Design Documents

Edit | New Page

Jump to bottom

[redacted] edited this page on Oct 6, 2019 · 14 revisions

Design Decisions

Frontend Decisions

For the frontend a few key decisions were made, importantly though, we decided to stick with the same design patterns already employed in iTrust2 for maximum cohesion. For patient survey functionality, we decided that we could use the same form to edit and create survey entries, thus limiting the amount of duplicated code. We also decided to put "Surveys" and "HCP Ratings" in the appointment dropdown menu. Additionally, we modeled our search function after other searches in iTrust2 (e.g. the search function in "edit patients" in the HCP view). We decided to put the link to HCP aggregate survey values in the "Patient" dropdown in the HCP's view.

Backend Decisions

For the backend, much like the frontend, we decided to stick with the established patterns of the existing system. The new REST API endpoints were all placed in a new class called APIOfficeVisitSurveyController which handles adding/editing surveys, and retrieving surveys and aggregated results. The HCPController and PatientController classes have had new mappings added that will return the appropriate pages for viewing HCP survey results and creating/editing surveys.

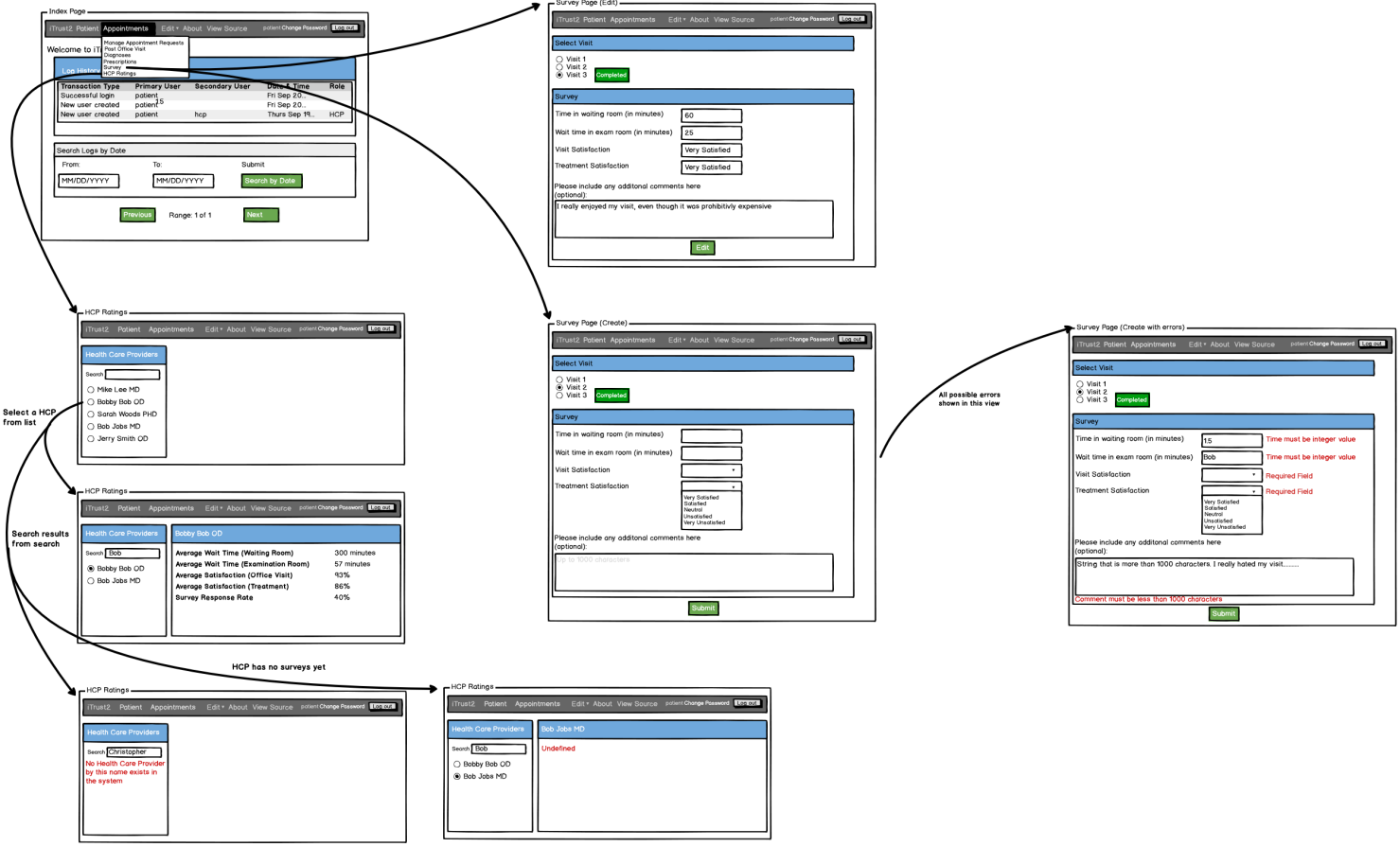
For the forms, we chose to use the same structure as many of the other forms in iTrust2, whereby we have a OfficeVisitSurveyForm class that contains only String, Integer, and Long types and represents the form as it is passed in from the frontend. The OfficeVisitSurveyForm is then used to construct a new OfficeVisitSurvey which is what is stored in the database.

