

Software Engineering

码农的自我修养

孟宁

How SE is usually taught

- 需求分析：学生们都不懂企业的需求是什么，上课睡觉。
- 设计阶段：学生们画了许多 UML 图，用设计工具画了各种形状的图形，仅此而已。
- 实现阶段：学生们开始热烈讨论，UML 图早已经扔到一边。
- 稳定阶段：学生们中十分之一的人开始写代码，其他人不知道在干什么。代码大部分情况下都不能工作，所有设计过的种种黑箱和白箱测试都无从开始。
- 发布阶段：这个阶段只有一天时间，就是最后检查的那一天，同时还有人在调试程序。
- 维护阶段：课程结束了，同学们对自己的产品没有任何维护，放假了！
- 最后大部分同学们都说自己根本没学到什么本事，然后下个学期，新的一批学生进来重复这一过程...

How SE should be taught

- 需求分析：分析现有软件，用你们写的软件的用户量来证明你了解用户的需求
- 设计阶段：用快速发布来证明设计是有效的，能适应变化的。
- 实现阶段：用各种软件工程的衡量手段来证明大家实现的能力。
- 稳定阶段：证明测试能否覆盖代码的大部分。
- 发布阶段：如期发布，用户量，用户评价。
- 维护阶段：网上的观众或下一个年级的同学能很愿意接手你们的软件。
- 最后大部分同学们能说：自己做了一个有人用，有生命的软件。然后下个学期，新的一批学生进来提高这一过程...

现实 vs. 理想

- 需求分析：学生们都不懂需求是什么，上课睡觉。
- 设计阶段：学生们画了许多 UML 图，用设计工具画了各种形状的图形，仅此而已。
- 实现阶段：学生们开始热烈讨论，UML 图早已经扔到一边。
- 稳定阶段：学生们中十分之一的人开始写代码，其他人不知道在干什么。
- 发布阶段：这个阶段只有一天时间，就是最后检查的那一天，同时还有人在调试程序。
- 维护阶段：课程结束了，同学们对自己的产品没有任何维护，放假了！

- 需求分析：分析现有软件，归纳初步需求；基础差的同学学习技术
- 设计阶段：用快速发布来证明设计是有效的，能适应变化的。
- 实现阶段：用各种软件工程的衡量手段来证明大家实现的能力。
- 稳定阶段：证明测试能否覆盖代码的大部分。
- 发布阶段：如期发布，用户量，用户评价。
- 维护阶段：网上的观众或下一个年级的同学能很愿意接手你们的软件。

I want to teach SE this way

- 现实与理想之间需要一些适当的台阶
- Step by Step, Learning by doing
 - 在工作中学习 – 做中学, 在项目中有针对性的训练程序员的基本功
- 由近及远、由浅入深、多学多练
- 科学理论理解 VS. 工程能力训练

Legionary of II Augusta,
Britain AD 43



- About ancient Roman soldier, which of the following is correct?
 - a) No training, they were thrown into real battles right away
 - b) They only learn theory, don't touch weapon
 - c) They use fake weapon, half the weight of real weapon
 - d) weapon has **same** weight
 - e) weapon has **twice** the weight

I want to teach SE this way

- 码农的必备技能——Typing - VSCode - Git - Vim - RegEx
- 工程化编程实战——一个工程化C语言项目范例
- 需求分析与设计——从分析到设计的基本方法
- 软件系统设计——代码的结构、特性和描述方法
- 工程过程与项目管理——软件危机的前生后世

课程小项目+工程项目实验

为啥那么多动手的练习？

- 因为以前动手太少
- 没有量的积累，不会有进一步讨论和探索的基础，也没有掌握理论的条件。

陶艺课的故事，把学陶艺课的学生分成两组，一组以量评分（只看数量），一组以质评分（只看质量）。一个学期后，真正好的陶艺作品出现在哪里？

陶艺课

一个学期后，真正好的陶艺作品反而是出现在那以量评分的组里。

- 要敢于开始
- 所以，请停止抱怨自己实力的不济和各种客观条件的不好，停止对开始的恐惧。只要迈出了第一步，埋着头认真做，不停的尝试，重复，失败再尝试，只要不放弃，就一定能成功的。
- 阅读：

