FOUNDATION
MEDICINE

Appendix B - CRM FooBar Post-Execution Report

Table of Contents

Issue Icon Key					
Jira Issues	⚡	Epic	QTest Issues	⊞	Test Case
	🔖	Story		⬚	Test Step
	○	Bug			

Traceability Matrix

Epic ID	Epic Summary	Issue ID	Issue Type	Issue Summary	Test Case ID(s)
CRMFB-44	Mock: Node Upgrade	CRMFB-156	Bug	MOCK: Bug in Progress	
CRMFB-44	Mock: Node Upgrade	CRMFB-60	Story	Mock Story	
CRMFB-49	MOCK: Critical Automation	CRMFB-97	Story	MOCK: I want to create a regular user physician account as a CSR/administrator	
CRMFB-49	MOCK: Critical Automation	CRMFB-96	Story	MOCK: I want to search for patient report by name and place a patient report in archive	
CRMFB-52	MOCK: Payer metadata for Billing Team	CRMFB-87	Story	MOCK: Billing metadata dashboard	
CRMFB-55	MOCK: Move to DevOps pipeline	CRMFB-145	Bug	MOCK: Backlog Bug	
CRMFB-55	MOCK: Move to DevOps pipeline	CRMFB-78	Story	MOCK: Migrate ICE backend	TC-33
CRMFB-55	MOCK: Move to DevOps pipeline	CRMFB-77	Story	MOCK: Migrate digital reporting	TC-34,TC-35
CRMFB-55	MOCK: Move to DevOps pipeline	CRMFB-75	Story	MOCK: Migrate dashboard client	
CRMFB-56	MOCK: Account Notification	CRMFB-79	Story	MOCK: Build the new notification emails	
CRMFB-56	MOCK: Account Notification	CRMFB-74	Story	MOCK: Retrieve data from CRM for ways to communicate with customer and set/update CRM with the changes	
CRMFB-56	MOCK: Account Notification	CRMFB-73	Story	MOCK: Build UI with proper notifications	
CRMFB-57	MOCK: Formulary Ordering	CRMFB-146	Bug	MOCK: Unable to complete Bug	
CRMFB-57	MOCK: Formulary Ordering	CRMFB-76	Story	MOCK: Implement new fields (both UI & PDF)	
CRMFB-57	MOCK: Formulary Ordering	CRMFB-71	Story	MOCK: Implement new logic	
CRMFB-57	MOCK: Formulary Ordering	CRMFB-69	Story	MOCK: Implement new UI experience	
CRMFB-58	MOCK: Secure Email Integration	CRMFB-143	Story	MOCK: Handle QTest Ids by Environment on Trace Matrix	
CRMFB-99	MOCK: QTest Attachments	CRMFB-153	Story	Mock: Story linked to 2 test cases	

CRMFB-99	MOCK: QTest Attachments	CRMFB-151	Story	Mock: Story 3 to a single test case	
CRMFB-99	MOCK: QTest Attachments	CRMFB-150	Story	Mock: Story 2 to a single test case	
CRMFB-99	MOCK: QTest Attachments	CRMFB-149	Story	Mock: Story 1 to a single test case	
CRMFB-99	MOCK: QTest Attachments	CRMFB-147	Bug	MOCK: Completed bug	
CRMFB-99	MOCK: QTest Attachments	CRMFB-144	Bug	MOCK: Bug in Report	
CRMFB-99	MOCK: QTest Attachments	CRMFB-102	Story	MOCK: QTest Multiple Attachments	
CRMFB-99	MOCK: QTest Attachments	CRMFB-101	Story	MOCK: QTest Test Step Attachments	
CRMFB-99	MOCK: QTest Attachments	CRMFB-100	Story	MOCK: QTest Case Attachments	
CRMFB-103	MOCK: Text Formatting	CRMFB-106	Story	MOCK: Test font formatting	
CRMFB-103	MOCK: Text Formatting	CRMFB-105	Story	MOCK: Test Long URLs within text boxes or modal windows	
CRMFB-103	MOCK: Text Formatting	CRMFB-104	Story	MOCK: Fields using markup numbering from Jira	
CRMFB-107	MOCK: qTest Formatting	CRMFB-154	Story	Mock: Prerequisites formatting	
CRMFB-107	MOCK: qTest Formatting	CRMFB-152	Story	Mock: Story 4 to a a single test case	
CRMFB-107	MOCK: qTest Formatting	CRMFB-141	Story	MOCK: qTest actual result field	TC-38
CRMFB-107	MOCK: qTest Formatting	CRMFB-140	Story	MOCK: qTest special characters validation	TC-37
CRMFB-107	MOCK: qTest Formatting	CRMFB-139	Story	MOCK: XML code in qTest	TC-36
CRMFB-107	MOCK: qTest Formatting	CRMFB-108	Story	MOCK: Test line breaks on test steps	TC-32
CRMFB-109	MOCK: Miscellaneous test data	CRMFB-110	Story	MOCK: Test issue with several test cases	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-120	Story	MOCK: Build out UI components for the Sample information	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-119	Story	MOCK: Ability to capture Copied Parties via the Order	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-118	Story	MOCK: Ability to add a support group to an Order	