Finally, we decided to create an additional class to represent the aggregated survey results which is HCPSurveyResults. A instance of the HCPSurveyResults class is created using a list of OfficeVisitSurveys from a single HCP. The values of each survey are aggregated together, and only a single object is returned to the frontend for each HCP. This was considered preferable over returning all surveys and leaving the bulk of the aggregation calculations for the frontend.

Frontend Design Elements

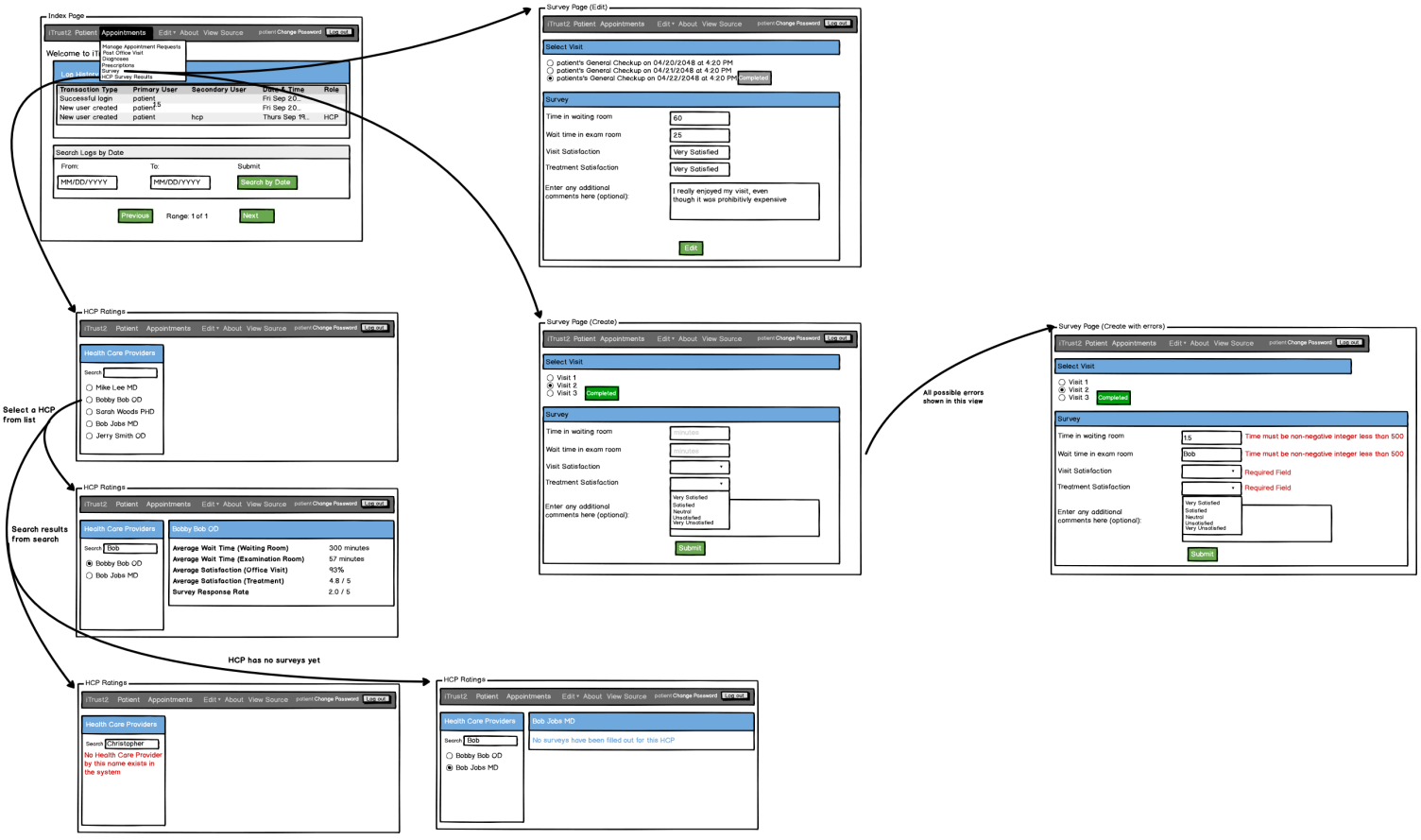
Patient Flow Map

Original Design



Created by [redacted] (Survey Form) and Nicholas Bruinsma (HCP Aggregate Survey Results)

