T-813 Final Documentation

Group Members:

Simon Yu (jundayu2)

Jack Yang (jlyang2)

Alan Zhu (azhu8)

Kuo Bao (kuobao2)

Bin Peng (binpeng2)

Zhenglun Chen (zchen142)

David Zhou (mzhou32)

Bayani Julian (bjulia2)

Table:

Description of the iTrust Project:

iTrust is a role-based healthcare application that provides patients with a means to keep up with their medical history and records as well as communicate with their doctors, including selecting which doctors to be their primary caregiver, seeing and sharing satisfaction results, and other tasks. iTrust is also an interface for medical staff from various locations. iTrust allows the staff to keep track of their patients through messaging capabilities, scheduling of office visits, diagnoses, prescribing medication, ordering and viewing lab results, among other functions.

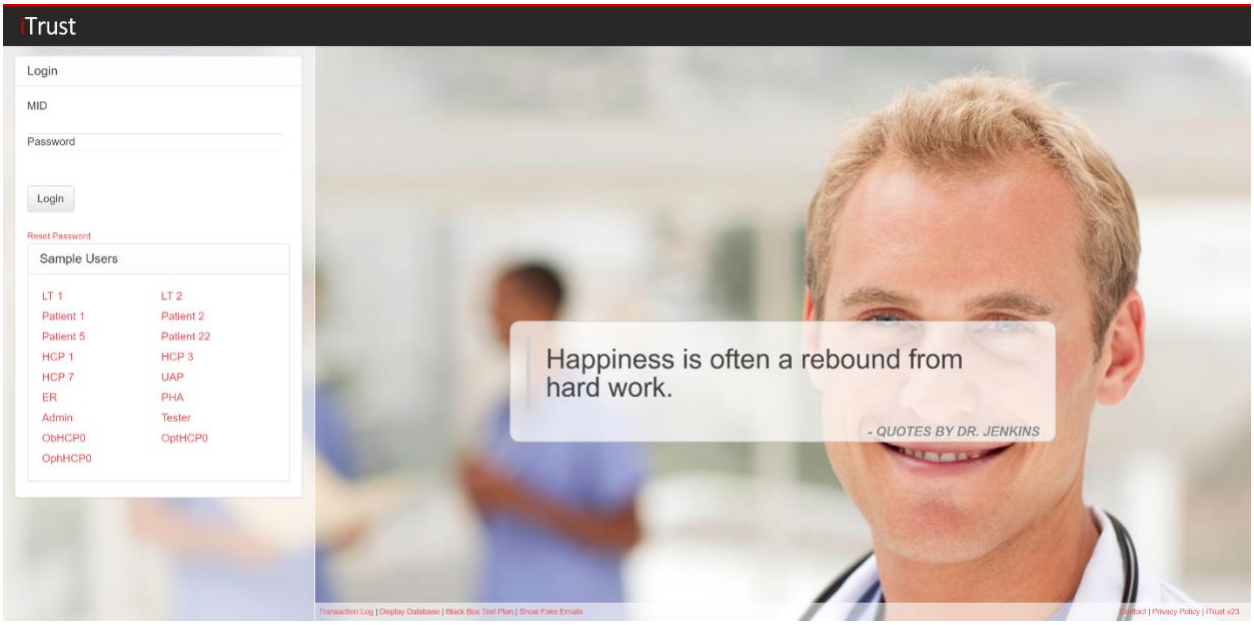

Figure 1: Login Page

Description of Feature:

We have developed new features for iTrust in the team project. We added a new obstetrics and gynecology (OB/GYN) module (including four use cases UC 93, UC 94, UC 95, UC 96) in addition to the new use case designed and implemented by our team to iTrust.