CRMFB-137	MOCK: Specimen Procurement	CRMFB-117	Story	MOCK: Ability to view current "Hound" Sample Tracking of any TRF/CRF	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-116	Story	MOCK: Prepare LIMS patient data for migration into CRM	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-115	Story	MOCK: Ability to Procure a New Specimen via Existing Order	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-114	Story	MOCK: Patient Entity Refinement from Sprint #1	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-113	Story	MOCK: Indicate Third Party Agreement	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-112	Story	MOCK: Create Auto-Number for New Patient Records	
CRMFB-137	MOCK: Specimen Procurement	CRMFB-111	Story	MOCK: Duplicate data detection for Patient entity	
CRMFB-138	MOCK: House Account Management	CRMFB-136	Story	MOCK: Create a Patient via detailed entry	
CRMFB-138	MOCK: House Account Management	CRMFB-135	Story	MOCK: Quick-create a Patient	
CRMFB-138	MOCK: House Account Management	CRMFB-134	Story	MOCK: Search for a Patient	
CRMFB-138	MOCK: House Account Management	CRMFB-133	Story	MOCK: Update an existing Patient record	
CRMFB-138	MOCK: House Account Management	CRMFB-132	Story	MOCK: Set Patient entity status	
CRMFB-138	MOCK: House Account Management	CRMFB-131	Story	MOCK: Ability to attach a document to CRM entity	
CRMFB-138	MOCK: House Account Management	CRMFB-130	Story	MOCK: Ability to capture required metadata with attached documents	
CRMFB-138	MOCK: House Account Management	CRMFB-129	Story	MOCK: Ability to select document Type when attaching it to Patient record	
CRMFB-138	MOCK: House Account Management	CRMFB-128	Story	MOCK: Add new version to an attached document on CRM entity	
CRMFB-138	MOCK: House Account Management	CRMFB-127	Story	MOCK: Create Business Units for Security	

CRMFB-138	MOCK: House Account Management	CRMFB-126	Story	MOCK: Configure default views for Patient Entity	
CRMFB-138	MOCK: House Account Management	CRMFB-125	Story	MOCK: Ability to view existing attachements	
CRMFB-138	MOCK: House Account Management	CRMFB-124	Story	MOCK: Ability to share attached document across multiple entities	
CRMFB-138	MOCK: House Account Management	CRMFB-123	Story	MOCK: Create Order Entity in CRM v1	
CRMFB-138	MOCK: House Account Management	CRMFB-122	Story	MOCK: Ability to update an existing Order record	
CRMFB-138	MOCK: House Account Management	CRMFB-121	Story	MOCK: Ability to Search for a Order in the Global search	
		CRMFB-84	Story	Test	

Requirements

Epic: CRMFB-44

Summary: Mock: Node Upgrade

Description: Upgrade Node

Notes *testing* Nothing *testing*

Testing line with bold half way **through** works correctly.

Testing line with bold at the **end**

- Bullet point with **BOLD** text
- **Bold** point with *italic* text

Bold italic text ***here*** with text after

- ONO-225 Done has notes about what prep work and insight into what we should consider

Risks

- LowDash
- Mongoose
- Mongoose Q
- Doing this Story!

