

CSE 4283 / 6283

Software Testing and QA

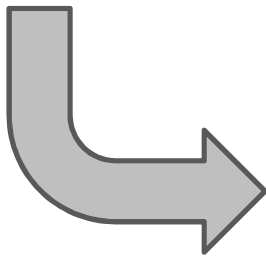
Dr. Tanmay Bhowmik
tbhowmik@cse.msstate.edu

Special thanks to Dr. Nan Niu & Dr. Byron Williams

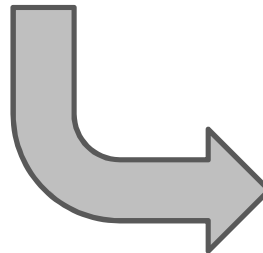


Agenda

Last Topic:
Testability



This Topic:
Requirement Based Testing (RBT)



Next Topic:
Testability



To Do



- Quiz-1
 - Feb 8, online, during the first 10 mins of the class
 - Covers up to the class of Feb 6

Picture source: Internet

3



Recap: what're requirements?

- Requirements = **stakeholders'** needs and desires
- Stakeholders = those who have a stake in the change being considered & who stand to (directly) gain or lose from the change



Recap: what's RE & why?

- RE is about:
 - Discovering stakeholders' needs & desires
 - Adjusting stakeholder expectations
 - Communicating these to system implementers
 - Adjusting implementer expectations
- Why RE:

defining "done"
timelines
budget
contractual (safety net)
feasibility

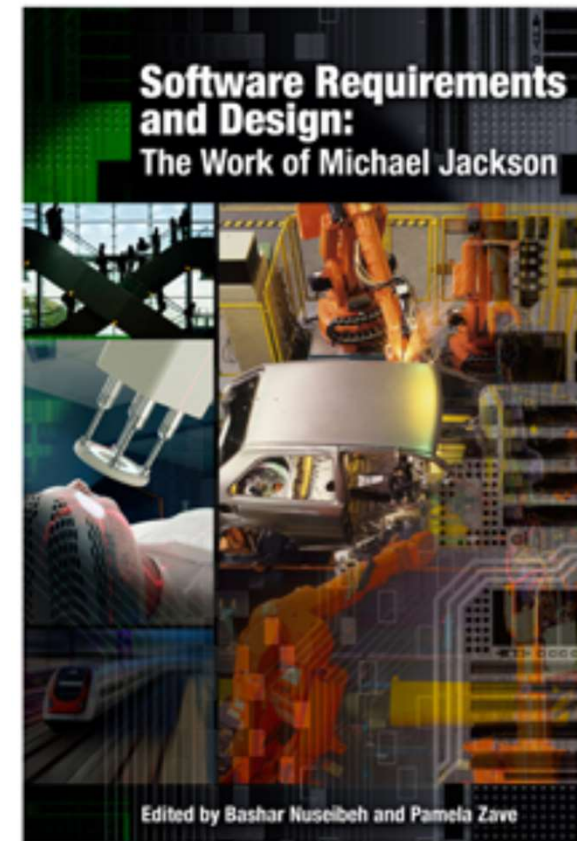


The Meaning of Requirements

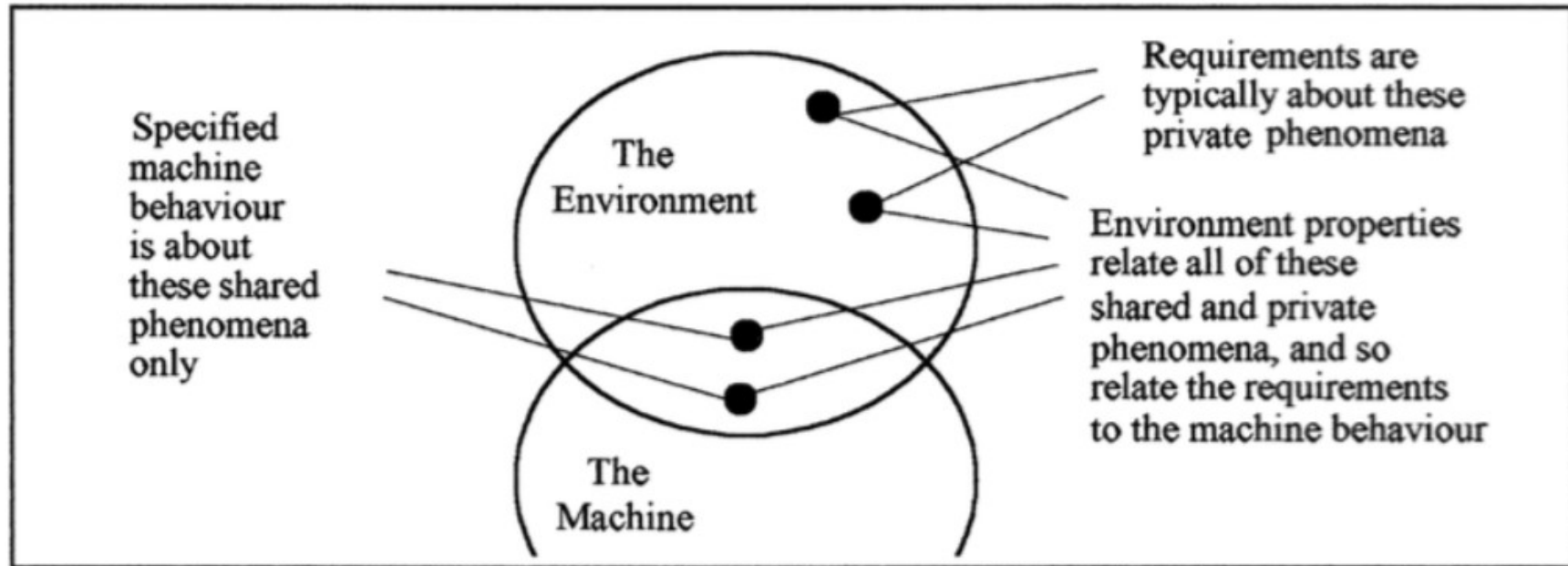
Software Requirements and Design: A Tribute to Michael Jackson



Michael Jackson (not the singer)



Understanding R, ϵ , S



R: requirements (optative/desired)

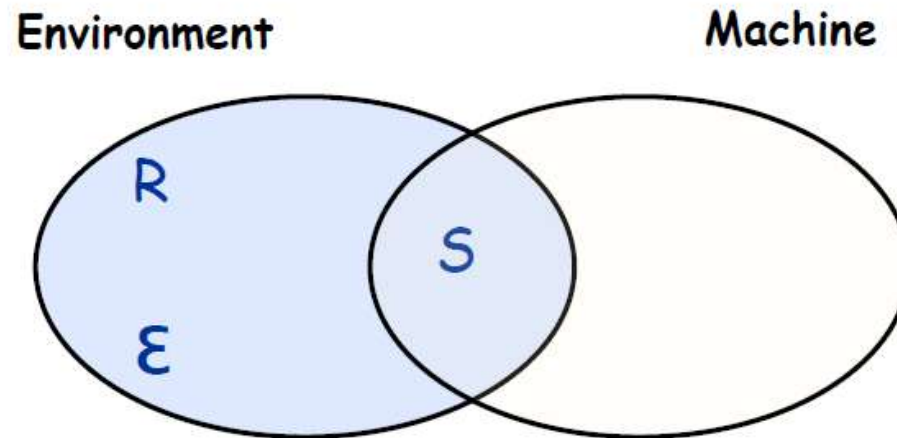
ϵ : environment assertions (indicative/given)

S: specifications (optative/desired)



Understanding R, ϵ , S

- Jackson defines the meaning of requirements as:



$\epsilon, S \vdash R$

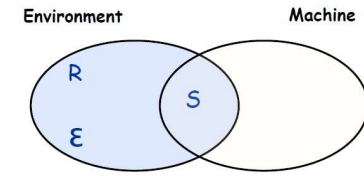
← THE meaning of req.s

Understanding R, ϵ , S

- Jackson defines the meaning of requirements as:

- ϵ

- things in the environment that are true whether or not we ever build the proposed system



$\epsilon, S \vdash R$ \leftarrow THE meaning of req.s

- R

- things in the environment that we wish to be made true by delivering the proposed system
 - Many of which will involve phenomena to which the machine has no access

- S

- is a precise description of the program's behavior in order to meet the requirements
 - Can only be written in terms of shared phenomena!



