

从“死亡行军”到“零缺陷交付”【1】 估算的作用

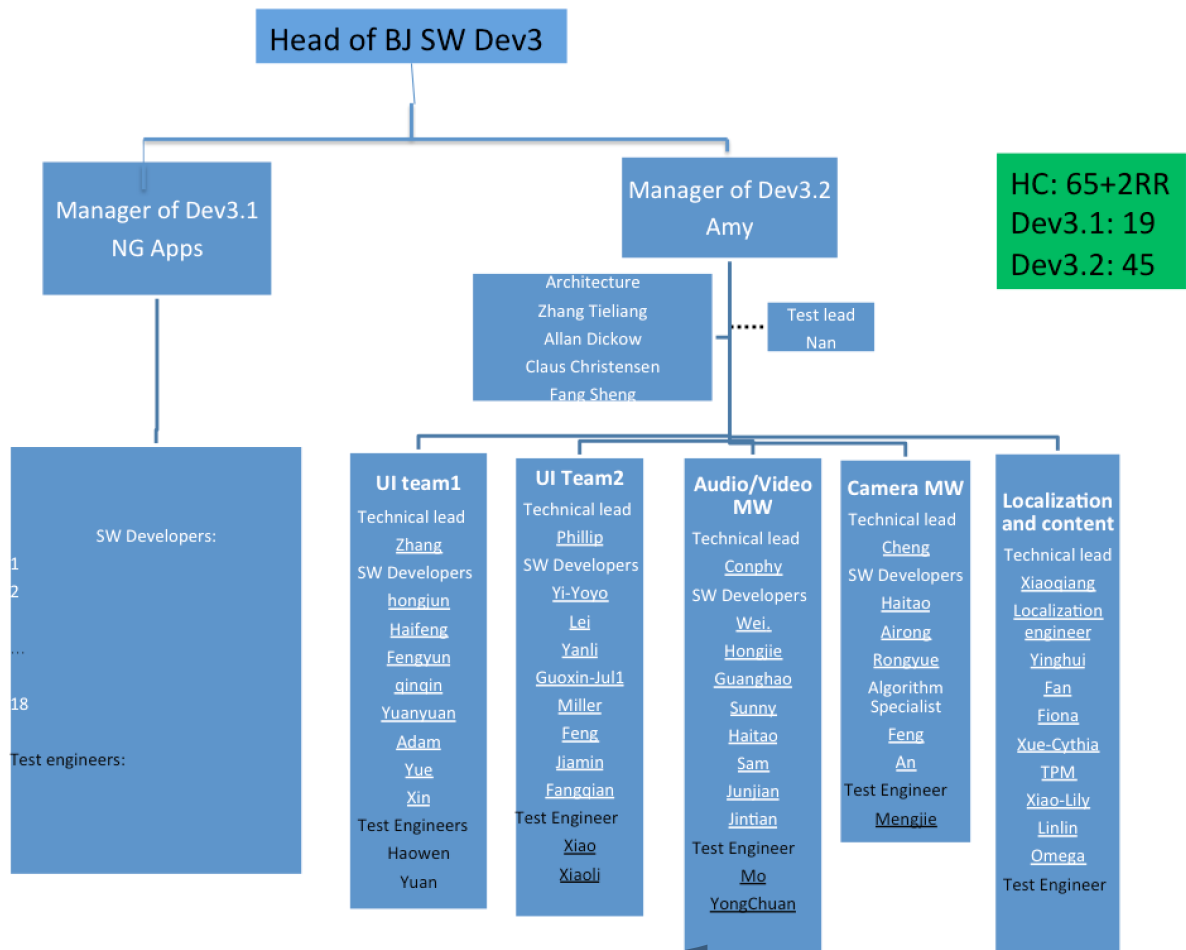
本故事发生在一个几乎已离开人们视线的知名公司的故事。它从一个伐木厂出发，转战纸箱生产，并在上个世纪九十年代走入电信设备制造行业，最后在移动大潮之时走上巅峰，也快速陨落的明星企业。没错，它就是“Nokia”。

