软件需求工程之新手内心十大问题

要求：找一个陌生的行业，根据需求的特性：完整性，可行性，可验性，正确性，必要性，划分优先级，无歧性，分析并提出十个问题，无需解答 上传github（陌生行业：机械制造，卫生死亡登记，殡葬，林业，军事武器管理系统软件等）

需求说明的特征

完整性：每一项需求都必须将所要实现的功能描述清楚，以使开发人员获得设计和实现这些功能所需的所有必要信息。

正确性：每一项需求都必须准确地陈述其要开发的功能。做出正确判断的参考是需求的来源，如用户或高层的系统需求规格说明。若软件需求与对应的系统需求相抵触则是不正确的。只有用户代表才能确定用户需求的正确性，这就是一定要有用户的积极参与的原因。没有用户参与的需求评审将导致此类说法：“那些毫无意义，这些才很可能是他们所要想的。”其实这完全是评审者凭空猜测。

可行性：每一项需求都必须是在已知系统和环境的权能和限制范围内可以实施的。为避免不可行的需求，最好在获取（elicitation）需求(收集需求)过程中始终有一位软件工程小组的组员与需求分析人员或考虑市场的人员在一起工作，由他负责检查技术可行性。

必要性：每一项需求都应把客户真正所需要的和最终系统所需遵从的标准记录下来。“必要性”也可以理解为每项需求都是用来授权你编写文档的“根源”。要使每项需求都能回溯至某项客户的输入，如使用实例或别的来源。

划分优先级：给每项需求、特性或使用实例分配一个实施优先级以指明它在特定产品中所占的分量。如果把所有的需求都看作同样重要，那么项目管理者在开发或节省预算或调度中就丧失控制自由度。

无二义性：所有需求说明的读者都只能有一个明确统一的解释，由于自然语言极易导致二义性，所以尽量把每项需求用简洁明了的用户的语言表达出来。避免二义性的有效方法包括对需求文档的正规审查，编写测试用例，开发原型以及设计特定的方案脚本。

可验证性：检查一下每项需求是否能通过设计测试用例或其它的验证方法，如用演示、检测等来确定产品是否确实按需求实现了。如果需求不可验证，则确定其实施是否正确就成为主观臆断，而非客观分析了。一份前后矛盾，不可行或有二义性的需求也是不可验证的。

需求规格说明的特点

1.完整性

不能遗漏任何必要的需求信息。遗漏需求将很难查出。注重用户的任务而不是系统的功能将有助于你避免不完整性。如果知道缺少某项信息，用TBD (“待确定” )作为标准标识来标明这项缺漏。在开始开发之前，必须解决需求中所有的TBD项。

2. 一致性

　　一致性是指与其它软件需求或高层（系统，业务）需求不相矛盾。在开发前必须解决所有需求间的不一致部分。只有进行一番调查研究，才能知道某一项需求是否确实正确。

3. 可修改性

　　在必要时或为维护每一需求变更历史记录时，应该修订SRS。这就要求每项需求要独立标出，并与别的需求区别开来，从而无二义性。每项需求只应在SRS中出现一次。这样更改时易于保持一致性。另外，使用目录表、索引和相互参照列表方法将使软件需求规格说明更容易修改。

4. 可跟踪性

　　应能在每项软件需求与它的根源和设计元素、源代码、测试用例之间建立起链接链，这种可跟踪性要求每项需求以一种结构化的，粒度好（fine-grained）的方式编写并单独标明，而不是大段大段的叙述。

——以上摘自Karl E.Wiegers的《software requirements》 第一章

我将对于军事武器管理系统软件设计的需求分析提出以下几个问题：

1. 对于用户的需求，怎样才能保证他们所想的所说的需求能满足现在的需求，而对于未来的新的需求又如何保证？
2. 要了解军事武器的管理以及用户的需求，又应该向谁咨询调查？中间肯定涉及一些国家机密和安全问题，又该如何处理？
3. 就算对于一名经验丰富的程序员，也有一些技术上的问题而不能满足用户的需求，更何况对于新手而言的我呢，这时候遇到用户有这方面的需求而团队又无法在技术上解决，应该如何巧妙解决？这里当然不能回避用户的需求。比如军事武器管理系统需要自动识别管理员的身份，要求瞳孔和语音共同识别，而我们在这方面技术缺乏，这时候又该如何解决？
4. 关于可行性，我觉得除了技术上，还应该又资金上，和时间上需要确定是否可行。如果因为其中某一项或者多项功能导致工程无法按照预期进行，那又该如何比较合理的解决？
5. 对于军事武器，每一种武器肯定有成千上万，而他们又十分形似，这时候如何确保高效的精准的区分每一件武器并进行取出和存储操作？
6. 对于军事武器的管理，安全需求肯定是第一位，它肯定放在第一位，并且高于一切其他功能的权限，那么，如何保证在特殊时期管理系统能够快速的识别危机并且做出反应？
7. 我们都知道军事管理系统的用户肯定来自五十个名族，甚至有可能来自其他国家，那么在使用系统的时候，说明书就一定要考虑到用户对使用书的理解程度，避免产生二义性。

而在英文中一个单词又有多只含义，在不同的语境又不同的意思，这时候需要解决的问题就不是软件需求功工程师软件工程师所能解决的，这时候，我的脑容量似乎不够用了，快帮我想想办法。

1. 如何验证最终的武器管理系统能够满足所有需求？很明显这是一项巨大的工程，又不能有丝毫的失误，一旦被不法分子利用，损失无可挽回。那在设计系统初，能不能让系统自动验证，自动赋值，使用穷举法，来验证每一个功能和每一种可能发生的情况，如果有，又该如何实现？这种方法又可行？（天马星空了，啊哈哈哈）
2. 在对武器管理系统研发初期，我们对用户的需求进行分析，更因该考虑到该系统以后的更新问题，那么具体向哪方面考虑？
3. 在武器管理系统没有经历过长时间使用的检验，内心忐忑不安，无法入睡，掉头发，全是不舒服。终极问题：如何让自己开发的产品可靠高效，让人踏实?或者说：让它很有说服力，有存在的价值?