  - test

**Test with no closing asterisks

Assumptions

- DevOps will be doing Staging and PROD
- Regression Testing might reveal things our functional testing didn't catch

Dependencies

- DevOps Team

Acceptance Criteria:

Issues for CRMFB-44:

ID	Issue Type	Summary	Status
CRMFB-156	Bug	MOCK: Bug in Progress	In Progress
CRMFB-60	Story	Mock Story	Done

- -

⚡ CRMFB-44 > ◯ CRMFB-156

Bug: CRMFB-156

Summary: MOCK: Bug in Progress

Affects Version(s):

Environment:

Steps to Reproduce:

Actual Behavior:

Expected Behavior:

⚡ CRMFB-44 > ◻ CRMFB-60

Story: CRMFB-60

Summary:	Mock Story
Description:	Not a Real Story, Test purposes
Acceptance Criteria:	1. Acceptance criteria 1
	2. AC 2
	3. another AC
	4. last AC

Epic: CRMFB-49

Summary:	MOCK: Critical Automation
Description:	As a QE I want to automate the critical pathways for ICE so that we can save time testing and have confidence in our product.��

Acceptance Criteria:��

- demonstrate that the 9 critical pathways have been automated
- demonstrate that the tests pass��

Acceptance Criteria:

Issues for CRMFB-49:

ID	Issue Type	Summary	Status
CRMFB-97	Story	MOCK: I want to create a regular user physician account as a CSR/administrator	In Progress
CRMFB-96	Story	MOCK: I want to search for patient report by name and place a patient report in archive	To Do

⚡ CRMFB-49 > ◻ CRMFB-97

Story: CRMFB-97

Summary:	MOCK: I want to create a regular user physician account as a CSR/administrator
Description:	1. login with _CSR.email and _CSR.password
	2. click on the create new user butoon while being in the admin section of the header menu
	3. type in _user_new.email into the email address field
	4. type in _user_New.firstname into the first name field
	5. type in _user_New.lastname into the last name field
	6. type in _user_New.physicianID into the physician ID field
	7. type in _user_New.physicianID inot the physician ID list field
	8. type in _user_New.medicalfacilityID into the medical facility ID field
	9. Type in "-1" into the medical facility ID list field
	10. select "Clinician/MD" from the occupation drop down
	11. make sure the authorized check box is checked
	12. make sure the notify checkbox is checked
	13. click the add new user button
Acceptance Criteria:	demonstrate that we can create a regular user physician account as a CSR/Administrator.
	demonstrate that the test passes
	demonstrate that the user can login with the user name & password

⚡ CRMFB-49 > ◻ CRMFB-96

Story: CRMFB-96

Summary: MOCK: I want to search for patient report by name and place a patient report in archive

Description:
1. login with _user.email and _user.password
2. check the checkbox next to a patient name
3. click the archive button
4. click the archived tab to make sure the report is present
5. search for a patient name using the search box
6. click on the patient name to open a report
7. click the archive button
8. click all patients to return to the worklist
9. click the archived tab to make sure the report has been un-archived

Acceptance Criteria:

- -

Epic: CRMFB-52

Summary: MOCK: Payer metadata for Billing Team

Description: The Billing Team is looking for a place to store meta data regarding the Insurance Payers and Plans. The data is then used by BI to create reports for Finance.

Acceptance Criteria

- CRM is automatically updated on a daily basis with any changes, additions or deletions from the Quadax flat file. This file can be upwards of 6K rows.

Risks

- If implementing this into CRM takes too long, the data may need to be placed in another table that is accessible to BI reporting.

Dependencies

- Daily Quadax file; BI evaluate Quadax data and provide exceptions;

Assumptions

- PHI data will not be included in this data

Notes

- Text

Acceptance Criteria:

Issues for CRMFB-52:

ID	Issue Type	Summary	Status
CRMFB-87	Story	MOCK: Billing metadata dashboard	To Do

- -

⚡ CRMFB-52 > 🔖 CRMFB-87

Story: CRMFB-87

Summary: MOCK: Billing metadata dashboard

Description:

Acceptance Criteria:

- -

Epic: CRMFB-55

Summary: MOCK: Move to DevOps pipeline

Description:

Acceptance Criteria:

Issues for CRMFB-55:

ID	Issue Type	Summary	Status
CRMFB-145	Bug	MOCK: Backlog Bug	To Do
CRMFB-78	Story	MOCK: Migrate ICE backend	To Do
CRMFB-77	Story	MOCK: Migrate digital reporting	To Do
CRMFB-75	Story	MOCK: Migrate dashboard client	To Do

⚡ CRMFB-55 > ○ CRMFB-145

Bug: CRMFB-145

Summary: MOCK: Backlog Bug

Affects Version(s):

Environment:

Steps to Reproduce:

Actual Behavior:

Expected Behavior:

⚡ CRMFB-55 > ▯ CRMFB-78

Story: CRMFB-78

Summary: MOCK: Migrate ICE backend

Description:

Acceptance Criteria:

⚡ CRMFB-55 > ▯ CRMFB-77

Story: CRMFB-77

Summary: MOCK: Migrate digital reporting

Description:

