# Change request log

# Team

Group 3

Susheel Pagadala - sxp155831

Mayuri Banubakode - mxb156530

# Change Request

CR-FEMR-SPMB-208 - encounter PDF not displaying amount of prescription dispensed

***Description:***

The generated PDF is not displaying the amount of a prescription that was dispensed. To recreate this:

1) Create a new patient in Triage

2) Prescribe them some medications in Medical. Make sure the amount is > 0

3) Dispense the prescriptions in Pharmacy (click Submit)

4) View the patients encounter summary by pulling their ID up in Triage and clicking "Patient History" on the bottom.

5) Select the encounter where the prescription was dispensed on the right. The dispensed prescription is displayed here with the amount.

6) Click the red "Generate PDF" button. The dispensed prescription is displayed here, but without the amount.

# Concept Location

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | We ran the system |  |
| 2 | We created a new patient with ID 12270 | To create a new patient encounter |
| 3 | We created a new medical prescription | To create a medical prescription. |
| 4 | Encountered errors while creating prescription. To resolved this followed the steps provided by the professor | To resolve pre-existing bugs |
| 5 | We navigated to pharmacy screen | To dispensed the prescription |
| 6 | Reload encounter for the patient 12270 | To pull up patient history. |
| 7 | Clicked the Generate PDF button and checked dispensed prescription | To generate PDF report. |
| 8 | We determined the location of the change required in “PDFController” class | Determine a class to be changed |

**Time spent (in minutes):** 60

# Impact Analysis

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | We reviewed the validate class and method called by PDFController.java and getAssessments() respectively. | To track the classes that could be impacted by the change. |
| 2 | Determined the required change and impact to overall application | To avoid impact and failures to other components |

**Time spent (in minutes):** 15

# Actualization

Using the table below, describe each step you followed when changing the code. Include as many details as possible, including why classes/methods were modified, added, removed, renamed, etc.

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | We updated the validation method for getAssessments | Concatenated amount with medication name |
| 2 | We committed our changes with git. | To track our changes and also in case we need to revert our changes |

**Time spent (in minutes):** 30

# Validation

Using the table below, describe any validation activity (e.g., testing, code inspections, etc.) you performed for this change request. Include the description of each test case, the result (pass/fail) and its rationale.

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | Test case defined: Valid data – Created new encounter with single prescription | The test passed with expected behavior. |
| 2 | Test case defined: Valid data -  Inputs: Created a new encounter with multiple prescriptions | The test passed with expected behavior. |
| 3 | Test case defined: Valid data –  Created new encounter with no prescriptions | The test passed with expected behavior. |
| 4 | Test case defined: Valid data -  Inputs: Changed the prescription to see original and new prescription. | The test passed with expected behavior. |

**Time spent (in minutes):** 15

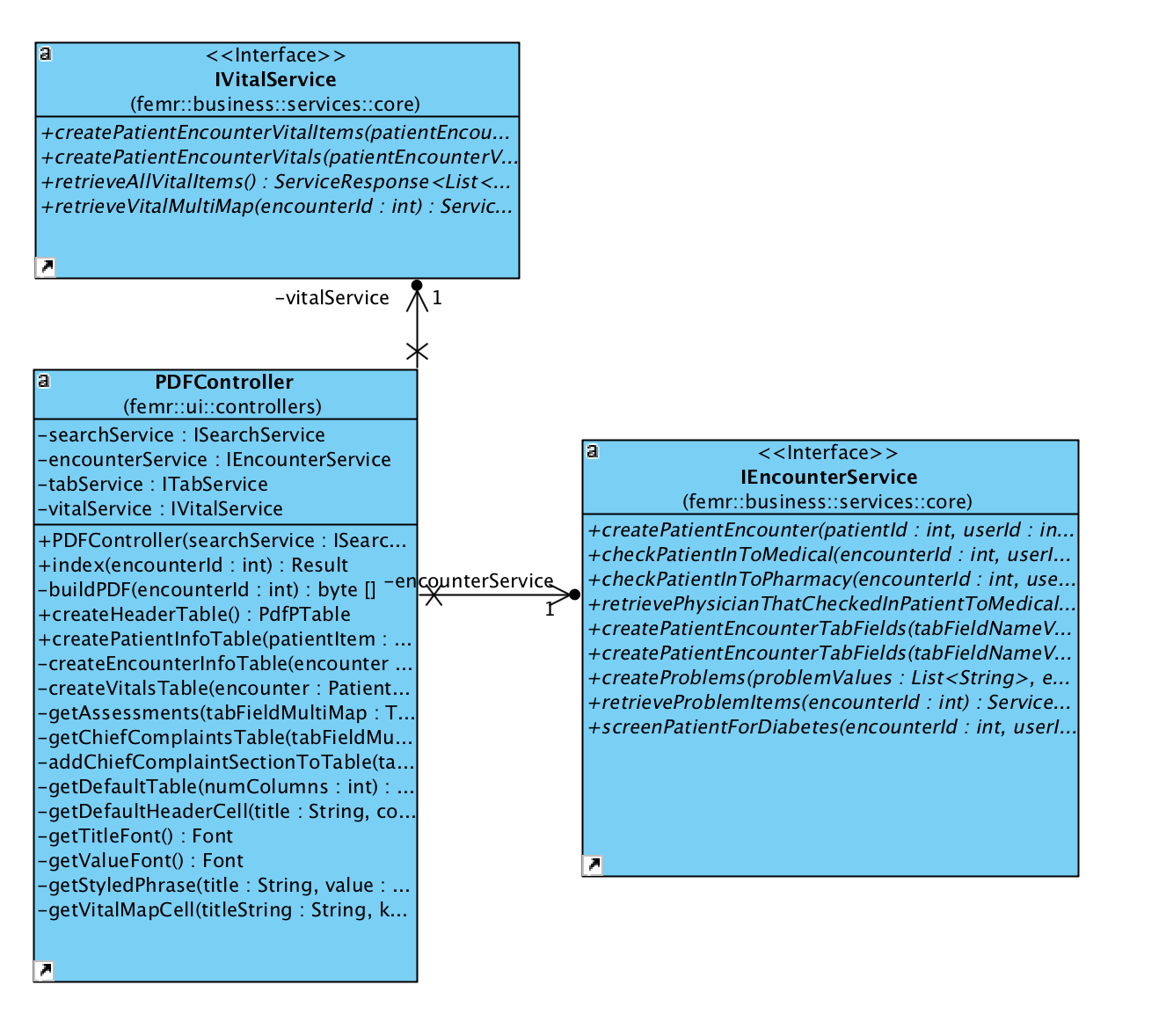
# Timing

Summarize the time spent on each phase.

|  |  |
| --- | --- |
| Phase Name | Time (in minutes) |
| Concept location | 67 |
| Impact Analysis | 15 |
| Prefactoring |  |
| Actualization | 32 |
| Postfactoring |  |
| Verification | 25 |
| Total | 149 |

# Reverse engineering

Class Diagram



# Conclusions

For this change, concept location was slightly confusing due to lack of expertise with jquery. Because change required is small and code is not complicated we are able to identify Concept location, impact analysis, actualization was done using IDE IntelliJ, which was very useful. Verification was performed manually with above listed test cases.

Classes and methods changed:

* **package** femr.ui.controllers;
* **public class** PDFController **extends** Controller
* **private** PdfPTable getAssessments(