案例背景

2012年初，我成了诺基亚（中国）“敏捷与精益顾问”岗位正式编制的第一人。到岗之后，才发现当时真正的组织状况。

尽管早在2005~2006年，Nokia网络就开始尝试引入Scrum，内部的倡导者Bas Vodde还提出了“Nokia测试”的一个评估清单，用于初步判断是否在该团队引入“敏捷教练”。然而，并不是诺基亚旗下的公司都是那么做的。Nokia生产功能机（用的是塞班系统）的部门从2009年才开始使用，而2010年底，才被引入在中国做功能机研发的团队。一夜之间，研发团队被分成数十个Scrum团队，原来类似于研发组长岗位的人都变成了ScrumMaster，而原来的产品经理就成了Product Owner。

整个功能机软件研发团队大约有500多人，下设多条研发线，每个研发线有一个line manager，每个line manager负责几个研发组，每个研发组都包括开发人员和测试人员。而产品经理属于Product Program Management Team，并不在Line manager的管理范围。下图是一个line Manager负责的团队结构示意图。



然而，似乎在北京研发中心引入Scrum开发模式并不顺利。虽然每个团队的运作都符合Scrum的迭代形式，而且专门有个团队负责持续集成服务器的环境维护，每天定时构建，而且运行一些自动化功能测试用例。但是，一个项目到了开发阶段的后期，仍旧出现很多集成问题，功能缺陷收敛很慢，项目进度和质量仍旧有很高风险。其实，这是进入了负向的ScrumBUT模式。

有一天我在等电梯时，听到了两位开发人员的简短对话，详细的对话内容记不清楚了，但大概的意思如下：

“你这是去开会吗？”
 “嗯，有一个关于新功能开发的讨论，我要去参加一下。”
 “咱们不是马上就要开回顾会议了吗？你不参加么？！”
 “哦，幸好有这个讨论会，否则又要到回顾会议上受罪去了。”
 “是啊，回顾会议感觉就是批斗大会！”

团队背景

我将要指导的Audio UI团队刚刚完成一次“死亡行军”。上个项目的开发时间为1个月，可是修复缺陷的时间却足足有1个半月的时间。团队成员都比较年轻，经验比较少，只有ScrumMaster在Nokia工作超过一年。其它开发人员和测试人员都不超过半年，有两个人还不到三个月。

新产品项目背景

马上要开始一款新手机的产品研发。这款手机产品一共200多人参与，整个产品的研发预计需要10个月的时间。其中，软件研发团队大约100人，根据系统的结构及所负责的模块不同，分成多个团队，如Audio UI团队，Vidio UI团队，Audio/Vidio中间件团队，Core团队，SMS团队和Phone call团队等等。一个Line Manager请我帮助她管理的团队，那就是Audio UI团队。

项目背景

作为一款手机产品的研发项目，其与互联网产品的研发有很大的不同。（1）项目历时长，近10个月；（2）同时参与的人员多，近200人；（3）质量相对要求高，因为手机售卖后若有问题，修复成本很高。另外，这个塞班系统已经有十年的历史，是一个拥有1500万行代码的遗留代码库。

项目前期准备

在我进入这个项目前，Product Owner（简称PO）及Scrum Master已经拿到了功能列表，而且这个功能列表已经进行了初始估算，这个估算用于初步确定开发工作量（如下图所示），其单位是“人周”。但是，团队成员仅知道要启动一个新项目的开发，并不知道具体内容，如下图所示。