Updated Design



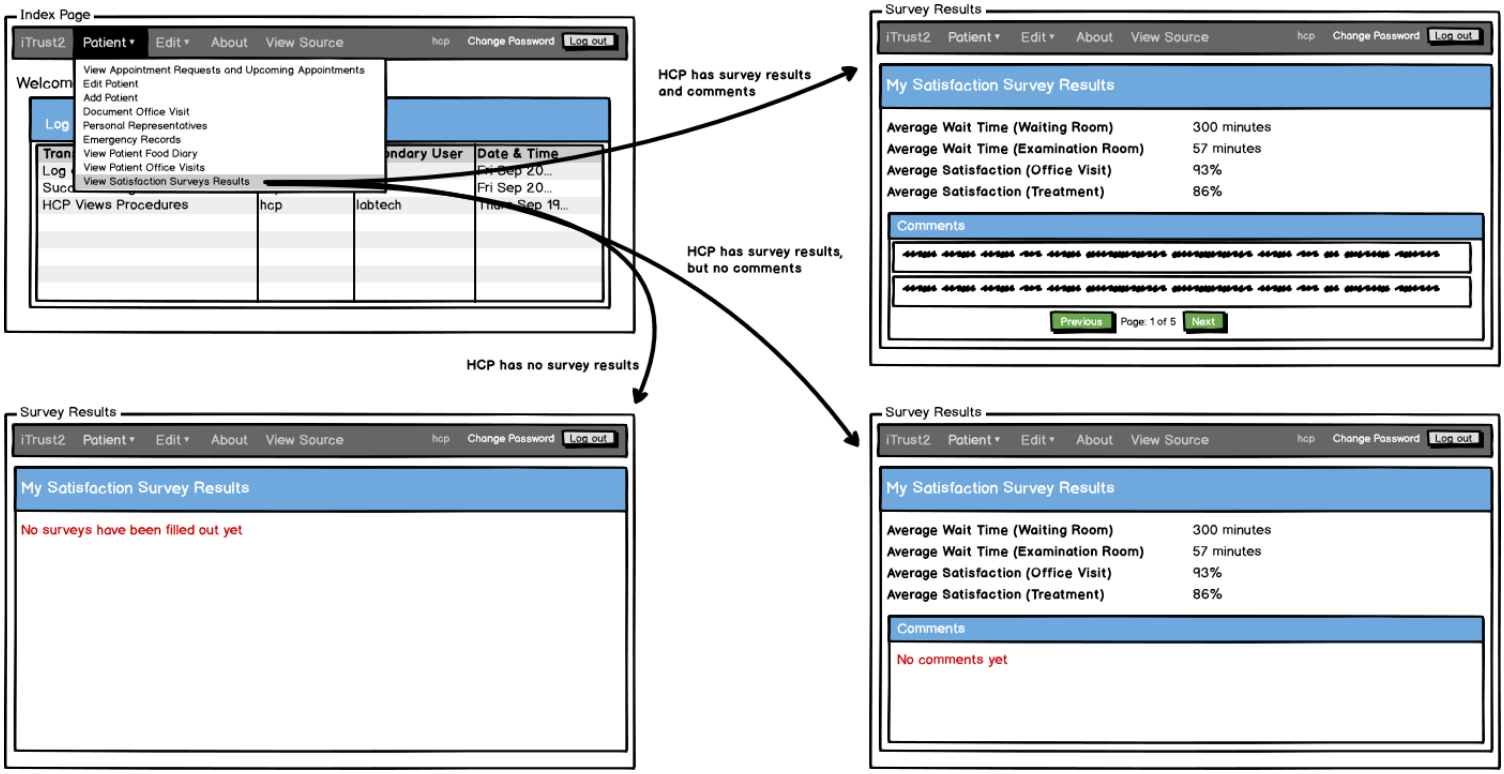
Changelog

- Changed average satisfaction formatting from **80%** to **4.0 / 5**
- Changed the wording of various labels/messages

- Removed different pages for comments, all comments are displayed on the same page