<http://www.cnblogs.com/codingcrazy/archive/2011/02/28/1967503.html>



为啥要做难的项目？

- 请先看视频
- <http://weibo.com/tv/v/F6Udt9Say?fid=1034:8003d777aa3f955d0a2d22dfe63bf02e>
- “老师， 为什么这些软件工程技术在我的项目中没有什么用？”
- 因为你的项目水太浅！

I want to teach SE this way

- Step by Step, Learning by doing
 - 在工作中学习 – 做中学，在项目中有针对性的训练程序员的基本功
- “教育是令人羡慕的东西，但是要记住：凡是值得知道的，没有一个是能够教会的。——奥斯卡·王尔德 (Oscar Wilder)”
- Learn from mistakes——Just do it!
- Collaboration with peer
 - 1:1
 - 1:many

大学教育 vs. IT行业培训

■ 大学：

- Preparing students for jobs that do not exist, using technologies that have not been invented, in order to solve problems we do not even know are problems. (Karl Fish)
- **课堂练习：请翻译上面这句话**
- 目的：打下基础，培养自我学习、探索的技能，为将来的未知挑战做准备。

■ IT 行业培训

- 用已有的技术解决现有的问题，获得实用技能
- 目的：人才马上能用

■ 大学当然可以讲目前流行的技术，解决目前的问题

- 目的是要知其然，并知其所以然，在其中锻炼自我学习，探索的能力。
- 要不断提问， 不断探索

■ 讨论：很多学校的计算机课程有很多“打代码”的练习

- 学生上机就是把代码输入，通过测试，下机
- 这是“大学教育” 还是“短期IT 行业培训”？
- 这种练习的效果如何？

What is Software Engineering (SE)

- Program = data structure + algorithm
- Software = Program + Software Engineering
- Software Company = Software + Business Model

What is SE in Waterfall Model

- 需求分析：分析现有软件，用你们写的软件的用户量来证明你了解用户的需求
- 设计阶段：用快速发布来证明设计是有效的，能适应变化的。
- 实现阶段：用各种软件工程的衡量手段来证明大家实现的能力。
- 稳定阶段：证明测试能否覆盖代码的大部分。
- 发布阶段：如期发布，用户量，用户评价。
- 维护阶段：网上的观众或下一个年级的同学能很愿意接手你们的软件。
- 最后大部分同学们能说：自己做了一个有人用，有生命的软件。然后下个学期，新的一批学生进来提高这一过程...

我们在实践中能逐步感知到的SE

- 码农的必备技能——Typing - VSCode - Git - Vim - RegEx
- 工程化编程实战——一个工程化C语言项目范例
- 需求分析与设计——从分析到设计的基本方法
- 软件系统设计——代码的结构、特性和描述方法
- 工程过程与项目管理——软件危机的前生后世

课程小项目+工程项目实验

“loser” 矛盾的行为和期望

- 父母，既看不起自己的孩子，又希望孩子出人头地；
- 孩子，既厌恶父母设计的人生，又怕走错路辜负了父母的期望；
- 老师，既要求学生乖乖听话，又希望学生有创新精神；
- 学生，既不认同老师的观点，又怕得不到那鸡肋的分数。

如何看待学习中的各种失败？

- 很多同学害怕失败，纠结于某个作业的分
- 请把学习看作是去游乐场玩耍
 - 不要纠结于一时的失误
 - 玩尽可能多的游戏
 - 希望最后还能学到点东西 😊

就像迪斯尼乐园，所有的小孩子即使摔倒了也不会哭。为什么呢，因为玩都来不及，要赶快爬起来去玩。如果你把自己的人生当作游乐场，你跌倒了就不会哭。

Text Books

■ 课本

- 《构建之法 – 现代软件工程》
- 《软件工程——理论与实践》

■ 必读书

- (必读) 《梦断代码》 <http://bestcbooks.com/dreamingincode/>
- (必读) 《人月神话》 Frederick Brooks

■ 其它参考书

- <http://book.douban.com/doulist/1204928/>