360 ID	Prio	Epic	User Story	EE (man week)	EE-UI (man month)	Status
30		Easy Editing - Audio	[DRM] As an user, I'm not able to modify DRM protected file.	1		Not started
50		Easy Editing - Audio	[Menu] As an user, I'd like to see an option to clip my music/voice file.	0.5	0.25	Not started
51-60		Easy Editing - Audio	[Container] Editing mode UI.		2	
57		Easy Editing - Audio	[Editing mode UI] As an user, I'd like to enter the edit mode by choose "Clip" from option menu.	2		Not started
54		Easy Editing - Audio	[Editing mode UI] As an user, I want to see the music file name at background in the editing mode. (MO) - Need to rewrite MO.	4		Not started
52		Easy Editing - Audio	[Editing mode UI] As an user, I want to see the progress bar in the editing mode.	0.5		Not started
53		Easy Editing - Audio	[Editing mode UI] As an user, I want to see the current playing time and the length of the music file.	0.5		Not started
59		Easy Editing - Audio	[Editing mode UI] As an user, I'm able to adjust the volum in the editing mode.	0.5		Not started
58		Easy Editing - Audio	[Editing mode UI] As an user, I'm able to quit the editing mode by press "Back" key.	1		Not started
51		Easy Editing - Audio	[Editing mode UI] As an user, After I choose "Clip" in option menu, current music should start to play from the top.	0.5		Not started
55		Easy Editing - Audio	[Editing mode UI] As an user, After I choose "Clip" in option menu, MSK change to "Select"	1		Not started
56		Easy Editing - Audio	[Editing mode UI] As an user, After I choose "Clip" in option menu, play icon change to clipping status icon	0.5		Not started
31-40		Easy Editing - Audio	[Container] Clip file.		1	
33		Easy Editing - Audio	[Clip file] As an user, I'd like to long press navigation key for searching (at certain point, could be start/end point)	2		Not started
31		Easy Editing - Audio	[Clip file] As an user, I'd like to use MSK to mark the start point.	2		Not started
37		Easy Editing - Audio	[Clip file] As an user, After I press MSK to mark the start point, clipping status icon change to pressed status.	2		Not started
34		Easy Editing - Audio	[Clip file] As an user, After I press MSK to mark the start point, progress bar start indicate recording time.	4		Not started
32		Easy Editing - Audio	[Clip file] As an user, After start point marked, I'd like to use MSK to mark the end point.	1		Not started
38		Easy Editing - Audio	[Clip file] As an user, After I press MSK to mark the end point, clipping status icon change to normal status.	1		Not started
39		Easy Editing - Audio	[Clip file] As an user, After I mark end point, LSK change to "Save".	2		Not started
40		Easy Editing - Audio	[Clip file] As an user, After I mark end point, MSK change to "Play".	1		Not started
41-50		Easy Editing - Audio	[Container] Saving			
42		Easy Editing - Audio	[Saving] As an user, I'd like to review the sound clip before saving.	6	0.5	Not started
45		Easy Editing - Audio	[Saving] As an user, I'd like to have a visual indicator while saving file.	1	0.5	Not started
41		Easy Editing - Audio	[Saving] As an user, I'd like to save the new clip as a new file instead of replace the original file.	3	0.5	Not started
49		Easy Editing - Audio	[Saving] As an user, I'd like to review the new audio clip after it saved.	2	0.5	Not started
61-70		Easy Editing - Audio	[Container] Error Handling.		1.5	
62		Easy Editing - Audio	[Error Handling] As an user, after I choose clip start point, before I choose end point, if I press "Back" key or red key, I want a warning message to double confirm before I quit the editing mode.	1		Not started
61		Easy Editing - Audio	[Error Handling] As an user, after I choose clip start point and end point, if I press "Back" key or end key, I want a warning message to double confirm before I quit the editing mode.	4		Not started
2843598	10	Slamming	As a user, I'd like to see "Slamming" option in the option list.	1	2	Not started
2844001	?	Slamming	As a user, when I choose "Slamming" option, it should start BT slamming.	1		
?	?	Sharing	As an user, I'd like to see an option to share the current video clip.		3	Not started
?	?	Sharing	As an user, I'd like to see an option to share the current music/voice file.			Not started