Acceptance Criteria: x

⚡ CRMFB-55 > ▯ CRMFB-75

Story: CRMFB-75

Summary: MOCK: Migrate dashboard client

Description:

Acceptance Criteria:

Epic: CRMFB-56

Summary: MOCK: Account Notification

Description: CRM is source of truth, fax/phone/email/secure email/push...all in CRM

We're building UI, data will be in CRM

Stories:

1. UX - Design the Account Page

1. UX - Design the new notification(s)

1. Eng - Build the new notification email(s)

1. Eng - Build UI with proper notifications

1. Eng - Retrieve data from CRM for ways to communicate with customer and set/update CRM with the changes (server side)

Acceptance Criteria:

Issues for CRMFB-56:

ID	Issue Type	Summary	Status
CRMFB-79	Story	MOCK: Build the new notification emails	To Do
CRMFB-74	Story	MOCK: Retrieve data from CRM for ways to communicate with customer and set/update CRM with the changes	In Progress
CRMFB-73	Story	MOCK: Build UI with proper notifications	In Progress

- -

CRMFB-56 > CRMFB-79

Story: CRMFB-79

Summary: MOCK: Build the new notification emails

Description:

Acceptance Criteria:

- -

CRMFB-56 > CRMFB-74

Story: CRMFB-74

Summary: MOCK: Retrieve data from CRM for ways to communicate with customer and set/update CRM with the changes

Description:

Acceptance Criteria:

- -

CRMFB-56 > CRMFB-73

Story: CRMFB-73

Summary: MOCK: Build UI with proper notifications

Description:

Acceptance Criteria:

- -

Epic: CRMFB-57

Summary: MOCK: Formulary Ordering

Description: As a physician support staff I want to understand the difference between F1CDx and F1 so that I can understand the financial and clinical implications for the ordering physician patients
Value:

- Minimize patient OOP expenses
- Maximize FMI's reimbursement rate

What if we don't alternatives:

- Retroactively call customers and suggest alternative tests
- Continuing risking a suboptimal reimbursement rate

indication = ICD

Potential Fields the FDA could come back with = ICD, Insurance (Medicare Part B), Stage, ECOG, Prior Therapy, Test Selection

Notes:

- We need the business rules and the sources from the data
- We will have logic from the FDA in December.

Risks:

- Complex logic late in the game
- Holiday Season, capacity constrained
- Stakeholder Alignment (Pathology, Marketing, Billing)

To do from 9/8/17
Mike will document everything he knows here so we can share it with the tech and design team
Meeting with CRM to review timelines for direct integration (will inform potential design change for "receipt" of order)

Stories:

1. Product Spike - Understand the requirements and implications of what this would be
2. Design Spike - Behavior for surfacing alternate test info (5)
3. Design - Additional fields and behavior for messaging user about testing options(8)
4. Eng - Implement new fields (both UI and PDF) (5)
5. Eng - Implement new logic (5)
6. Eng - Implement new UI experience (5)

Acceptance Criteria:

Issues for CRMFB-57:

ID	Issue Type	Summary	Status
CRMFB-146	Bug	MOCK: Unable to complete Bug	Done
CRMFB-76	Story	MOCK: Implement new fields (both UI & PDF)	Done
CRMFB-71	Story	MOCK: Implement new logic	In Progress
CRMFB-69	Story	MOCK: Implement new UI experience	To Do

- -

⚡ CRMFB-57 > ◯ CRMFB-146

Bug: CRMFB-146

Summary: MOCK: Unable to complete Bug

Affects Version(s):

Environment:

Steps to Reproduce:

Actual Behavior:

Expected Behavior:

⌂ CRMFB-57 > ⌂ CRMFB-76

Story: CRMFB-76

Summary: MOCK: Implement new fields (both UI & PDF)

Description:

Acceptance Criteria:

⌂ CRMFB-57 > ⌂ CRMFB-71

Story: CRMFB-71

Summary: MOCK: Implement new logic

Description:

Acceptance Criteria:

⌂ CRMFB-57 > ⌂ CRMFB-69

Story: CRMFB-69

Summary: MOCK: Implement new UI experience

Description:

Acceptance Criteria:

Epic: CRMFB-58

Summary: MOCK: Secure Email Integration

Description:

As a customer I want to be able to be confident that my conversations with FMI via email are secure so that I don't have to worry about my patient's information getting out in the world
As FMI we want to centralize our password requirements, communication points and UI with customers to foster a seamless experience while still enabling CS to work well