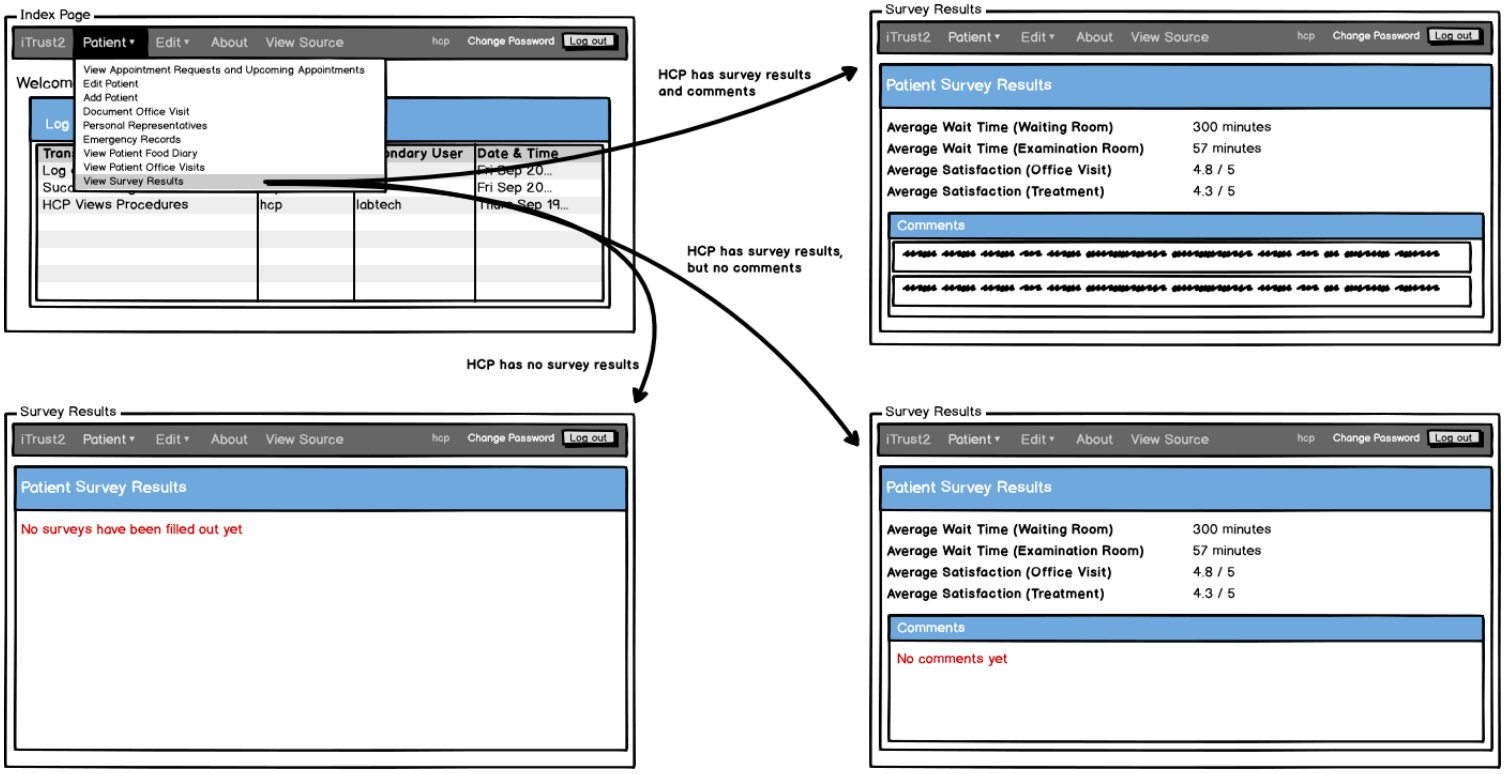
HCP Flow Map

Original Design



Created by [redacted]