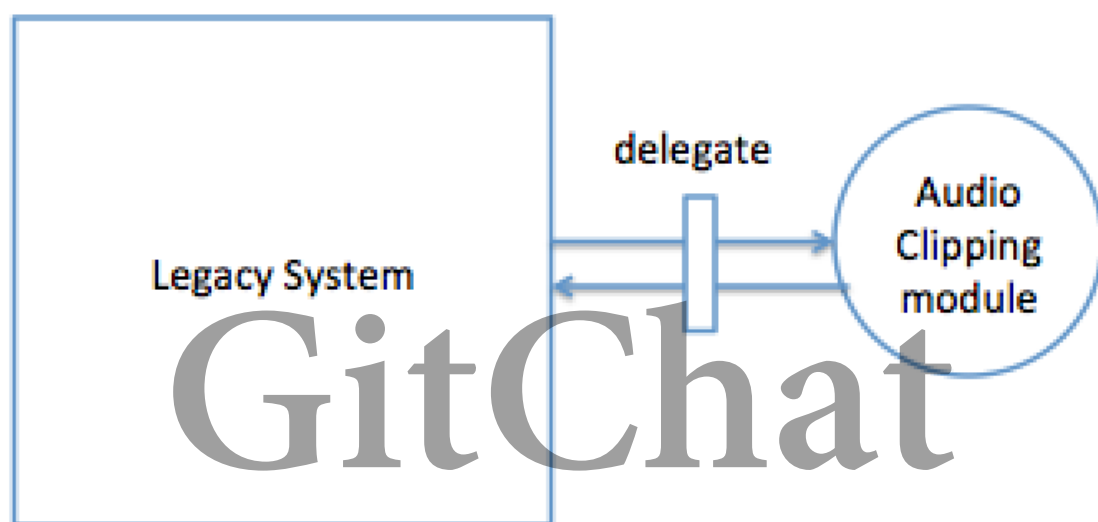
项目的估算

功能列表的拆分

在与团队一起听PO解释这个将要开发的新功能时，我们也将原来的功能列表进行了重新的拆分。并在拆分过程中，澄清了一些原本存在疑问的地方。拆分的原因在于原有的功能列表中很多功能都超过了一人周。对于一个开发经验较少的团队来说，很难一下子想清楚如何实现这么大一块功能。

User Story的估算

需求拆分的同时，我们还进行了一些设计上的讨论。□由于这个功能在之前的产品中根本没有提供过，所以，我们希望将其与原有的系统代码进行隔离，通过一个代理层，与原有系统的交互。这样就能够保障该功能的独立性，后期容易做功能增强或代码维护。基本设计如下图所示。



与此同时，我们还讨论了如何保证代码质量，通读了代码规范，制定了我们自己的团队代码约定，□而且决定引入单元测试，另外，部门技术领导还要求，在这个项目上要做提交前的Code Review。

现在就到了估算和排计划的时候了。有的同事提出了几个疑问，例如：

1. “如果我们写单元测试，原来的工作量估计是否要翻倍？”
2. “因为我们没有写过单元测试，是否要加上学习单元测试的时间？”
3. “我们这code review的时间需要加多少呢？”
4. “我们对需求的测试时间是不是也要加到估算中去？”

根据大家的提问，我提出了一种“新”的估算方法，即：Point估算，这是一种没有时间单位的估算。根据前面的拆分和这种相对估算方法，我们得到了新的用户故事列表和工作量，共计39个用户故事，126个点数，如下图所示：

360 ID	Numbe	Depender	Staus	User Story	User Story Points	Comments	Colu mn
3105290	UI_011		Done	Current track in NPS as input to the Audio Clipping Delegate	1	2D1C1T	4
3105294	UI_012		Done	Start the Audio Clipping UI Delegate and make an empty panel	5		
3106352	UI_013		Done	Handle end key in the Audio Clipping UI Delegate	3	2D5C	7
3264184	UI_024		Done	Spike on US_014	5		
3106387	UI_014	Core_005	Ready	As an user, After I choose "Clip" in option menu, current music should start to play from the top. - Create a new MO MO_Clip which inherit from MO_Sound - Make the Audio Clipping Controller create it - Make the MO_Clip have the interface to the Tone	8		
3106485	UI_015		Done	As an user, After I choose "Clip" in option menu, MSK change to "Select"	1	3D4C	7
3264199	UI_022		Ready (2nd)	As an user, I'd like to see a progress bar which can indicate: - Music playing progress	2		
3264208	UI_023		Ready (2nd)	As an user, I'd like to see a progress bar which can indicate: - Marked start point / end point	3		
3106490	UI_016		Done	As an user, I'd like to use MSK to mark the start point.	2	3D2C1P1T	7
3106508	UI_017		Done	As an user, After start point marked, I'd like to use MSK to mark the end point.	2	1D3C1T	5
3106514	UI_018		Ready	Display FF button in the Audio Clipping UI Delegate	1		