What if we do:

• We can unify UI experience, reduce FMI password requirements, unify customer engagement points, send insights/sct information

What if we don't alternatives

• Continue to have two login points, UIs, password requirements and Tech Prod can send things via link with token to a PDF (Today)
• Explore consolidating Auth so customers don't need to remember two passwords, access points are unified, UIs are separate, send things via token to a PDF

Acceptance Criteria:

• Demonstrate that the solution is the same level of security that we have today
• Our applications can plug into secure email and leverage it to send information
• Users authenticate secure email with ICE credentials
• Users access secure email from the dashboard
• CS workflow

Notes:

• Current provider is Axway https://www.axway.com/en
• See Current Experience of receiving and viewing secure email attached
• Need to maintain the ability to use Mandril to send emails that don't have PHI
• Client Services currently has this workflow (we need to support this at minimum):

  • CS can type encrypt into subject line to send a secure email with attachments
  • customers get an email, click a link, authenticate and can view their report
  • Reports disappear after x amount of time

Open Questions:

• How can we auth with existing secure email (SAMEL 2.0?)
• How do we send data from our systems via secure email

Estimated Epic Size: ~34

Stories (from Agile Planning high-level grooming session):
1. SPIKE - How do we want to send secure emails? (5)
2. Backend service to talk to 3rd party provider to create account (5)
3. Store account information (5)
4. Design - the UI element for getting to secure email (5)
5. App Dashboard – add element for getting to secure email inbox (3)
6. API for other applications to access secure email (8)c

Acceptance Criteria:

Issues for CRMFB-58:

ID	Issue Type	Summary	Status
CRMFB-143	Story	MOCK: Handle QTest Ids by Environment on Trace Matrix	To Do

- -

⚡ CRMFB-58 > 🔖 CRMFB-143

Story: CRMFB-143

Summary: MOCK: Handle QTest Ids by Environment on Trace Matrix

Description: As a tester, I want to ensure that a qTest test run executed on an environment that isn't STG, will not show up in the report view or trace matrix view in Artifact Generator.

Acceptance Criteria:

Epic: CRMFB-99

Summary: MOCK: QTest Attachments

Description: QTest Attachments

Acceptance Criteria:

Issues for CRMFB-99:

ID	Issue Type	Summary	Status
CRMFB-153	Story	Mock: Story linked to 2 test cases	Requested
CRMFB-151	Story	Mock: Story 3 to a single test case	Requested
CRMFB-150	Story	Mock: Story 2 to a single test case	Requested
CRMFB-149	Story	Mock: Story 1 to a single test case	Requested
CRMFB-147	Bug	MOCK: Completed bug	Done
CRMFB-144	Bug	MOCK: Bug in Report	To Do
CRMFB-102	Story	MOCK: QTest Multiple Attachments	In Progress
CRMFB-101	Story	MOCK: QTest Test Step Attachments	In Progress
CRMFB-100	Story	MOCK: QTest Case Attachments	In Progress

⟳ CRMFB-99 > ⊞ CRMFB-153

Story: CRMFB-153

Summary: Mock: Story linked to 2 test cases

Description:

Acceptance Criteria:

⟳ CRMFB-99 > ⊞ CRMFB-151

Story: CRMFB-151

Summary: Mock: Story 3 to a single test case

Description:

Acceptance Criteria:

⟳ CRMFB-99 > ⊞ CRMFB-150

Story: CRMFB-150

Summary: Mock: Story 2 to a single test case

Description:

Acceptance Criteria:

⚡ CRMFB-99 > ⬚ CRMFB-149

Story: CRMFB-149

Summary: Mock: Story 1 to a single test case

Description:

Acceptance Criteria:

⚡ CRMFB-99 > ○ CRMFB-147

Bug: CRMFB-147

Summary: MOCK: Completed bug

Affects Version(s):

Environment:

Steps to Reproduce:

Actual Behavior:

Expected Behavior:

⚡ CRMFB-99 > ○ CRMFB-144

Bug: CRMFB-144

Summary: MOCK: Bug in Report

Affects Version(s):

Environment:

Steps to Reproduce:

Actual Behavior:

Expected Behavior:

⚡ CRMFB-99 > ⬚ CRMFB-102

Story: CRMFB-102

Summary: MOCK: QTest Multiple Attachments

Description: Show that Artifact Generator can handle many attachments coming through from a QTest Execution

Acceptance Criteria: 1. Demonstrate that AG can handle many attachments coming through from a QTest Execution

