# 6.7 CHECKLIST: Design in Construction

## 6.7.1. Design Practice

**6.7.1.1 Have you iterated, selecting the best of several attempts rather than the first attempt?**

**6.7.1.2 Have you tried decomposing the system in several different ways to see which way will work best?**

**6.7.1.3 Have you approached the design problem both from the top down and from the bottom up?**

**6.7.1.4 Have you prototyped risky or unfamiliar parts of the system, creating absolute minimum amount of throwaway code needed to answer specific questions?**

**6.7.1.5 Has your design been reviewed, formally or informally, by others?**

**6.7.1.6 Have you driven the design to the point that its implementation seems obvious?**

**6.7.1.7 Have you captured your design work using an appropriate technique such as a Wiki, e-mail, flip charts, digital photography, UML, CRC cards, or comments in the code itself?**

## 6.7.2. Design Goals

**6.7.2.1 Dose the design adequately address issues that were identified and deferred at the architectural level?**

**6.7.2.2 Is the design stratified into layers?**

**6.7.2.3 Are you satisfied with the way the program has been decomposed into subsystems, packages, and classes?**

**6.7.2.4 Are you satisfied with the way the classes have been decomposed into routines?**

**6.7.2.5 Are classes designed for minimal interaction with each other?**

**6.7.2.6 Are classes and subsystems designed so that you can use them in other systems?**

**6.7.2.7 Will the program be easy to maintain?**

**6.7.2.8 Is the design lean? Are all of its parts strictly necessary?**

**6.7.2.9 Dose the design use standard techniques and avoid exotic, hard-to-understand elements?**

**6.7.2.10 Overall, dose the design help minimize both accidental and essential complexity?**