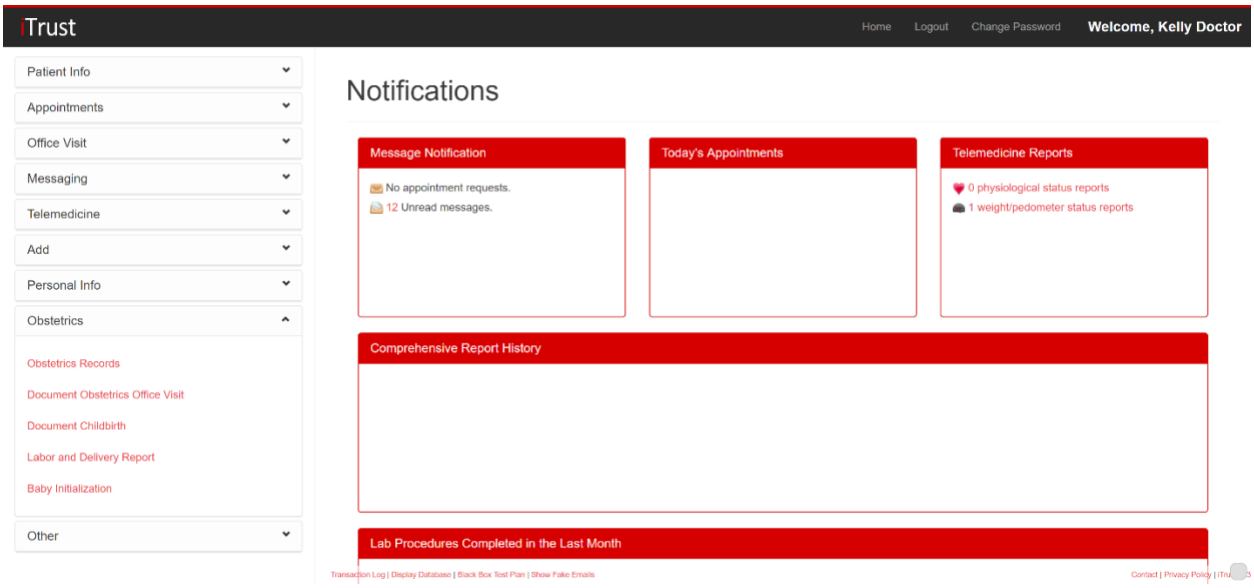

Figure 2: Obstetric Interface

There are five options in the obstetrics menu: 1. Obstetrics Records; 2. Document Obstetric Office Visit; 3. Document Childbirth; 4. Labor and Delivery Report; 5. Baby Initialization.

Obstetrics Records helps user search a patient, select an obstetrics record to modify or create new obstetrics records, and finally save all changes to database. It provides information about patient's Obstetric initialization data and prior pregnancy data for user to check or modify. (UC93)

Document Obstetric Office Visit makes user able to check or modify office visit record of an obstetric patient. User can have a look at the Office Visit Overview as well as the Ultrasound Records. Besides, after one office visit, the next office visit will be automatically scheduled. (UC94)

Document Childbirth helps the user check all information about the most recent obstetric records, and prior pregnancy data. Then the user can create child birth visit, set information about this child birth, and save all to database. (UC96)

Labor and Delivery Report will show all information (from obstetric to delivery) of this delivery on a table. The whole process of this delivery will be shown on the overview. (UC95)

The last feature, Baby Initialization, allows only OB/GYN HCP to use. Using it, HCP can initialize the baby as a new patient in our database by filling only 3 grids. All other information will be automatically saved. (New UC)

Overall Architecture and Design:

We used the Model-View-Controller concept to divide work to pairs. We had one pair for model, one pair for view, one pair for controller and one administration pair for planning, managing and merging the other pairs' code. Each pair first wrote the basic JUnit Tests and then wrote the code to pass the tests. We also incrementally wrote Junit tests to reach above 80% statement coverage for each non-GUI class. While merging each pair's code, the controller and view parts are implemented together by JavaScript and Java Servlet.

Model part is in iTrust/src/main/edu/ncsc/csc/itrust/model/old. It is in duty of creating structs for the need of front end. Beans and DAO are what we especially add most. Beans represent object of patients, records, office visit or any other stuff. They can save many information like MID, recordID, etc. And support get() and set() any stuff inside a bean. DAO is helping interact with database. It uses mysql sentences in its functions so that we can insert or update rows in database. Besides, we have also a loader class which helps us generate mysql sentences. It can accept a bean as a parameter and automatically merge information in the bean to mysql sentences.

