数学，可高等数学并不只研究变量。至于与“高等数学”相伴的课程通常有：线性代数（数学专业学高等代数），概率论与数理统计（有些数学专业分开学）。

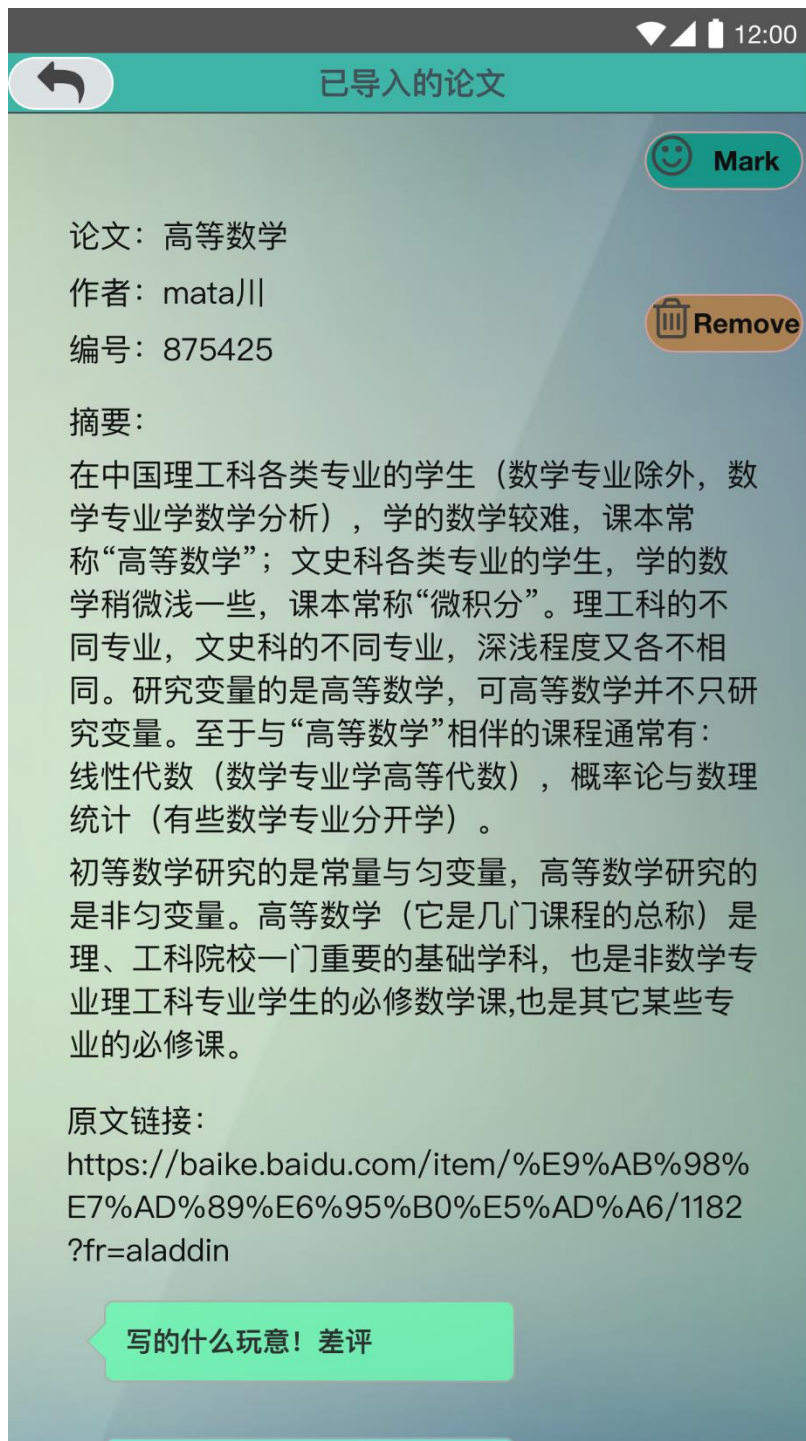
初等数学研究的是常量与匀变量，高等数学研究的是非匀变量。高等数学（它是几门课程的总称）是理、工院校一门重要的基础学科，也是非数学专业理工科专业学生的必修数学课，也是其它某些专业的必修课。

原文链接:

<https://baike.baidu.com/item/%E9%AB%98%E7%AD%89%E6%95%B0%E5%AD%A6/1182?fr=aladdin>

写的什么玩意！差评

呜呜呜一点也看不懂。。。



通过论文检索或者查看本地论文可以对论文进行 **Mark**（右上角），被 **Mark** 的论文将会出现在我的收藏中，此外也可以对论文进行评论（可以多条），同时，也可以在论文详细页面进行 **Remove** 操作，对当前论文进行移除。