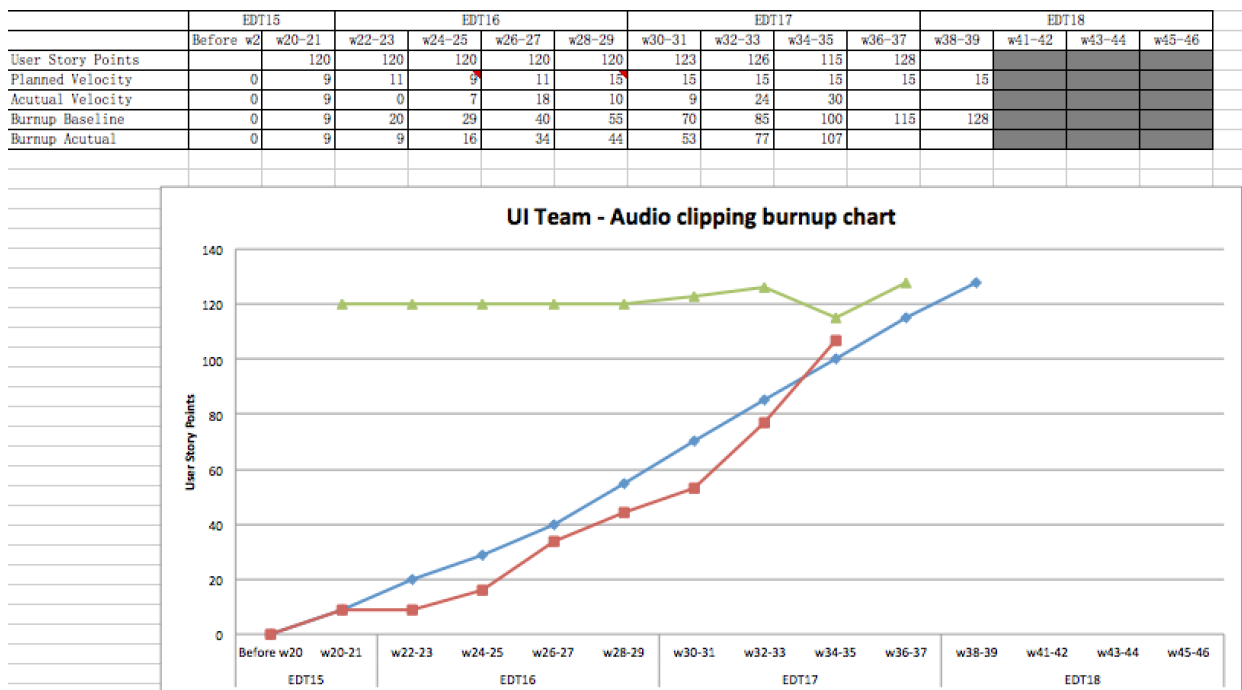
我们采用的点数序列是斐波那契数列。使用的估算方法于2011年发表在InfoQ上，名为《使用排序法对User Story进行相对估算》，这种方法的好处在于，对于每个故事的估算，实际上都是将它放在了一个有很多上下文的环境里，有更多的信息可以对照参考。方法的详细描述见后面的附录。同时，我们在估算时要求对每一个需求，下面这些时间均不考虑：

1. 用户故事的测试时间。
2. CodeReview的时间。
3. 写单元测试的时间。

其假设前提如下：

1. 写单元测试与Code Review的时间与开发活动的时间成正比例。
2. 开发活动是我们整个生产活动的瓶颈，对用户故事的验收测试可以满足开发节奏，不需要另行估计。我们并不是不承认测试人员的工作量，而是其不会影响开发计划的安排。

估算和执行结果都还不错，burn up chart如下图所示，其中绿色的线是需求范围，蓝色的线是计划，红色的线是实际完成情况。想知道为什么红色线有两处下凹吗？且听下回分解。



估算用途的讨论

在软件项目中，工作量估算的目的一般会有两种。一种是为了粗略估计项目成本或对外报价。另一种是为了做研发计划。

对外报价

通常这种情况下，并不需要整个团队参与估算，只需要有经验者（一个或者几个）拿到的招标文件或标书，根据过往的经验，进行估计即可。

做研发计划

在这种目标下，需求的解读、用户故事的拆分以及估算过程实际上是帮助团队成员了解项目目标及内容，对需求进行合理修正，补充可能被遗漏的需求细节，还可能包含一定的□设计讨论。

估算单位为什么不用“时间单位”

这样可以让大家更聚焦于需求本身，而不是很快联想到谁来做，需要花多长时间；

如果一定要用时间单位，怎么办？

我建议：

1. 尽可能使用较短的时间单位，比如“小时”。
2. 该时间为理想工作时间，即不受外界干扰的开发时间。
3. 每个需求开发的理想时长最好在2小时到8小时之间。通常不超过2个自然日。

为什么使用“理想小时”