Controller part is in iTrust/src/main/edu/ncsc/csc/itrust/action. It is written in java in responsible of connecting front end and back end. The front end can call function in controller to get any information in database very easily and clearly. To be specific, each use case has two controller files: viewAction and editAction. As their names claim, viewAction provides helper functions for viewing any information, and editAction helps adding or editing any information. Some very tricky problems are solved in this part very concisely, like the Google Calendar feature.

View part is in iTrust/WebRoot/auth. It is written in various languages like JSP, JavaScript, Ajax, and CSS. The job of this part is to show everything in a good appearance, create buttons so that users can interact with databases. In addition, we have created our own style to make the UI looks better.

Additionally, we have also modified createTable to make sure all data in MySQL will be correct after initialization. In the following few paragraphs, we will list all files we have created according to which part they belong to (Model, Controller, View). In addition, we will also introduce some important files specifically.

Use Case Architecture and Design:

Use Case 93

The following tables shows the relevant files and their code coverage percentages:

Model	Code Coverage Percentage
ObsteticInitBean	88.3
PriorPregnancyBean	94.6
ObsteticInitLoader	100
PriorPregnancyLoader	100
ObsteticInitDAO	84.4
PriorPregnancyDAO	84.6

Contoller	Code Coverage Percentage
ViewObstetricCareRecordsAction	89.8

View	Pass
viewObstetricCareRecordList	True

Figure 3: Code Coverage

The implementation of UC 93 is broken down to model, controller, view part separately. ObstetricInitBean saves necessary data like patientMID for the need of other parts. ObstetricInitDAO helps we modify and add data in database. In controller part, file viewObstetricAction, editObstetricAction provides useful helper function for view part to get data. Finally, in viewObstetricRecord and editObstetricRecord, we use CSS to create a professional appearance and functions from other parts to get data. To make interaction more naturally, we use JavaScript to create popup on buttons like “Delete”.

Use Case 94

The following tables shows the relevant files and their code coverage percentages:

Model	Code Coverage Percentage
ObstetricOfficeVisitBean	83.2
UltrasoundRecordBean	100
ObstetricOfficeVisitLoader	91.6
UltrasoundRecordLoader	100

ObstetricOfficeVisitDAO	81.6
UltrasoundRecordDAO	81.9

Controller	Code Coverage Percentage
ViewObstetricOfficeVisitAction	96.1

View	Pass
viewObstetricOfficeVisit	True

Figure 4: Code Coverage

The structure of UC 94 is similar to UC 93 which is separated to 3 parts. Model files support interactions with database and View part displays everything. One thing to mention is that we have written the support function for front end in Servlet, but also the trickiest google calendar part in Action file.

Use Case 96

The following tables shows the relevant files and their code coverage percentages:

Model	Code Coverage Percentage
ChildBirthVisitBean	95.1
BabyDeliveryInfoBean	92.8
ChildBirthVisitLoader	100
BabyDeliveryInfoLoader	100
ChildBirthVisitDAO	82.2
BabyDeliveryInfoDAO	81.7

Controller	Code Coverage Percentage
ChildBirthVisitAction	86.3

View	Pass
viewChildlBirthVisit	True

Figure 5: Code Coverage

We keep using the logic structure and style of previous use case. But in this part, we have merged some controller and view together in viewChildBirthVisit and editChildBirthVisit and related

Servlets. By using JavaScript and Servlet instead of traditional JSP, we have created different effect.

