Homework 2

DEFECT LIST

Defect #	Req #	Description
1.	FR #2	Complexity: The sentences detailing how loans should be updated for both loans in the Fanny May portfolio and loans that aren't in the portfolio should each be their own sub-requirement.
2.	FR #2	Vagueness: The requirement states that "loans not in the Fanny May portfolio will also be examined in order to determine if a borrower's standing *should be* updated", but what is the metric for whether or not it "should be updated"?
3.	FR #12	Inconsistency: If a borrower has at least one default loan AND at least one late loan, this requirement states that they can be both in both a "default" and "late" standing simultaneously.
4.	FR #13	Contradiction: The requirement states that "once a loan or borrower is in default standing, it cannot be changed to another standing", but the requirement #12 says that a "borrower is considered to be in "late" standing if any of the loans to that borrower have default standing".
5.	FR #14	Vagueness: In what situations would a loan change from ARM to FM or vice versa?
6.	FR #19	Vagueness: The requirement states that a "loan analyst *should be able to* review all of the information in the repository...", but "should be able to" is non-committal and implies that they might not be able to do reviews.
7.	FR #27	Inappropriate Implementation: The requirement gives instruction to allow ranges for the organization of attributes of a loan, but this is a "how" detail (how attributes should be implemented) rather than a "what" (what attributes should be remembered about a loan).
8.	FR #27	Untestable/Impossible Implantation: The requirement gives instruction to allow ranges for the organization of *any* attributes of a loan, but how do you apply a range for the "type of loan", "borrower", "lender" etc... attributes? The requirement should specify which attributes should have ranges.
9.	FR #29	Vagueness: The requirement states that "the loan analyst should be able to request reports" but "should be able to" is non-committal and implies that they might not be able to request reports.

10.	FR #29	Vagueness: What does it mean for the loan analyst to receive a report in the format of "the screen"? Is that different from seeing it in a file?
11.	FR #31	Vagueness: The requirement states that "The bundling function should consist of two major steps", but "should" is non-committal. Do we need less steps? More steps?
12.	FR #31	Complexity: The three sentences outlining if a bundle is accepted, rejected, or modified should be subclauses of what can happen after a candidate bundle is marked.
13.	FR #32	Omission: The requirement states that when a loan is marked for a bundle, it is not made available for inclusion in other bundles. But this and the previous requirement make no mention about what should happen if a marked loan was removed from a bundle. Is it still marked and therefore inaccessible? The requirement should mention what happens when a bundle is removed (i.e., it gets unmarked).
14.	FR #35	Ambiguity: What does the requirement mean by a "transaction" being resolved? What is a "transaction" in this context? The only other time transaction is mentioned in the document is for when a bank uses money from loans for various other activities.
15.	FR #37	Untestable: Without detailing what format these reports can come into the system, there's no way to test if the application can handle any "new file format".
16.	FR #39	Vagueness: The requirement states that "the application should work on a Unix System" but "should" is non-committal and implies that it could work on other systems outside of Unix such as Windows
17.	FR #40	Vagueness: The requirement states that "the loan analyst should be able to look at information about more than one [X] at a time" but "should" is non-committal and implies that it would be ok if the loan analyst is only ever able to look at one [X] at a time
18.	FR #43	Vagueness: What is meant by "a single record or line of output"? What kind out output is the requirement talking about?
19.	FR #44	Vagueness: What is considered an "inappropriate or illegal" search request?
20.	FR #45	Vagueness: The requirement states that the app "should return a user to the previous screen" if an error occurs, but "should" is non-committal and implies that it's fine if the user is not returned to a previous screen.
21.	FR #49	Untestable: Are we running that app for 8 hours and making sure it was accessible for 7.76 hours? Should it run for at least that? At most that? Exactly that?