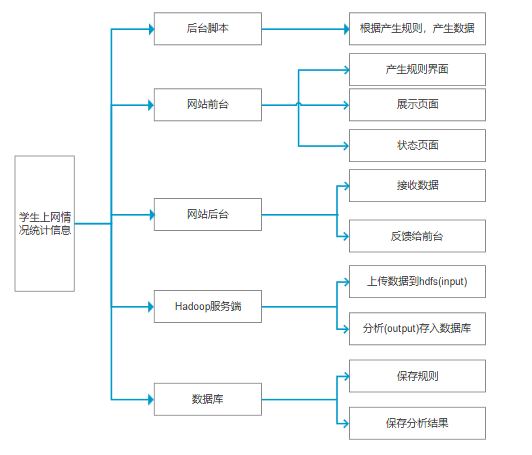
大数据项目

组员：车家竹 欧丽粤 姚文静 郑智强 赵尹苓 唐彬烈

系统具体需求划分



一、编码

1.将要预先进行的设计工作：需求分析设计：首先进行问题识别，分析细化，制定规格说明书后再进行评审工作，最后才能进行下一阶段；在键盘上完成的工作：搭框架，填代码（根据序列图、类图数据库进行编码）。

2.编码规范：（1）命名：变量命名采用类型前缀+有意义的单词组成，用小驼峰法。例如：aList；函数命名统一使用动词或者动词+名词的形式：init()，getName()；（2）注释：将有用的注释写的详细，但不要写一些可有可无的，显而易见的东西。比如描述一个类，一个方法的用途是什么，如果有重点或者难点代码，可以指出来。加注释时不要影响代码的可读性和美观。

3.异常错误处理：现在的编程语言大多使用异常系统作为错误报告技术，而不再是以往传统的传递和检查故障代码。现在的编程语言使用新的关键字来处理和捕获异常，其名称为throw、try、finally和catch等。通过正确地运用异常处理，我们能够使得软件更为强大。比如说catch能让我们捕获异常，并根据异常类型执行某种行为。

4.错误处理：

检测——语法、语义、运行

测试（灰盒测试、白盒测试、黑盒测试）

5.类型和接口的选择：优先采用类而不是接口；接口的缺点是语义变化时改变困难。优先使用抽象类而不是接口来解除协定与实现间的耦合。

6．大学阶段的我们在技术浪潮中的地位是属于学习阶段的，在这个阶段需要不断进步，拓宽自己的知识面，将自己学到的东西运用于实践，进一步深入了解。另外，深入一种语言去编程，而不是在一种语言之上去编程。其间的区别是，在一种语言之上去编程，思想受限于语言特性的限制；深入一种语言去编程，放在首位的是我们的思想，我们首先要确定我们的思想，之后，要用语言特性去实现它。不要将我们的思想局限于语言特性。无论是c，c++，c#，我们都可以编写出优秀的代码，这其中，最重要的因素，是你的思想，你的思路，而不是你使用的语言。

二、团队工作

1.Check in要求：

Check in之前做好静态测试和桌前检查；

（1）静态测试：静态测试包括代码检查、静态结构分析、代码质量度量等

（2）桌前检查：对源代码进行分析、检验，并补充相关文档，调整自己的代码符合编码规范

2.程序员是结对编程和独自编程的组合

结对编程的好处：1、互相鼓励，不容易沮丧：团队工作能增加成员的工作积极性。因为在面对问题的时候，会有人一起分担，共同尝试新的策略。2、互相监督，不容易偷懒：两个人一起工作需要互相配合，如果想偷懒去干别的，就会拖延工作进度。3、互相学习编程技巧：在编程中，相互讨论，可以更快更有效地解决问题，互相请教对方，可以得到能力上的互补。4、多双眼睛，少点 bug：两人互相监督工作，可以增强代码和产品质量，并有效的减少 BUG。

但独自编程也有相应的好处就是可以按照程序员想要的方式进行，所以两种编程方式可以结合起来，各取其长，达到高效率的目的。

三、质量保证

1. 程序员在编写代码之前，首先要为之编写测试用例，目的是提高代码质量，降低风险。通过等价类划分法、边界值分析法、场景法、错误猜测法等方法来编写测试用例，最后形成测试用例文档。

2. 程序员需要为自己的代码写单元测试，在编写代码之前先编写单元测试，会带来非常多的好处：例如：单元测试记录软件系统内部和外部的结构；单元测试帮助你和其他开发者迅速地看到是否“升级”的代码打乱了已有的代码。

3. 程序员在check in代码之前，要进行集成测试，集成测试是指在单元测试的基础上，将所有的模块按照设计要求组装成一个完整的系统进行的测试。一些模块虽然能够单独地工作，但并不能保证连接起来也能正常的工作。程序在某些局部反映不出来的问题，在全局上很可能暴露出来，影响功能的实现，所以在check in代码之前还需要进行集成测试。

4. 程序员要复审或者检查别人的代码。首先如果程序知道别人会看自己的代码，那么你编写的时候就会更加的慎重和仔细。你的代码会更加的整洁，注释更加完善，并且更有组织性。其次代码审查更大的好处是就是同事之间可以分享技术和编程方法。

四、工具

1. 我们选用的版本控制工具是svn，团队进行项目开发时，项目代码都储存在服务器上，成员可用svn在本地获得并更新代码。

2. 我们项目选择的编程语言是java 8。

3. 我们项目选择的编程框架是Spring框架。Spring是一个轻量级的控制反转(IoC)和面向切面(AOP)的容器框架，它的目标是为J2EE应用提供了全方位的整合框架，在Spring框架下实现多个子框架的组合，这些子框架之间可以彼此独立，也可以使用其它的框架方案加以代替。

4. 项目不使用非标准的语言特性。

5. 我们项目选定并且拥有编辑器、调试器、测试框架、语法检查器。