这样便于项目负责人通过管理手段提升效率。不要误解了我的意思，我并不是想炸干每个开发者的血汗，而是让项目管理者可以意识到：“除了这些理想开发时间，还有多少时间浪费在不产生价值的事情上了，比如：（1）因为修复线上bug而插入的临时任务；（2）低效的团队会议占用了多少工作时间；（3）低效的沟通浪费了多少工作时间”。所有这些都是让你的效率下降的原因。请做好项目管理，让开发人员能够快乐地写代码吧
~~~~~

## 附录：

下面内容于2011年首发于InfoQ，链接地址如下：<http://www.infoq.com/cn/articles/ql-using-sort-method-to-estimate-user-story/>.

使用排序法对User Story进行相对估算

### 使用目标

制定发布计划，通常有一定数量的故事卡。

### 曾使用过的场景

项目规模较大（一到三个月的周期），已根据用户故事的拆分原则（INVEST原则）得到一个用户故事列表，该列表由各角色共同讨论过，且对每个用户故事的内容已达成共识。同时，团队基本上可以保证，每个卡片都会有两个或两个以上的开发人员了解，并有能力开发。

### 基本依据或假设

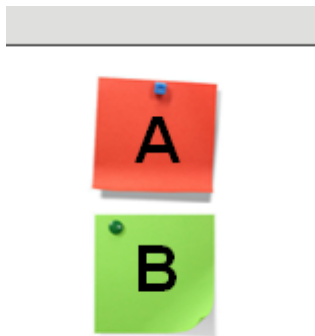
1. 用户故事的客观规模不会因为具体开发人员的差异而不同（尽管人员不同，花费的时间可能不同）。
2. 开发人员基本了解每个用户故事的工作内容。
3. 由于用户故事有一定的数量（从统计学的角度看），所以客观规模不会偏差太多。
4. 开发人员是整个交付过程的瓶颈，所以仅由开发人员估算其开发规模，不包括用户故事的测试规模。

### 估算过程



1. 将待估算的所有用户故事写在卡片上，并分发给每个参与项目的开发人员（均分即可）；