Updated Design



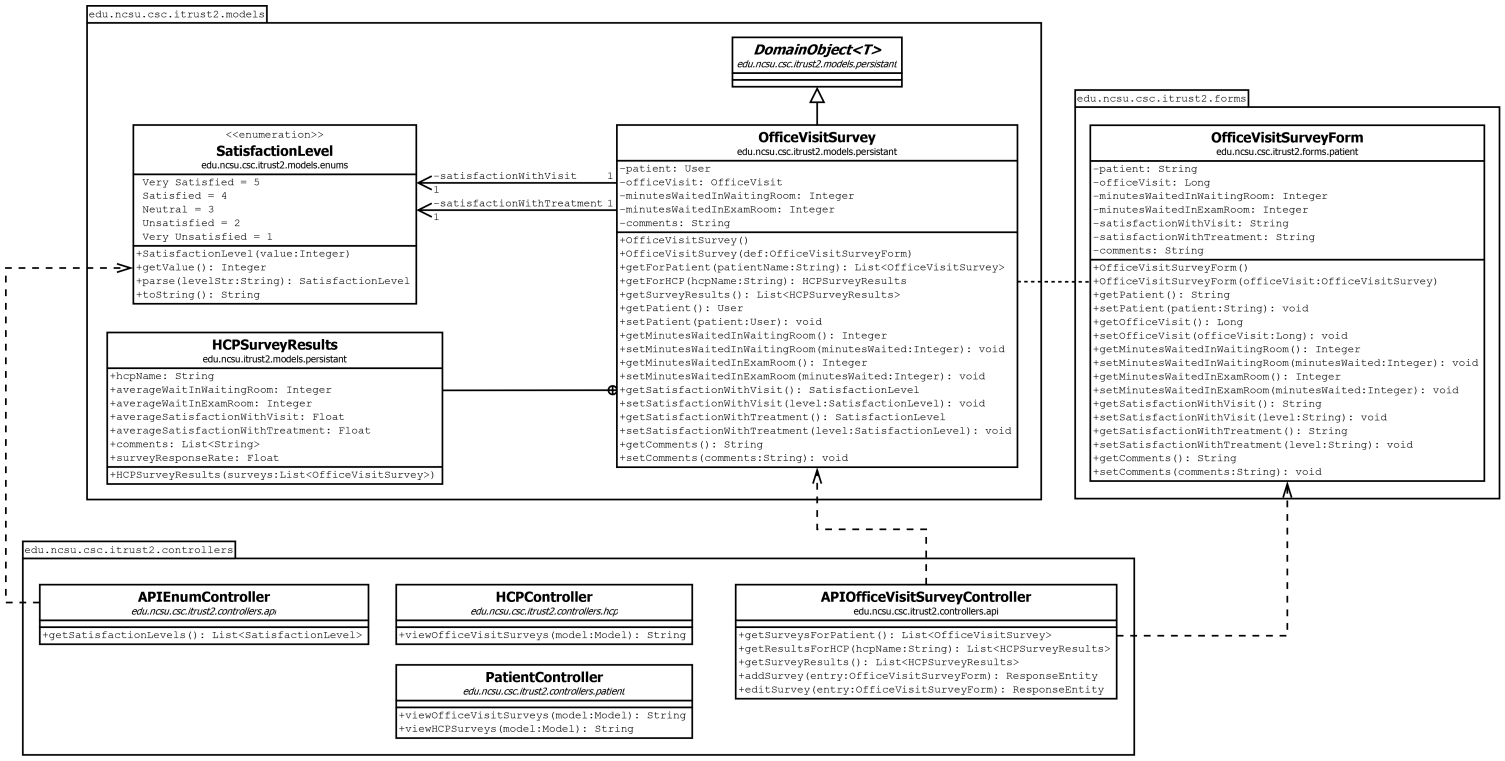
Changelog

- Changed average satisfaction formatting from **80%** to **4.0 / 5**