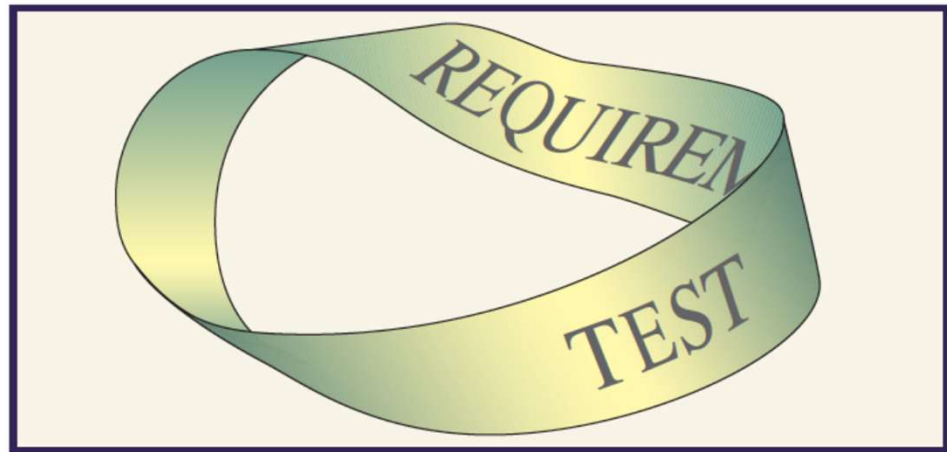
Example

- **R**
 - “The lab machine shall be accessible by only authorized personnel”
- **ϵ**
 - Auth. person has username
 - Auth. person has password
 - Passwords are never shared with non-authorized personnel
- **S**
 - Access to the lab machine shall be granted only after the user types an authorized “username, password” pair
- **S + ϵ entail R: What does it mean?**



Requirements = Tests

- Can you connect?
 - As formality increases, tests and requirements become indistinguishable. At the limit, they're the equivalent.



- Guiding practice
 - For most (if not all) software teams, the passing of acceptance tests, as opposed to an examination of the requirements, is the final criterion for shipping a system.

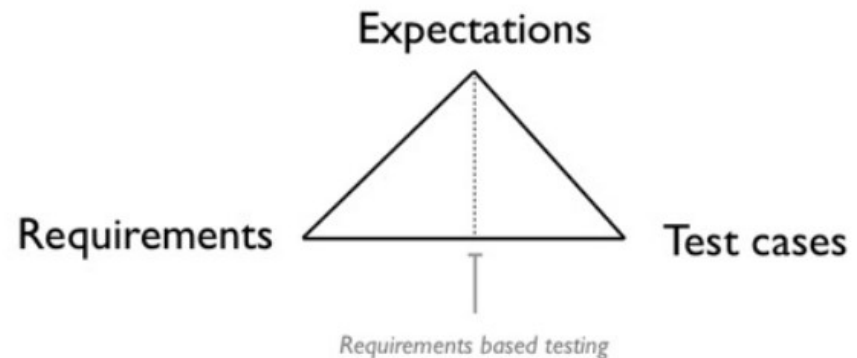
Picture source: Internet

11



Requirements-Based Testing (RBT)

- Addresses 2 major issues:
 - Check/validate the requirements for:
 - Correctness
 - Completeness
 - Ambiguity
 - Logical Consistency



- Designing a set of test cases
 - For those requirements
 - From a black-box perspective

Picture source: Internet

12



Testing: RBT Approach

1. Validate requirement

- a) Validate against business objectives (Usefulness, Novelty)
- b) Ambiguity analysis / Domain Expert review (Clarity, ϵ)

**Req.
Quality**

2. Define test completion criteria

3. Design test cases (define logical test cases)

- a) Structure/Formalize requirements
- b) Build/Define test cases

**Design
Test Cases**

4. Execute tests

5. Verify test results (pass/fail/improvement)

- a) Pass, fail, describe requirement improvement with justification

**Execute,
Outcome,
Expectation**

6. Verify test coverage

7. Manage and track defects/improvements

8. Manage the test library

Maintain

13



Testing: RBT Approach

1. Validate requirement

- a) Validate against business objectives (Usefulness, Novelty)
- b) Ambiguity analysis / Domain Expert review (Clarity, ϵ)