⚡ CRMFB-99 > ⬚ CRMFB-101

Story: CRMFB-101

Summary: MOCK: QTest Test Step Attachments

Description: Attachments added to QTest Test Steps should be picked up by Artifact Generator

Acceptance Criteria: 1. Demonstrate that attachments added to a QTest Test Step are brought through by Artifact Generator

⚡ CRMFB-99 > ⬚ CRMFB-100

Story: CRMFB-100

Summary:	MOCK: QTest Case Attachments
Description:	Attachments added to a QTest Test case should be picked up by Artifact Generator
Acceptance Criteria:	1. Demonstrate that files attached to a QTest Test Case are picked up by Artifact Generator

- -

Epic: CRMFB-103

Summary:	MOCK: Text Formatting
Description:	Text formatting

Bold

Italics

Bold and italics

Acceptance Criteria:

Issues for CRMFB-103:

ID	Issue Type	Summary	Status
CRMFB-106	Story	MOCK: Test font formatting	Done
CRMFB-105	Story	MOCK: Test Long URLs within text boxes or modal windows	Done
CRMFB-104	Story	MOCK: Fields using markup numbering from Jira	Done

- -

⚡ CRMFB-103 > 🔖 CRMFB-106

Story: CRMFB-106

Summary:	MOCK: Test font formatting
Description:	This is an example of text in bullets:

- Normal text, **Bold text**.
- *Italic text*. **Bold text**.
- Normal. **Bold text**. Normal.
- ***Bold and Italic text***. Normal text.
- Normal text. ***Italic and Bold text***.

This is a line using **bold text**, *italic text* and ***bold italic*** text.

Acceptance Criteria:	Demonstrate that when a user generates a PDF report via AG, the Jira issue details are formatted to be displayed in the same style as shown on Jira. Artifact Generator should be able to handle the Jira markdown syntax.

Heading 1	Heading 2
Col A1	Col A2

- -

⚡ CRMFB-103 > 🔖 CRMFB-105

Story: CRMFB-105

Summary:	MOCK: Test Long URLs within text boxes or modal windows
Description:	Example with a long URL:

https://foundationmedicine.service-now.com/nav_to.do?uri=%2Fincident.do%3Fsys_id%3D2e41d87ddb74ef0429427b5f2996192a%26sysparm_record_target%3Dincident%26sysparm_record_row%3D1%26sysparm_record_rows%3D4%26sysparm_record_list%3Dcaller_id%253Djavascript%253Ags.getUserID%2528%2529%255Eactive%253Dtrue%255Esubcategory%2521%253DIT%2BSecurity%255EORsubcategory%253DNULL%255EORDERBYDESCnumber

Acceptance Criteria: Demonstrate that text on screen that contains long URLs do not overflow outside text areas or modals. All text should be wrapped within text areas and no text should go off screen.

Demonstrate that the same format is present for the print preview on the browser.

⚡ CRMFB-103 > 🔖 CRMFB-104

Story: CRMFB-104

Summary: MOCK: Fields using markup numbering from Jira

Description: Example of numbering with Jira Markup:

1. First numbered row
2. Second numbered row
3. Third numbered row
4. Fourth numbered row

Acceptance Criteria:
1. Demonstrate that AG can handle Jira Markup
2. Demonstrate that Jira Markup is replaced ('#' with numbers)
3. Demonstrate that numbering is displayed correctly

Epic: CRMFB-107

Summary: MOCK: qTest Formatting

Description: qTest formatting.

Acceptance Criteria:

Issues for CRMFB-107:

ID	Issue Type	Summary	Status
CRMFB-154	Story	Mock: Prerequisites formatting	Requested
CRMFB-152	Story	Mock: Story 4 to a a single test case	Requested
CRMFB-141	Story	MOCK: qTest actual result field	Done
CRMFB-140	Story	MOCK: qTest special characters validation	To Do
CRMFB-139	Story	MOCK: XML code in qTest	Done
CRMFB-108	Story	MOCK: Test line breaks on test steps	Done

⚡ CRMFB-107 > 🔖 CRMFB-154

Story: CRMFB-154

Summary: Mock: Prerequisites formatting

Description:

Acceptance Criteria:

⚡ CRMFB-107 > 🔖 CRMFB-152

Story: CRMFB-152

Summary: Mock: Story 4 to a a single test case

Description:

Acceptance Criteria:

CRMFB-107 > CRMFB-141

Story: CRMFB-141

Summary:	MOCK: qTest actual result field
Description:	Verify qTest actual result field is displayed on Artifact Generator.
Acceptance Criteria:	Demonstrate "Actual Result" field from qTest is populated in executions

---

CRMFB-107 > CRMFB-140

Story: CRMFB-140

Summary:	MOCK: qTest special characters validation
Description:	qTest special characters validation.
Acceptance Criteria:	Demonstrate that Artifact Generator can handle special characters when generating reports

---

CRMFB-107 > CRMFB-139

Story: CRMFB-139

Summary:	MOCK: XML code in qTest
Description:	XML code in qTest is formatted in AG reports.
Acceptance Criteria:	Demonstrate that a generated PDF in Artifact Generator displays xml code block content in any of the columns related to qTest case or test run results.

CRMFB-107 > CRMFB-108

Story: CRMFB-108

Summary:	MOCK: Test line breaks on test steps
Description:	Test line breaks on test steps
Acceptance Criteria:	1. Demonstrate that any QTest test case table, in an Artifact Generator printed report, allow for line breaks where needs be. This is to allow the text to be readable and in the same format as shown in QTest

---

Epic: CRMFB-109

Summary:	MOCK: Miscellaneous test data
Description:	Miscellaneous test data.
Acceptance Criteria:	

Issues for CRMFB-109:

ID	Issue Type	Summary	Status
CRMFB-110	Story	MOCK: Test issue with several test cases	Done

---

CRMFB-109 > CRMFB-110

Story: CRMFB-110

Summary:	MOCK: Test issue with several test cases
Description:	Example of test issue with several test cases associated to it.
Acceptance Criteria:	1. Demonstrate that when a user views the traceability matrix for a large project with many tests assigned to one jira issue, the user is able to view all test cases linked to the issue.

Epic: CRMFB-137

Summary:	MOCK: Specimen Procurement
Description:	As CS Specimen Procurement Specialist, I need to procure a specimen whenever an order is Pre-Accessioned, so that the specimen is delivered to FMI as soon as possible to ensure the quickest TAT possible for the patient.
Acceptance Criteria:	

Issues for CRMFB-137:

ID	Issue Type	Summary	Status
CRMFB-120	Story	MOCK: Build out UI components for the Sample information	Done
CRMFB-119	Story	MOCK: Ability to capture Copied Parties via the Order	Done
CRMFB-118	Story	MOCK: Ability to add a support group to an Order	Done
CRMFB-117	Story	MOCK: Ability to view current "Hound" Sample Tracking of any TRF/CRF	Done
CRMFB-116	Story	MOCK: Prepare LIMS patient data for migration into CRM	Done
CRMFB-115	Story	MOCK: Ability to Procure a New Specimen via Existing Order	Done
CRMFB-114	Story	MOCK: Patient Entity Refinement from Sprint #1	Done
CRMFB-113	Story	MOCK: Indicate Third Party Agreement	Done
CRMFB-112	Story	MOCK: Create Auto-Number for New Patient Records	Done
CRMFB-111	Story	MOCK: Duplicate data detection for Patient entity	Done

⟳ CRMFB-137 > ▯ CRMFB-120

Story: CRMFB-120

Summary:	MOCK: Build out UI components for the Sample information
Description:	For now, build the placeholders on the Order that are necessary to house all of the fields and populate with test data, as there is no API with LIMS yet.
Acceptance Criteria:	1. Verify that the UI components for the Sample are built out on the Order. (see specification)

⟳ CRMFB-137 > ▯ CRMFB-119

Story: CRMFB-119

Summary:	MOCK: Ability to capture Copied Parties via the Order

Description:	Copied parties will eventually need access to automated reporting.Kimberly Snader will complete manual testing. Automated test to be deferred to subsequent sprint.
Acceptance Criteria:	1. Verify that an existing Contact can be selected as a Copied party.1.1.Only allow Contacts which are designated as an ordering authority and have a reporting profile to be selected.2. Verify that non-contacts can also be captured as a copied parties.2.1. Verify that the following pieces of data can be captured for non-contacts: Name; Fax#, Email, Facility3. Verify that copied parties can be entered at the time of pre-accessioning as well, after the initial save of the Order.

⟁ CRMFB-137 > ⊓ CRMFB-118

Story: CRMFB-118

Summary:	MOCK: Ability to add a support group to an Order
Description:	
Acceptance Criteria:	1. Verify that an existing Account can be selected as a Support Group.