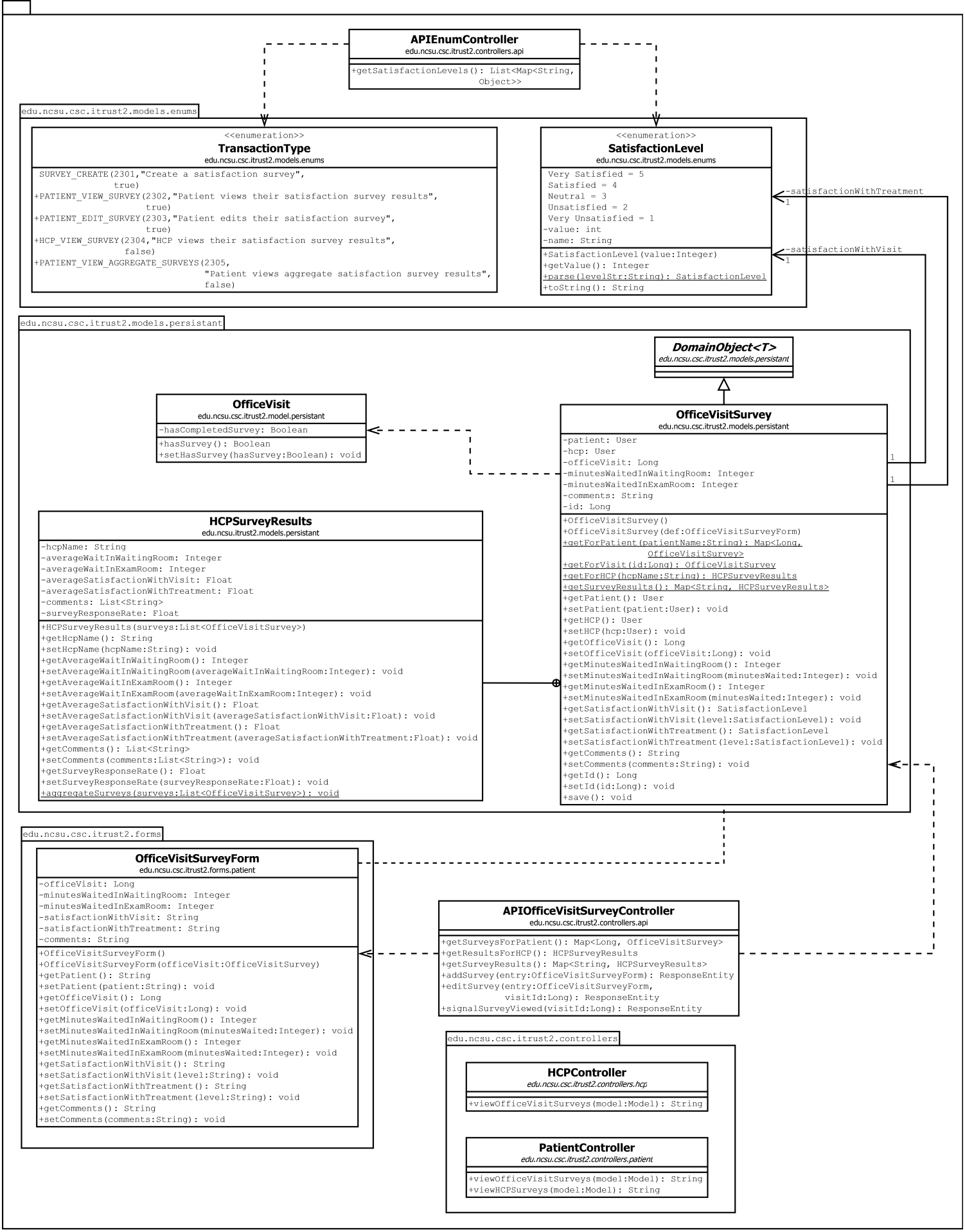
Backend Design Elements

Class Diagram

Original Design



Updated Design



