Change Request Log fEMR-208

Team 5

Barbara Maweu - Logging
Juan Manuel Florez - Programming

Change Request

encounter PDF not displaying amount of prescription dispensed
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 patient's 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	Run fEMR following the instructions of the change request but were unable to save Prescription, we got an exception error in class **MedicalController.**	To recreate and identify the issue presented in the change request.
2	Review the **editPost()** method reported by the exception in the class **MedicalController**. Found that there was some missing ids not being returned to the application from the web client.	Trying to determine why Prescription wouldn't save.
3	Look at the mySQL database to review the data store for medication tables	To make sure there is medicine data in the system.
3.	***medication_medication_active drugs*** There were no active drugs stored in	This was one of the tables we found associated with medication. This was not

	this table but we also found two foreign key constraints to this table **medication_active_drugs medications**	the problem, however.
5	Check the admin console to see how to add drugs to the inventory. Add the current logged in user to a trip.	When we went to the admin console we found an alert saying that the current user was not in any trips. After this change we could successfully prescribe medicine.
6	Replicate the issue as described. This time we were successful.	Once we had a generated PDF, we were able to understand the issue to be addressed.
7	Search "PDF", only result was **PDFController**	We wanted to find the class where the PDF was being generated.
8	Search "Dispensed Prescription" in class **PDFController,** found a match in *getAssessments()*	We used the string from the generated PDF. We realized this method was where the change would need to be performed.

Time spent (in minutes):240

Impact Analysis

Step	Description	Rationale
1	In **getAssessments()** we found that the class **PrescriptionItem** stores information related to a prescription. We found that the quantity is being stored in field *amount* inside this class*.*	This meant that no other modifications to the class hierarchy were necessary, because the amount for the prescription was known at the time of PDF generation. Since this method is private and has no side effects, no other methods or classes needed to be examined for possible changes.

Time spent (in minutes):10

Actualization

Step	Description	Rationale
1	Modify method **PDFController.getAssessments()** to include the prescription amount along with its name.	This change would ensure that the prescription amount is output on the PDF alongside with the prescription name.

Time spent (in minutes): 20

Validation

Step	Description	Rationale
1	Generated the PDF we had generated previously with the same prescription for our test patient and the quantity of the the prescription was now being displayed on the PDF	We did this to make sure that our code change resolved the issue of not displaying the prescription quantity on the PDF
2	Added another drug in the inventory, then replaced the prescription with its equivalent and generated a PDF to verify that the quantity of the replaced medication would be displayed on the PDF	To make sure that the PDF was now output with a replacement prescription amount.

Time spent (in minutes):10

Timing

Phase	Time (in minutes)
Concept Location	240
Impact Analysis	10
Actualization	20

Validation	10
Total	**280**

Reverse Engineering

Diagrams are included along with this document. The fields and methods included in the class diagram are mostly restricted to the ones we had to modify or use for impact analysis.

Conclusions

We had trouble at the beginning of the process because the state of the SQL dump provided didn't allow prescribing medicine if logged in as the suggested user in the instructions. Once we figured out how to replicate the issue, the change was simple to implement, though. Since there was only one place in the code base where the *PDF* concept was implemented, it only took a simple text search to locate this functionality. Additionally, since the Play Framework implements the MVC pattern, we know that most or all logic related to a business logic context will most likely be found inside a single controller. This meant this change had no repercussions in other classes, which made impact analysis easy.

Classes and methods changed:
- femr.ui.controllers.PDFController
  - getAssessments(TabFieldMultiMap, List<PrescriptionItem>, List<ProblemItem>) : PdfPTable