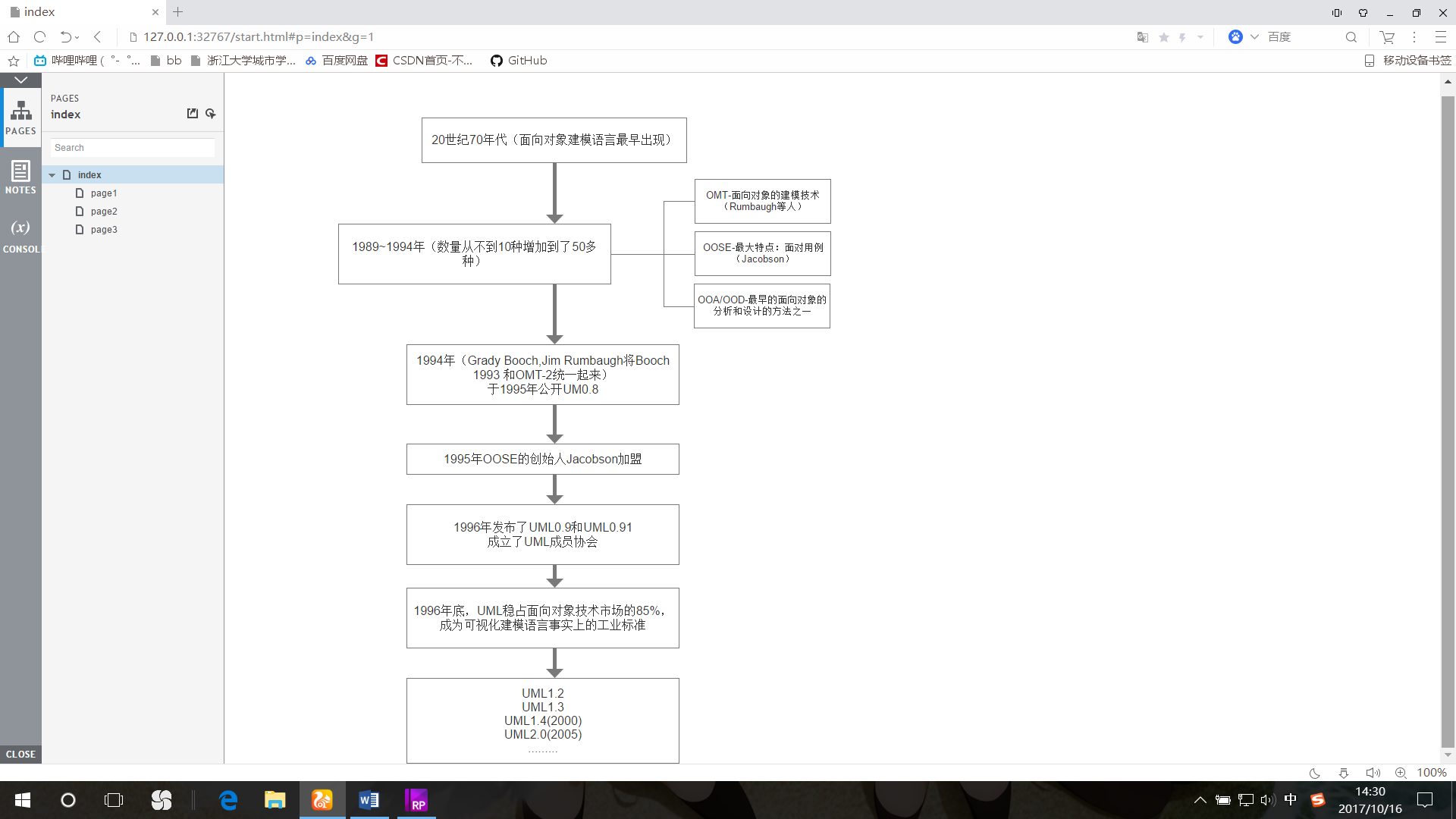
# UML语言

## UML的发展历程

UML发展历程简略图



# 软件建模概述

建模是为了更好的理解正在开发的系统。

软件建模是开发优秀软件的一个核心工作，是要把设计的结构和系统的行为联系起来，并对系统的体系结构进行可视化和控制。

要达到的目的

1. 有助于按照实际情况或安装所需要的样式对系统进行可视化
2. 规约系统的结构或行为
3. 给出了知道构造系统的模板
4. 对做出的决策进行文档化

# 常用的建模技术

《UML用户指南》p39-40

1 对系统的词汇建模

1. 识别用户或者实现者用于描述问题或者解决方案的哪些事物。用CRC卡和基于用况分析的技术帮助用户发现这些抽象
2. 对于每个抽象，识别一个职责集。确保能清楚的定义每个类，而且这些职责能在所有的类之间很好的均衡。
3. 提供为实现每个类的职责所需的属性和操作。

2 对系统中的职责分布建模

1. 识别一组为了完成某些行为而紧密地协同工作的类
2. 对上述的每个类识别出一组职责
3. 从整体上观察这组类，把职责过多的类分解成较小的抽象，吧职责过于琐碎的小类合成较大的类，重新分配职责以使每一个抽象合理的存在
4. 考虑这些类的相互协作方式，相应的重新分配他们的职责，是协作中没有哪个类的职责过多或过少

UML建模的技巧和提示

对最终用户或者实现这来说，各个类都应该映射到某个有幸的或者概念性的抽象。一个结构良好的类，应满足如下条件

1. 为取自问题域或者解域的词汇中的事物提供明确的抽象
2. 嵌入一个小的、明确定义的职责集，并且能很好的实现它们
3. 把抽象的规约和它的实现清楚地分开
4. 简单而且可理解，并具有可适应性和可扩展性

当用UML绘制一个类时，要遵循如下策略

1. 仅显示在该类的语境中对于理解抽象较为重要的类的特性
2. 按属性和操作的种类进行分组，以更好地组织其长列表
3. 把相关的类显示在同一个类图中

# 问题

什么是CRC卡？

CRC卡是一个标准索引卡集合，每一张卡片表示一个类。

Class-Responsibility-Collaborator

类名在最上方，类的职责在左侧，类的协作关系放在右侧。

简述面向对象开发的过程。

《UML2基础、建模与设计教程》p22

1. 系统调查和需求分析，分析问题并求解
2. 整理问题：对第一阶段的结果进一步抽象、归类整理
3. 程序实现
4. 系统测试

开放题

在我们生活中有什么可以利用建模来解决问题，并谈谈你对建模的理解。