Requirements Statements - Examples

*These are example requirements for a computer game.*

1. Can run on any operating system.
2. Needs the latest java installed to run the project.
3. The password should be encrypted for safety.
4. The software will not cause the user's computer to explode.
5. Software can run decent on 56K modem and above.
6. The applet needs to download in less than ten seconds on all kinds of computer stations.
7. Software must run on any browser.
8. The website should be accessible from any location with access to the WWW.
9. Software can run on any internet explorer.
10. User has internet.
11. Needs to run on website for universal access.
12. Must be finished on time.
13. Software must be reliable and not crash.
14. Must be free of defects.
15. Any crashes in the applet should be fixable in a subsequent login
16. Needs to fail less than once a week.
17. User has password so no other user can mess with his/her game.
18. Needs modifiability for future updates and enhancements.
19. Must provide hours of fun!
20. Needs to use minimal system requirements.
21. Amateur and advanced users can run the GUI.
22. Display should look good.
23. Software must be intuitive or easily understood after reading the rules.
24. Needs comfortable layout that accommodates beginners as well as veterans.
25. The layout should be self-explanatory enough so that any user needs only access the user manual 3 times to understand the functions of the product.

*Here's a critique:*

1. Can run on any operating system.   
   Even MS-DOS?  CP/M?  MVS?  RTOS? ...  
   Not a sentence.
2. Needs the latest java installed to run the project.   
   Java or Java? JRE?  SDK?  Which version is "latest?"
3. The password should be encrypted for safety.    
   This is a security issue, not a safety issue.  
   There are many different kinds of encryption. A captain crunch decoder ring is encryption. Quantify the encryption strength required.
4. The software will not cause the user's computer to explode.  
   How about if it just smokes a little?
5. Software can run decent on 56K modem and above.  
   "decent" is too colloquial for describing software performance.  Not quantifiable.
6. The applet needs to download in less than ten seconds on all kinds of computer stations.  
   Even my old 300-baud TTY-style terminal?
7. Software must run on any browser.  
   Even Lynx? (A text-only browser)  "Any" is a tipoff to an infeasible requirement.
8. The website should be accessible from any location with access to the WWW.  
   What is location?  What if it's accessible but not viewable?  What does "access" mean? My PDA has access to the web.  My refrigerator might have access to the web.
9. Software can run on any internet explorer.  
   Does this refer to a specific product? Microsoft Internet Explorer?  Specify version.
10. User has internet.  
    Too vague.
11. Needs to run on website for universal access.  
    What does "run" mean here?  What is "universal access?"
12. Must be finished on time.  
    Not a quality requirement.  This is a project requirement, not a product requirement.
13. Software must be reliable and not crash.  
    Not feasible.
14. Must be free of defects.  
    Not feasible.
15. Any crashes in the applet should be fixable in a subsequent login  
    Confused.
16. Needs to fail less than once a week.  
    Pretty good, but what constitutes failure?
17. User has password so no other user can mess with his/her game.  
    "Mess" is colloquial.  Not stated as a requirement of the software.  Vague purpose.
18. Needs modifiability for future updates and enhancements.  
    Vague.  What will be modified?
19. Must provide hours of fun!  
    Fun is a subjective state.
20. Needs to use minimal system requirements.  
    Wouldn't minimal be zero?
21. Amateur and advanced users can run the GUI.  
    "Amateur" has a connotation, use less judgemental terms.  What is "run?"  Launch?  Navigate? Enter data?
22. Display should look good.  
    "Good" is not measurable.
23. Software must be intuitive or easily understood after reading the rules.  
    Intuitive is not quantifiable.
24. Needs comfortable layout that accommodates beginners as well as veterans.  
    Comfortable is not measurable.  "Veterans" isn't quite the best way to describe advanced users.
25. The layout should be self-explanatory enough so that any user needs only access the user manual 3 times to understand the functions of the product.  
    Good try.  What is layout?  Preconditions on user?  Understand or use?  Is "functions" too general? Spelling errors.