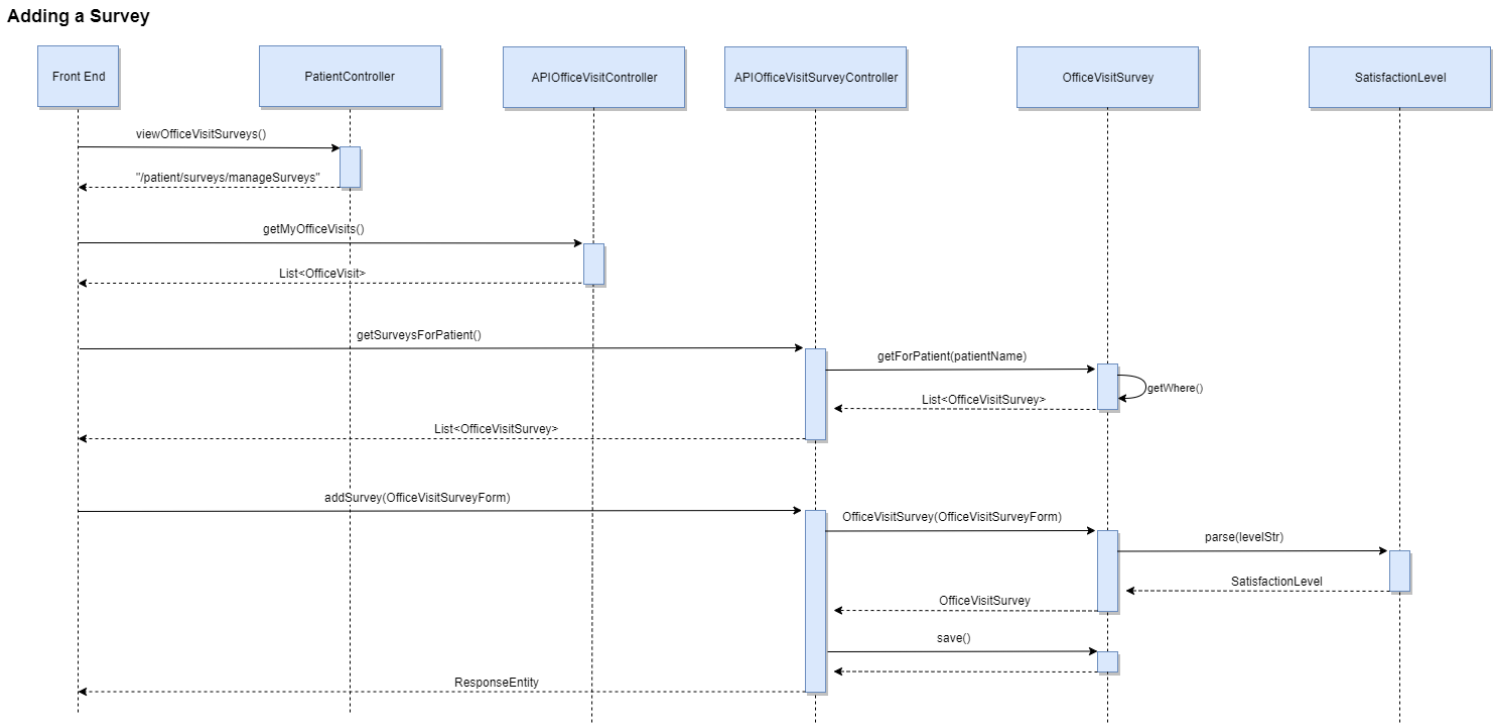
Changelog

- Changed `officeVisit` field in `OfficeVisitSurvey` from type `OfficeVisit` to `Long`