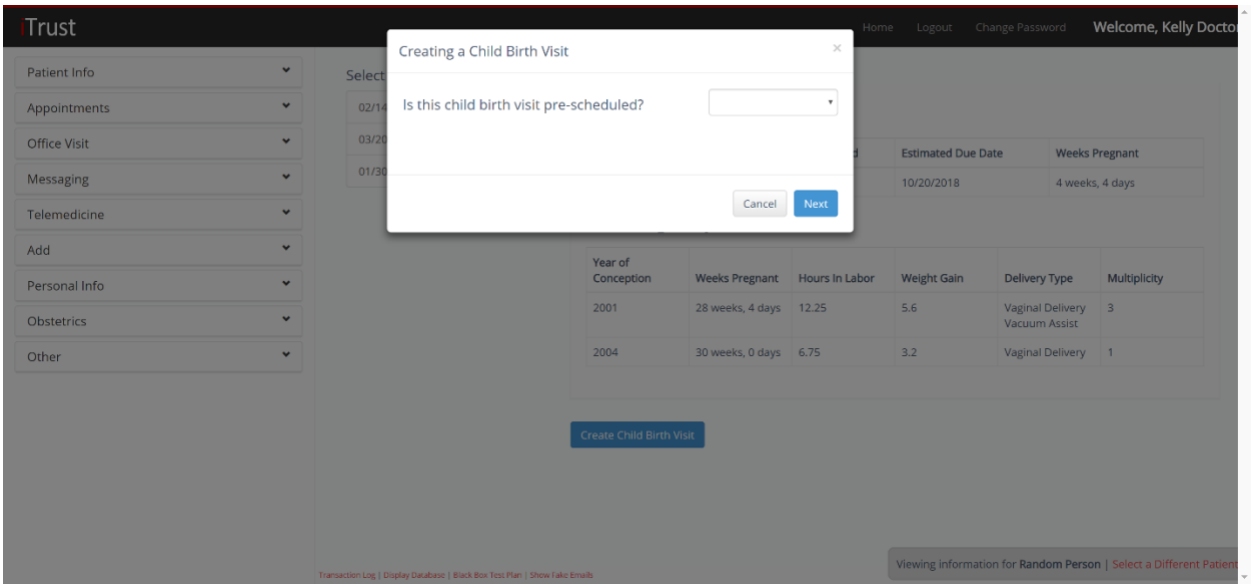

Figure 3: Popup Questionnaire

Use Case 95

The following tables shows the relevant files and their code coverage percentages:

Controller	Code Coverage Percentage
LaborDeliveryReportAction	80.4

View	Pass
viewLaborDeliveryReport	True

Figure 6: Code Coverage

The structure of this part is relatively simple compared to other parts since we can just use all beans and controllers form previous use case. We focus most on view part in this use case. (ViewLabourDeliveryReport)

Use Case Baby

The following tables shows the relevant files and their code coverage percentages:

Controller	Code Coverage Percentage
AddBabyInfoAction	85.3

View	Code Coverage Percentage
editBabyInitialization	True

Figure 7: Code Coverage

To meet the need of use case baby, we have modified the Beans and Servlet of Use Case 96 by adding a variable called patientMID. Then we created the view file called editBabyInitialization and the controller file called AddBabyInfoAction which provided helper function called getBabiesByMotherMID and updateBabyInfo in order to help front end interact with database.

Appendix:

When testing our interface, please make sure to click the "Save" button on each individual sections before clicking the overall "Finish" button; otherwise, the entered information may not be registered and recorded into the database. We have made sure to have at 80 percent code coverage for all the white box tests that we have written, and the detailed code coverage percentage is listed in the following sections and can be verified by running the respective test packages. The tests for the views are under edu.ncsu.csc.itrust.selenium package. The tests for the controllers and actions are under edu.ncsu.csc.itrust.unit.action package. The tests for DAOs and loaders are under edu.ncsu.csc.itrust.unit.dao.obstetrics package. The tests for beans are under edu.ncsu.csc.itrust.unit.bean package.

The manually testing instructions for each Use Case:

Figure 8: Left Menu Bar

On the left menu bar, there are five options for UC 93, 94, 95, 96, and new use case under "Obstetrics". The user has at least to be an HCP to view all the records and has OB/GYN to edit or create records. If a user wants to modify information in tables like this:

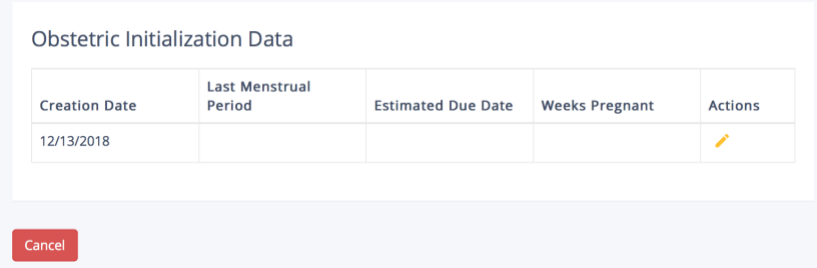

Figure 9: Obstetric Initialization Record

The user has to click the yellow pencil icon under Actions first. Then the table will look like this:

