Unit Design Phase Defects

|  |  |  |  |
| --- | --- | --- | --- |
| **Review Form** | | | |
|  |  |  |  |
|  | **Document Title:** | **Project\_RequirementsDocument\_LabAssiAsseProjectV02** | |
|  | **Reviewer Name:** | **Diaconu Ana-Maria, Duma Amalia-Diana** | |
|  | **Review date:** | **02.03.2023** |  |
|  |  |  |  |
| **Crt. No.** | **Checked Item** | **Doc. page/line** | **Comments/ improvements** |
| 1 | D\_06 | Codebase | If the format of the files from where it’s read is changed, the code should be modified in a lot of places. A better practice is to use a property or a variable to hold the separator. |
| 2 | D\_05 | Page 4/ Table 3.1/FR.2.0 | The subject entity is mentioned but it is not defined nor described beforehand, unlike Assignment, Student and Grade. |
| 3 | D\_03 | Overall | No error handling strategy is provided – only the basic functionalities are described. Example: what if we want to filter the student by an unexisting criteria? (such as age) |
| 4 | D\_01 | Page 4/ Table 3.1/FR.2.0 | The meaning of “manage laboratory themes and subjects” is vague, not properly explained. |
| 5 | D\_06 | 2.2 | The current architecture accommodates only one actor – the teacher and may not be modified for other types of actors (such as student, administrator) |
| 6 | D\_02 | Page 4/ Table 3.1/FR.3.0 | The student filtering criteria is not specified |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |
| 13 |  |  |  |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 |  |  |  |
|  |  |  |  |
|  | **Effort to review document (hours): 1** | |  |

Coding Phase Defects

|  |  |  |  |
| --- | --- | --- | --- |
| **Review Form** | | | |
|  |  |  |  |
|  | **Document Title:** | **Java project – LabAssiAsseProjectV02** | |
|  | **Reviewer Name:** | **Duma Amalia-Diana; Diaconu Ana-Maria** | |
|  | **Review date:** | **02.03.2023** |  |
|  |  |  |  |
| **Crt. No.** | **Checked Item** | **Doc. page/line** | **Comments/ improvements** |
| 1 | D\_07 | Codebase | Inconsistent use of language when naming variables, methods and comments: either Romanian or English which can lead to confusion. |
| 2 | D\_01/D\_07 | AbstractCrudRepository – save(E entity) method | Else branch is not necessary since if the if statement is true, null is returned and we exit the function. |
| 3 | D\_01 | UI | When you want to add a student/grade/lab topic and use an existing id or when you want to update a student/grade/lab topic and provide invalid data - you get redirected to the main menu instead of the student/grade/lab topic menu. |
| 4 | D\_07 | Codebase | Insufficient validation when adding a student: you can give “13” as the name and it’s saved successfully. |
| 5 | D\_01 | Codebase | Inconsistency in decision regarding the way update works. In repository it’s returned null if entity was updated but in ui if we get null from repository (entity was updated) it’s considered that the entity doesn’t exist. |
| 6 | D\_07 | Codebase | Printing shows insufficient information:  Afisare lista note  1,1,10.0,2018-10-10  No details about what each field means. |
| 7 | D\_01 | UI - updateStudent | For the update functionality, the user has to complete all fields with non-empty values regardless of the field they want to change. If the user does not complete any of the fields, the program crashes instead of autocompleting the fields with default values |
| 8 | D\_01 | UI - updateTema | For the update functionality, the user has to complete all fields with non-empty values regardless of the field they want to change. If the user does not complete any of the fields, the program crashes instead of autocompleting the fields with default values |
| 9 | D\_07 | Grades file input | The application provides the possibility to input, in the initial grades file, grades for non-existent student and lab topics. No validation is done. |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |
| 13 |  |  |  |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 |  |  |  |
|  |  |  |  |
|  | **Effort to review document (hours): 1** | |  |