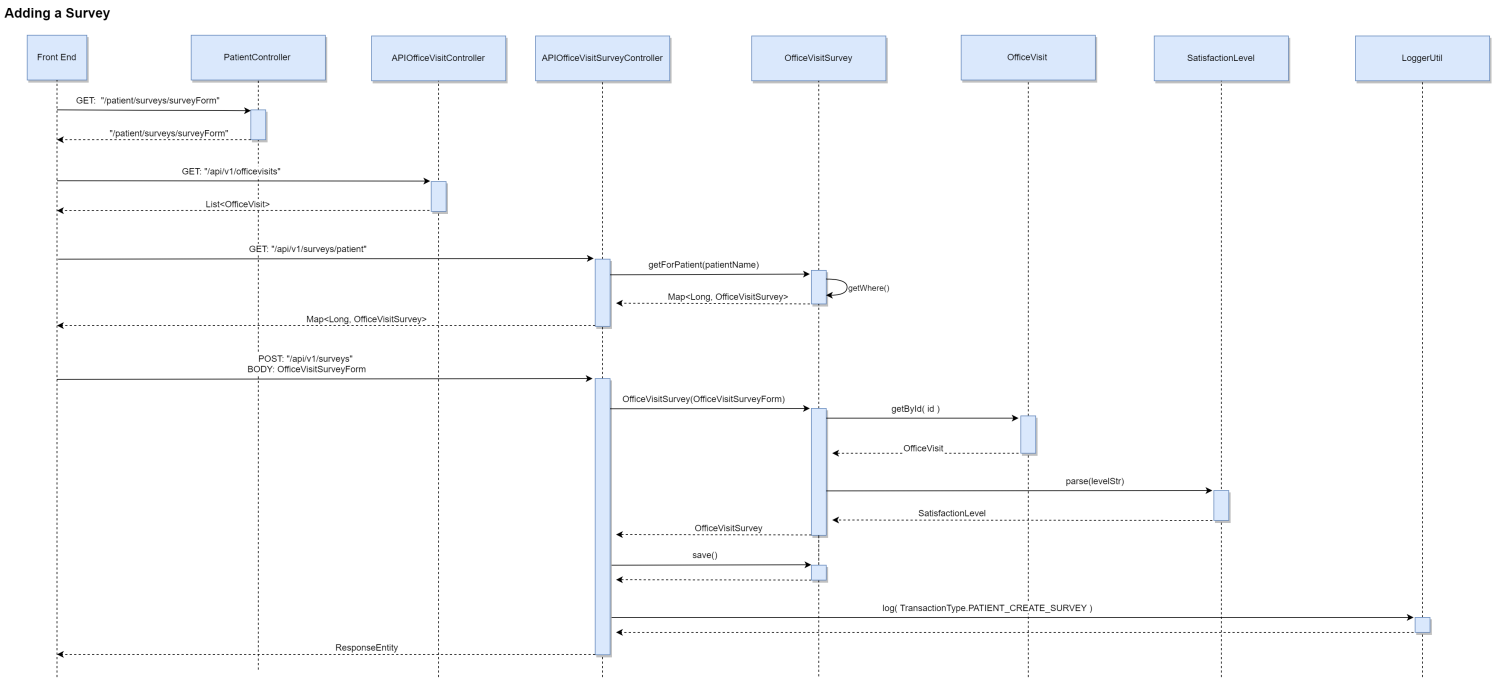
- Added `hcp` field to `OfficeVisitSurvey` with type `User`
- Added `id` field to `OfficeVisitSurvey` with type `Long`
- Added `signalSurveyViewed()` endpoint to `APIOfficeVisitSurveyController`
- Added new transaction types to `TransactionType`
- Changed return type of APIEnumController's `getSatisfactionLevels()` from `List<SatisfactionLevels>` to `List<Map<String, Object>>`
- Changed `HCPSurveyResults` from using public fields to using getters/setters

Sequence Diagram

Original Design



Updated Design



Changelog