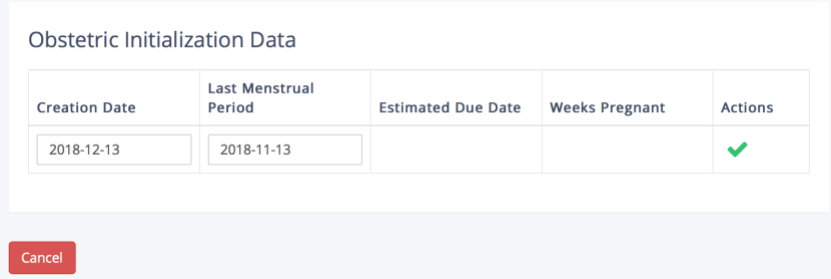

Figure 10: Obstetric Initialization Record

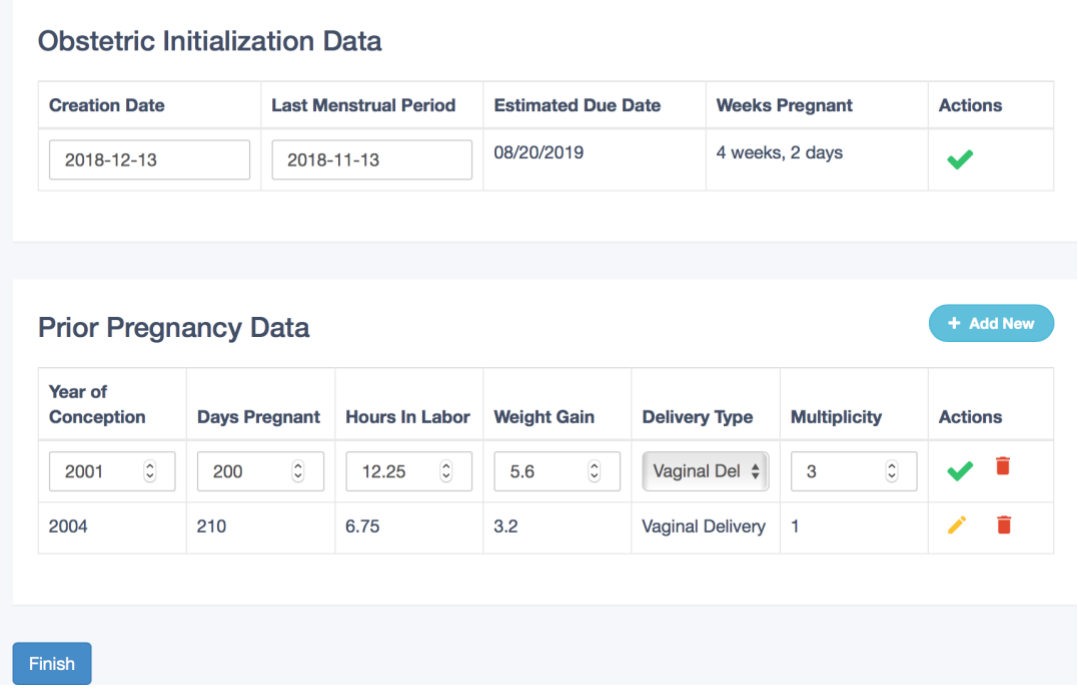

Figure 11: Obstetric Initialization and Prior Pregnancy Record

After putting down the information, the user needs to click the green check icon to save information and then click “Finish” under the table if there is one.

Use Case 93:

When the user clicks the “Obstetrics Records” for the first time, they will be directed to a page to choose a patient. Since we have sufficient data for patient number 1, it is better to test with patientMID = 1. On the page of “Obstetrics Records”, there is a list of dates of records corresponding the patient we previously chose (patientMID = 1), and it is ordered by the most recent date.