2. 先让一个开发人员取一张卡片A，贴到墙上（随便哪张都可以）；
3. 然后让每个开发人员按以下规则将手中的卡片贴在墙上：
4. 从自己手中取出一张卡片B，与墙上已有的那些卡片进行对比（开始时仅有一张卡片，即A）。

如果手中的卡片B与墙上A的大小相当，就将其贴在卡片A的下方；如下图所示。

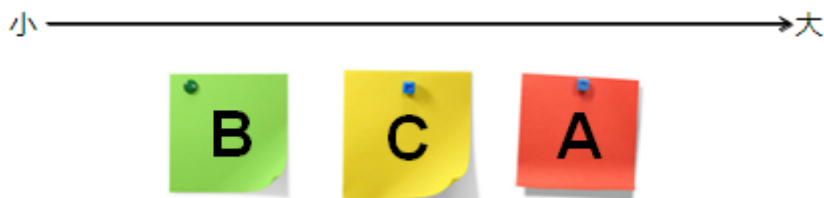


如果手中的卡片B比卡片A大，就将其贴在A的右侧，如图2所示。



如果手中的卡片B比卡片A小，就将其贴在A的左侧。

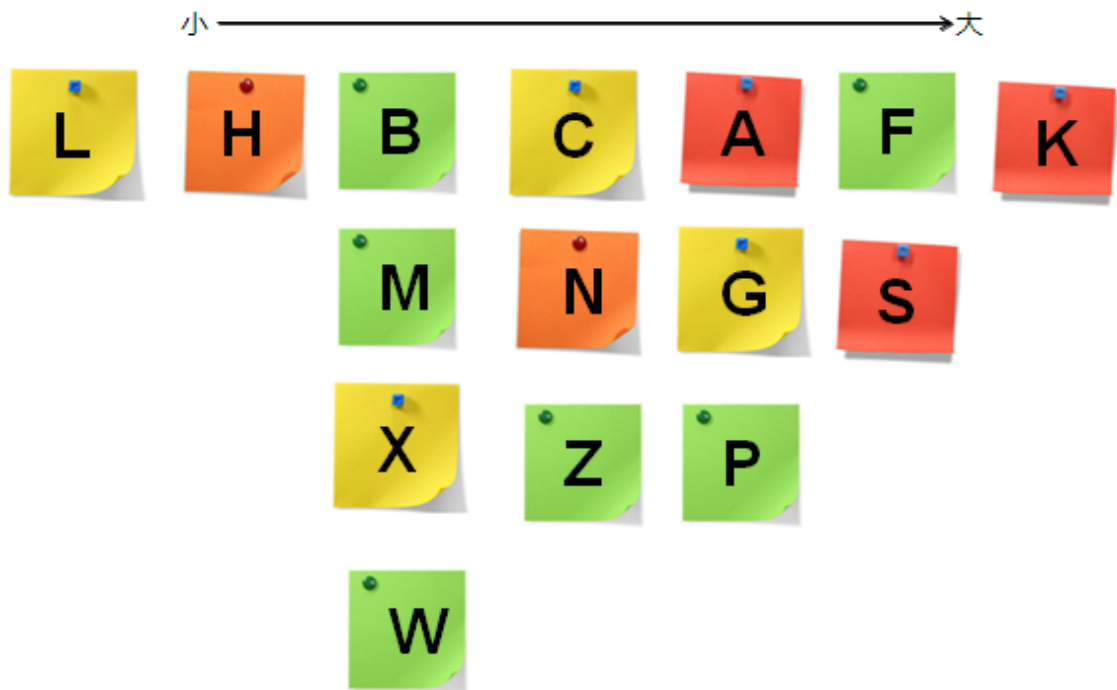
5. 继续再拿出一张卡片C，与墙上的卡片相比，如果比墙上所有卡片都小，则放在左侧；如果比它们大，放在右侧；与墙上某张卡片差不多大小，就放在其下方。如果C的大小在A和B之间，就把它放在A和B之间，如下图所示。



以此类推。在这个过程中，可以让所有开发人员同时贴卡片，只要彼此不干涉，保持独立判断即可。

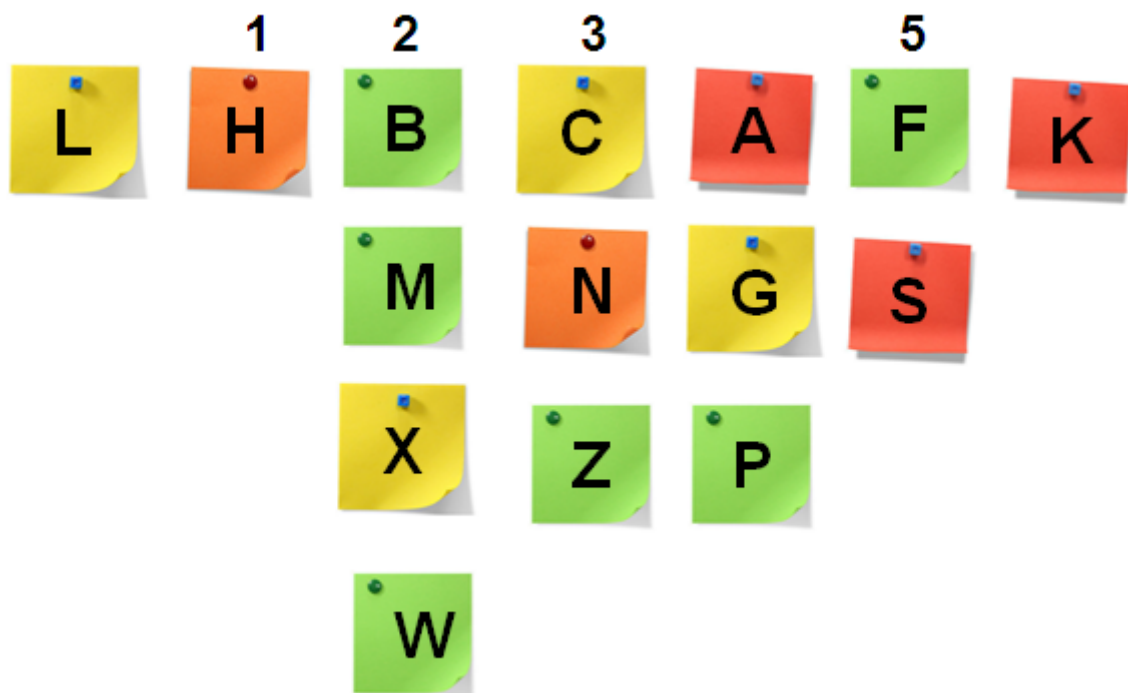
6. 当全部卡片贴完以后，再请全体开发人员重新看一看，是不是每个卡片的位置都很适当。



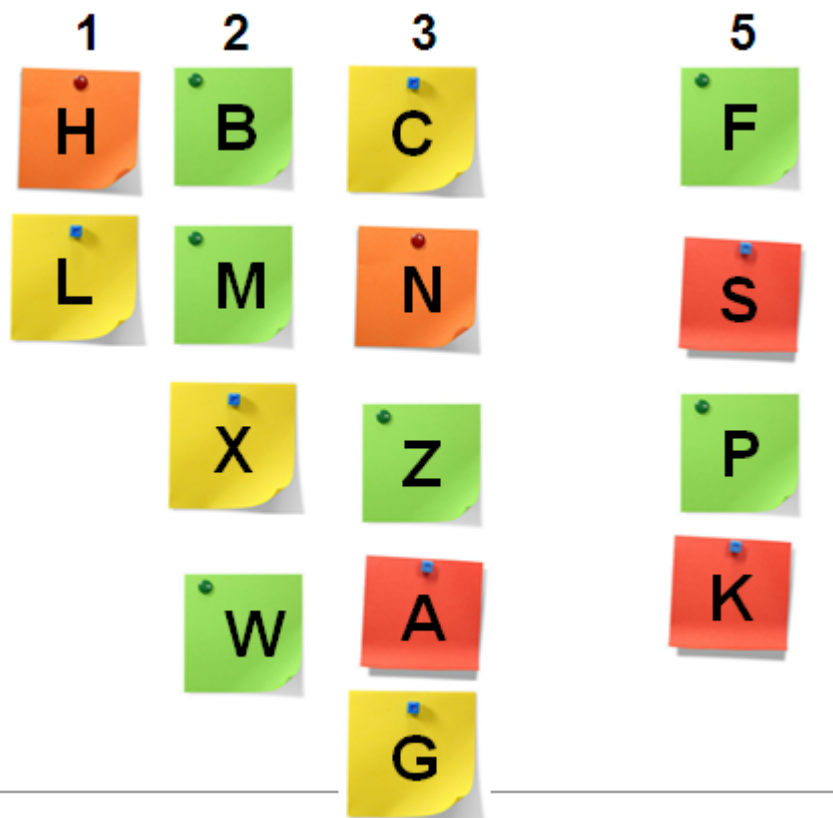


如果对某个卡片的位置有异议，请拿出来讨论。这也许是因为大家对它的内容和理解不一致造成的。因此需要对其进行深入讨论，直至一致认为它应该在某一列为止。

7. 使用数字 1,2,3,5,8,13 或 1,2,4,8,16 按顺序将其放在有对比关系的列上，如下图所示。注意，这时团队成员间还会发生对一些卡片的大小进行讨论，有可能也进行位置调整。图中，列在“2”的每个故事卡片，其大小应该是“1”列的一倍左右。由于是估算，所以并不需要精确，只要相当即可，以此类推。