Req.
Quality

2. Define test completion criteria

3. Design test cases (define logical test cases)

- a) Structure/Formalize requirements
- b) Build/Define test cases

Design
Test Cases

4. Execute tests

5. Verify test results (pass/fail/improvement)

- a) Pass, fail, describe requirement improvement with justification

Execute,
Outcome,
Expectation

6. Verify test coverage

7. Manage and track defects/improvements

8. Manage the test library

Maintain

14



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com
- Validate requirements against business objectives
 - But How?

Source: Internet 15



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
 - **Description:** A user should be able to successfully login at gmail.com
 - **Validate requirements against business objectives**
 - **But How?**
 - Think about “Usefulness”
 - Very High?
 - Think about “Novelty”
 - Very Low?
- Likert Scale:
- 1: Very Low
 - 2: Low
 - 3: Medium
 - 4: High
 - 5: Very High



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com
- Perform an initial ambiguity review
 - Think about “Clarity” (Likert Scale)
- Perform domain expert review
 - But How?
 - Think about “Correctness and Completeness” → How?
 - Think about “**Environment Assertions**”
 - User is already registered
 - The web browser supports accessing email clients



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com
- Define completion criteria
 - ??????



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com
- **Structure and formalize requirement**
 - **Description:** A registered user should be able to successfully login at gmail.com.
 - **Precondition:** the user must already be registered with an email address and password.
 - **Assumption:** a supported browser is being used.



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com
- **Design test cases**
 - Using a supporting browser, navigate to gmail.com
 - In the 'email' field, enter the email of the registered user.
 - Click the 'Next' button.
 - Enter the password of the registered user
 - Click 'Sign In'



The RTB Methodology: An Example

- **Title:** Login Page – Authenticate Successfully on gmail.com
- **Description:** A user should be able to successfully login at gmail.com.
- Execute tests
- Verify test results (pass/fail/improvement)
 - Pass?



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2]. The resulting list should include the following for each access:
 - **Name of non-DLHCP accessor** (with a link to contact information if the viewer is an LHCP)
 - **Role of non-DLHCP accessor** relative to the patient
 - **Date and time of access**
 - **Transaction Type**
- [S1] By default, the patient is presented with a list sorted by dates, most recent access first.
- [S2] The patient may choose to view the list **sorted** by the **role** of the accessor relative to the patient (personal health representative, LHCP, UAP, Emergency Responder; any order is fine as long as the list is sorted by role) as well as by **date for each role type**, most recent access first.



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2].
- **Validate requirements against business objectives**
 - ????



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2].
- Perform an initial ambiguity review
- Perform domain expert review
 - What are some environment assertions?
 - Multiple accessor may have same name
 - Same accessor may have multiple roles
 - A patient may have a long history
 - A working/supporting browser is available



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2].
- **Define completion criteria**
 - ????



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2].
- **Structure and formalize requirement**
 - Skip for now