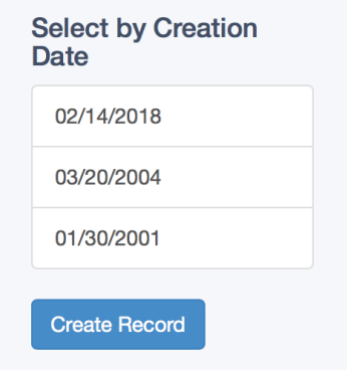

Figure 12: Dates of Obstetric Records of the Selected Patient

To view the tables of obstetrics and prior pregnancies, the user can simply click the date in the list. If the user is of OB/GYN and wants to create a new record, they can click the “create record” button under the list of dates. On this page, the user needs to click the yellow pencil icon under the “Actions”, then they can put in the “Last Menstrual Period” that must be earlier than the “Creation Date”. Then the user should click the green check icon to finish adding the new record.

For editing a record, the user just needs to click the “Edit Record” button under the tables for each record, and the next procedures will be the same as clicking the yellow pencil to modify data in the table and clicking the green check icon. Once the user finishes editing, they needs to click “Finish” to finish the editing.

Use Case 94:

The user needs to click “Document Obstetrics Office Visit”, and the user has to be OB/GYN to create and edit office visit records. After clicking the “Create Record” button under the list of dates on the left, the user needs to put down information of this office visit of the patient, such as office visit date, weight gain, blood pressure and etc. Also, under the table of “Office Visit Overview”, there is a “Ultrasound Records” table where the user can put down related information and upload an image. Once the user has put down the information, they have to click the “save” button and then click “Finish” button on the very bottom.

If the patient is of RH but does not have RH Immunization, and their current pregnancy term has past 28 weeks, the following message is displayed on each Obstetric Office Visit when selected: "Patient should be given an RH immune globulin shot if they have not already."

If the user clicks a record that has related information in the tables, “Office Visit Overview” and “Ultrasound Records”, they can click the modify data in the “Office Visit Overview” table and then

click “save” button. Then they can click “Add New” to add a new Ultrasound Record, click “Edit” or “Delete” to modify information of each record or delete that record.

Use Case 96:

The user needs to click “Document Childbirth” on the left. Then, if the user clicks one of the lists of dates, the corresponding tables “Obstetrics Initialization Data” and “Prior Pregnancy Data” related will show up. Under these two tables, if the there is no childbirth visit record for that selected obstetrics record, there will be a “Create Visit” on the bottom; Otherwise, there will be a table of childbirth information with an “Edit Visit” button on the bottom. If the user clicks the “Create Visit” bottom, a modal interface will pop up and allow the user to select if this childbirth visit is pre-scheduled or emergency. Then it will direct to page that is same as editing where user can change the type of preferred delivery type and modify the drug administered during delivery and baby delivery information including birth date, delivery type, and sex.

Use Case 95:

The user needs to click “Labor and Delivery Report” on the left menu bar, and the page will just display the relevant information of pregnancy of the selected patient, including Information for each past pregnancy, Estimated delivery date, Blood type, Obstetrics Office Visit Information, Pregnancy complications, and drug allergies.

Use Case Baby:

What this use case does is in fact adding new-born babies as patients and automatically filling new-born baby’s information to database. All social information about the baby will be the same as their parents. A registered HCP has logged in iTrust, and after a new baby delivery (we have created child birth visit in Document Childbirth), the HCP can click the Baby Initialization (The last item in the Obstetrics side bar). After clicking that, our controller will detect whether there is a baby born recently. If no baby has been born by the patient selected, a message “Please choose a mother who has just born babies.” If we have created a child birth visit in using Document Childbirth, a short form will be displayed under the Baby Initialization with three fields.

The father’s MID represents the MID of the baby’s father and will be associated with the baby’s patient information in the database. The other two fields let HCP to enter the name of the baby. After clicking the submit button, the alert message “Success!” will be displayed indicating the success of the operation. And after filling and submitting the necessary information, the baby will be initialized as a patient and all information will be automatically filled in. The created baby information can be verified by going to the “Display Database” page and search for the name of the baby. The information about the baby should be stored under the “patients” table with their name and father’s MID recorded.