对于那些列头上还没有数字标识的列（如图5中“L”列、“A”列和“K”列），请开发人员根据其规模的接近程度将其放到相邻的列中。如下图所示。



1. 将每列的用户故事数与列头的数字相乘，得到的数字再相加以后，就得到该项目的总体规模。这种方法的好处在于，对于每个故事的估算，实际上都是将它放在了一个有很多上下文的环境里，有更多的信息可以对照参考。

## 注意事项

1. 在估算之前，应该确保所有用户故事之间的规模差异不要过大。例如，某个用户故事大约需要一个小时完成，而另一个用户故事则需要两周。此时说明，用户故事的粒度不合理，不INVEST原则中的S原则，需进行合并或分拆。
2. 如果各列卡片的数量不符合正态分布，而是两头的卡片多，中间的卡片少，也说明用户故事粒度可能有问题，需要重新审视一下。在这种情况下，会为后续的迭代计划带来困难。
3. 如果在一列中有数个相关联且同样大小的故事（比如支持银联卡、支持MasterCard、支持VISA卡），且先做完一个卡片，其它两个工作量会减少的情况下，可以将任意一个放在当前列，其它两个可以考虑放在比较小的一列中。但是，具体情况还是要具体分析，因为实际情况较复杂。但在整体审视环节中，这类的分析和验证工作不可缺少。
4. 在讨论和移动卡片时，应该更多地与其它卡片进行对比，而不是直接说某个卡片属于哪一数字列。因为列头的数字都是相对值，没有其它卡片的对比，这些数字没有意义。
5. 在讨论的过程中，会捕捉到一些之前没有发现的问题或信息，此时一定要及时记录下来。对于不清楚的问题，应该当场由业务人员给出结论。

6. 这种方法在那些没有尝试过相对估算的团队中首次使用时，最好能有一定经验的人加以引导，对已排定的顺序进行适当验证。

# GitChat