RBT: An example using iTrust use cases

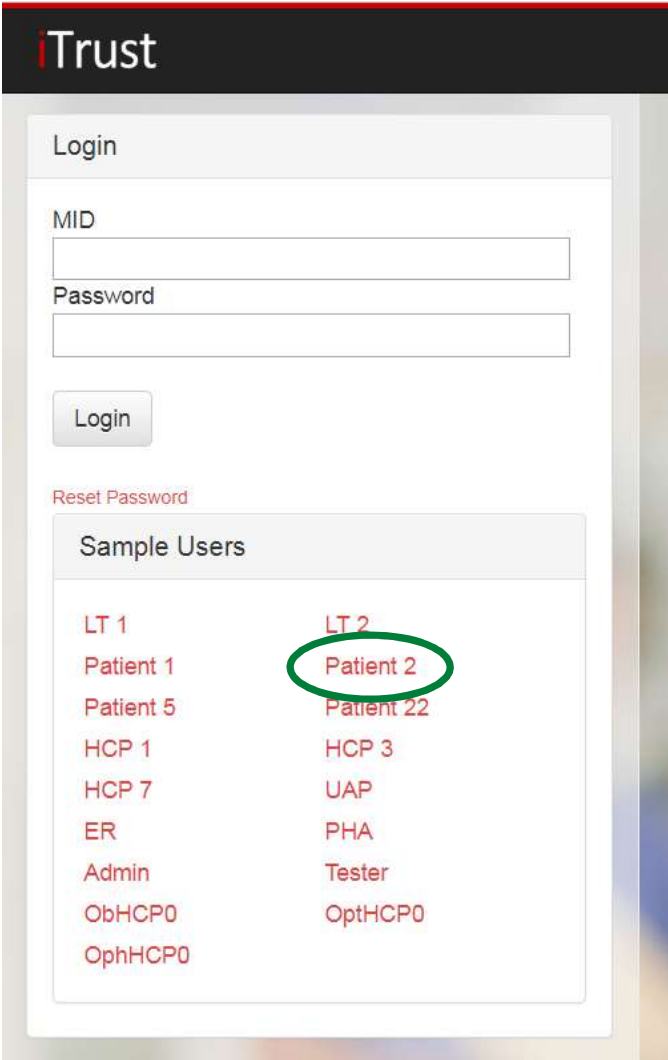
- **Title:** View Access Log
- **Design test cases**
 - Using a supporting browser, log-in to iTrust as a patient
 - Click “View → Access Log”
 - Check the column headers in the log for Name, Role, Date and time of access, Transaction Type
 - Sort on “Date”
 - Sort on “Role”
 - Type “Start Date”, “End Date” and click “Filter”
 - Select a different name, if any, from “View log for” and click “Filter”. Do the same changing the dates as well.

(Check the outcome at every stage and note down if anything is wrong)



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Execute tests:** Using a supporting browser, log-in to iTrust as a patient



The image shows a screenshot of the iTrust web application interface. At the top, there is a dark header with the 'iTrust' logo. Below the header, the main content area is titled 'Login'. It contains two input fields: 'MID' and 'Password', each with a corresponding text box. Below these fields is a 'Login' button. Underneath the login section, there is a link for 'Reset Password'. Further down, there is a section titled 'Sample Users' which displays a list of user roles and names in two columns. The roles listed are LT 1, LT 2, Patient 1, Patient 5, HCP 1, HCP 7, ER, Admin, ObHCP0, and OphHCP0 in the first column; and LT 2, Patient 2, Patient 22, HCP 3, UAP, PHA, Tester, and OptHCP0 in the second column. The text 'Patient 2' is circled in green.

Sample Users	
LT 1	LT 2
Patient 1	Patient 2
Patient 5	Patient 22
HCP 1	HCP 3
HCP 7	UAP
ER	PHA
Admin	Tester
ObHCP0	OptHCP0
OphHCP0	

28



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Execute tests:** Click “View → Access Log”

The screenshot displays the iTrust web application interface. On the left, a sidebar menu is visible with a dark header containing the 'iTrust' logo. Below the header, there are two expandable/collapsible sections: 'Edit' (collapsed) and 'View' (expanded). The 'View' section contains a list of links: 'Access Log', 'Full Calendar', 'Find an Expert', 'My Bills', 'View My Records', 'View My Office Visits', 'View My Prescription Report', and 'View Representee Prescription Report'. The 'Access Log' link is highlighted with a green oval. On the right side of the interface, there is a 'Notifications' section. It features a red header 'Message Notification' followed by a list of status messages: '1 Unread message(s)', 'No unpaid bills.', 'No denied insurance claims.', and 'No approved insurance claims.'. Below this, there is another red header 'Your Upcoming Appointments' followed by a list of appointments: '06/04/2020 - Consultation' and '10/14/2020 - Colonoscopy'.