Need（需求）——客户需求是什么？

*.简化流程，方便使用：

用户希望可以让通过论文列表找出热词或热门领域更加容易，同时操作简单。

*.快速查找原文。

用户希望可以通过论文名称、编号等论文相关信息查找出原文（即可以通过原文

链接查看原文。)及其论文的相关信息。

*.分析可视化。

用户希望能通过图表将诸如：热词、热门领域和对各校论文录用的情况以及各校更注重地领域十分清晰的表示出来。

*.论文数据分析。

用户希望能够对用户给定的论文列表进行分析，分析出热词或热门领域。同时可以根据对多年间、不同顶会的热词进行对比。

*.用户可给定论文列表

通过论文列表，爬取论文的题目、摘要、原文链接。

可对论文列表进行增删改操作(今年、近两年、近三年)。

*.对爬取的信息（即论文列表）进行结构化处理，分析 top10 个热门领域或热门研究方向

可对论文属性（oral、spotlight、poster）进行筛选及分析。

形成如热词图谱之类直观的查看方式。

*.可进行论文检索，当用户输入论文编号、题目、作者等基本信息，分析返回相关的 paper、source code、homepage 等信息

*.可对多年间、不同顶会的热词呈现热度走势对比（这里将范畴限定在计算机视觉的三大顶会 CVPR、ICCV、ECCV 内）

*.可进行数据统计，例如每个国家录用文章的分析、每个学校录用文章的分析、哪个学校哪方面的研究方向比较强等

Approach（方法）——要满足这种需求，我们能够提出什么独特的方法吗？

为了满足这些需求，我们制定了一款 APP，用于帮助用户快速的选取论文原文，以及快速的找出论文列表里的热词，从而使得用户能更加高效、方便地了解热门领域的动态。

Benifits（收益）——该方法给顾客提供的便利是什么？

主要有以下一些便利：

- *. 简化用户对于所需的查询论文的流程，将论文原文链接十分方便地提供给用户。
- *. 便于用户进行管理，方便用户对论文列表进行增删改查。
- *. 减少用户分析热门领域、论文热词的时间。
- *. 利于用户了解所提供的论文的整体走向。

Competition（竞争）——对于竞争对手和其他可选择的方案来说，优势在哪里？

*. 易上手，使用方便：

我们设计的 APP 对新用户十分友好，操作简洁明了，从而能使得用户极易上手，我们对此产品的定位是轻量级，即不需要太多冗杂的功能，浓缩即是精华。

*. 便于管理：

由于是使用 APP 的形式，因此用户可以直接通过手机来进行相关的操作，同时简明的分块和界面也使得管理起来更加的容易。

*. 界面精美：

我们参考了许多 app 的界面，并使用许多设计来美化我们的界面。

*. 功能丰富:

除了用户相关的需求之外,我们还适当地加入了一些人性化的功能,提高了APP的可用性。

Delivery (推广) ——如何把产品交到用户手中?

由于是小樱的需求,因此我们可以直接交给小樱。但是由于出现了小樱的需求,我们不禁可以联想到,我们将会拥有一些潜在的客户,因此我们可以通过宣传:在学校里开讲座进行宣讲;分发传单;微信、空间、微博、h5、贴吧等一系列平台进行推广;邀请使用者分享给身边的人也是一种好方法。

PSP Table:

PSP2.1	Personal Software Process Stages	预估耗时 (分钟)	实际耗时 (分钟)
Planning	计划	20	30
· Estimate	· 估计这个任务需要多少时间	10	5
Development	开发	600	800
· Analysis	· 需求分析 (包括学习新技术)	120	200
· Design Spec	· 生成设计文档	50	30
· Design Review	· 设计复审	20	10
· Coding Standard	· 代码规范 (为目前的开发制定合适的规范)	0	0
· Design	· 具体设计	120	150
· Coding	· 具体编码	0	0
· Code Review	· 代码复审	0	0
· Test	· 测试 (自我测试, 修改代码, 提交修改)	120	60
Reporting	报告	120	150

PSP2.1	Personal Software Process Stages	预估耗时 (分钟)	实际耗时 (分钟)
· Test Report	· 测试报告	20	10
· Size Measurement	· 计算工作量	10	20
· Postmortem & Process Improvement Plan	· 事后总结, 并提出过程改进计划	20	30
	合计	1210	1495

- 学习记录表

第 N 周	新增代码 (行)	累计代码 (行)	本周学习耗时(小时)	累计学习耗时 (小时)	重要成长
1	500	500	15	15	学习 VS2017, GitHub 使用, 复习 C++ 相关知识
2	500	1000	20	35	阅读《构建之法》, 从零开始学习 Java 语言
3	1000	2000	15	50	阅读《构建之法》, 学习 Java, 学习墨刀等工具使用

Experience:

困难:

- ①平台的选择, 我们最初纠结于做何种平台的软件, 考虑到便携性我们还是做了 app。
- ②在设计的同时, 我们还担心我们的设计是否在后期具有可行性, 以我们的能力是否能够将它实现。
- ③由于我们之前都没有做过 APP 这类项目的经验, 因此, 我们也不得不去学习一些制作 APP 的方法, 以及学习制作原型设计的方法。
- ④我们对这方面了解甚少, 包括相关的编程语言 (现学), 以及模型构造。

收获:

①学到了很多课内外的知识，如：**Android Studio** 的部分内容、**Java** 基础、原型设计工具的使用（如：**墨刀**）、**NABCD** 模型。

②了解了制作软件的大体步骤，以及如何去挖掘用户的需求。

③懂得了如何去深入地构思，如何去分配我们的工作时间。

④最重要的是，我们更加懂得该如何去融入一个小集体，如何去配合对方，如何协助对方进行高效、稳定的工作。