RBT: An example using iTrust use cases

- **Title:** View Access Log
 - Name of accessor
 - Role of accessor
 - Date and time of access
 - Transaction Type
- **Execute tests:** Check column headers: Name, Role, Date & time, Transaction Type

Viewing Log For: Andy Programmer

Date	Accessor	Role	Description
2008-11-14 09:32:00.0	Justin Time	Emergency Responder	View emergency report
2008-09-07 16:30:00.0	Kelly Doctor	LHCP	Edit Office Visits
2008-03-04 10:15:00.0	Kelly Doctor	LHCP	View risk factors

View log for:

Start Date: End Date:

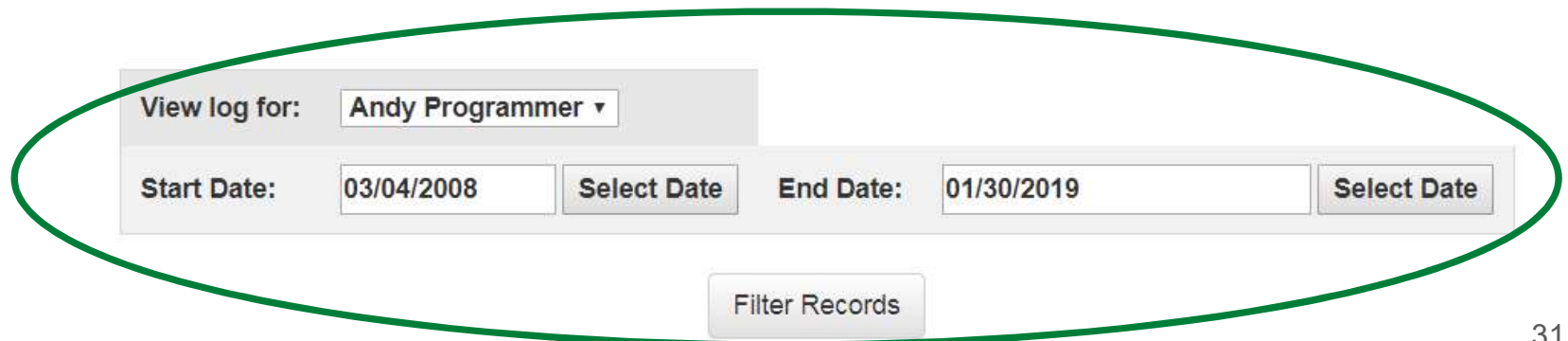


RBT: An example using iTrust use cases

- **Title:** View Access Log
 - **Execute tests:** Sort on “Date”, “Role”, Type “Start Date”, “End Date” → click “Filter”; Select different name, if any, from “View log for” and click “Filter”. Do the same changing the dates as well.

Viewing Log For: Andy Programmer

Date	Accessor	Role	Description
2008-11-14 09:32:00.0	Justin Time	Emergency Responder	View emergency report
2008-09-07 16:30:00.0	Kelly Doctor	LHCP	Edit Office Visits
2008-03-04 10:15:00.0	Kelly Doctor	LHCP	View risk factors



View log for:

Start Date: End Date:



RBT: An example using iTrust use cases

- **Title:** View Access Log
- **Description:** The patient chooses to view his or her access log or that for a person for whom they are a personal health representative. The patient then chooses the beginning and end date for the period of time they would like to view their access log for [S1, S2].
- **Verify test results (pass/fail/improvement)**
 - **Failed**
 - Transaction type do not appear in column header names
 - Name, Role, Date and time of access, Transaction Type in the log are not shown in the order indicated
 - **Improvement:** Patient should be able to sort the log based on accessor name as well
 - Justification: Patient may have a long history, therefore a log with many accessors and roles. Multiple accessor may have same name. Same accessor may have multiple roles. Patient may be interested to find a specific accessor, not just the role or date.



Assignment on RBT

1. **Validate requirement**

- a) Validate against business objectives
- b) Ambiguity analysis / Domain Expert review

2. Define test completion criteria

3. **Design test cases (define logical test cases)**

- a) Structure/Formalize requirements
- b) Define test cases

4. **Execute tests**

5. **Verify test results (pass/fail/improvement)**

- a) Pass, fail, describe requirement improvement with justification

6. Verify test coverage

7. Manage and track defects/improvements

8. Manage the test library

Overall,
conduct
these
four
activities
along
given
use case



THANK YOU



34



MISSISSIPPI STATE UNIVERSITY

TANMAY BHOWMIK

COMPUTER SCIENCE AND ENGINEERING