⟁ CRMFB-137 > ⊓ CRMFB-117

Story: CRMFB-117

Summary:	MOCK: Ability to view current "Hound" Sample Tracking of any TRF/CRF
Description:	http://appuat014:8081/tracking/houndCT.htm?&AF=1&trfL=CRF002916,CRF003452,CRF003463
Acceptance Criteria:	1. Verify that the current Sample Tracking status of any TRF/CRF in CRM can be viewed via a "link" to Hound.1.1. Provide a link at the Sample entity.1.2. Make the link configurable per environment via the Configuration entity.

⟁ CRMFB-137 > ⊓ CRMFB-116

Story: CRMFB-116

Summary:	MOCK: Prepare LIMS patient data for migration into CRM
Description:	• address info: make sure fields contain valid info- update missing data with scripts, where possible- if unable to update data, pull aside and send to client services for manual updates
Acceptance Criteria:	1. Verify that Patient data in LIMS is "clean" and ready for import into CRM.1.1. Identify Patient data which cannot be "cleaned" and pass along to Client Services for manual updates.

⟁ CRMFB-137 > ⊓ CRMFB-115

Story: CRMFB-115

Summary:	MOCK: Ability to Procure a New Specimen via Existing Order
Description:	n number of Block Procurement requests can be open at any given point in time._
Acceptance Criteria:	1. Verify that a new Block Procurement request can be created via an existing Order (not a pre-order).

⟁ CRMFB-137 > ⊓ CRMFB-114

Story: CRMFB-114

Summary:	MOCK: Patient Entity Refinement from Sprint #1
Description:	
Acceptance Criteria:	• Add field validation for Patient Entity - Gender field drop down selection of "Other" modified to "Unknown" - Add Patient Transplant Status field to UI - Add the LIMS Patient ID field to UI - Add External ID field to UI - Move the Patient UI header in the SiteMap to somewhere easier to locate. - Validate all fields are captured in the patient entity.

⚡ CRMFB-137 > ⬚ CRMFB-113

Story: CRMFB-113

Summary:	MOCK: Indicate Third Party Agreement
Description:	This story is purely an indication of the presence of the form. There will be no rules to validate the attachment/availability of the document.
Acceptance Criteria:	1. Verify the "Third Party Agreement" field is displayed on the Patient entity (refer to field specification).

⚡ CRMFB-137 > ⬚ CRMFB-112

Story: CRMFB-112

Summary:	MOCK: Create Auto-Number for New Patient Records
Description:	
Acceptance Criteria:	1. Verify that a unique ID is generated for a new Patient record.

⚡ CRMFB-137 > ⬚ CRMFB-111

Story: CRMFB-111

Summary:	MOCK: Duplicate data detection for Patient entity
Description:	Note: This is NOT case sensitive
Acceptance Criteria:	1. Verify that if a Patient Record is saved which contains the same First Name, Last Name, and DOB of another Patient record already in the system, an alert is displayed to the user.1.1. Verify the user can override the alert and choose to save a new Patient record.1.2. Verify the user can choose to discard the current entry and use the existing record.2. Verify that Patient ID is included in separate duplicate detection rule.

Epic: CRMFB-138

Summary:	MOCK: House Account Management
Description:	Data Steward currently manages House Accounts from MOTHER. Workflow will need to be migrated to CRM as a result of pre-a/acc go-live.
Acceptance Criteria:	

Issues for CRMFB-138:

ID	Issue Type	Summary	Status
CRMFB-136	Story	MOCK: Create a Patient via detailed entry	To Do
CRMFB-135	Story	MOCK: Quick-create a Patient	To Do
CRMFB-134	Story	MOCK: Search for a Patient	To Do
CRMFB-133	Story	MOCK: Update an existing Patient record	To Do
CRMFB-132	Story	MOCK: Set Patient entity status	To Do
CRMFB-131	Story	MOCK: Ability to attach a document to CRM entity	To Do
CRMFB-130	Story	MOCK: Ability to capture required metadata with attached documents	To Do

CRMFB-129	Story	MOCK: Ability to select document Type when attaching it to Patient record	To Do
CRMFB-128	Story	MOCK: Add new version to an attached document on CRM entity	To Do
CRMFB-127	Story	MOCK: Create Business Units for Security	To Do
CRMFB-126	Story	MOCK: Configure default views for Patient Entity	To Do
CRMFB-125	Story	MOCK: Ability to view existing attachements	To Do
CRMFB-124	Story	MOCK: Ability to share attached document across multiple entities	To Do
CRMFB-123	Story	MOCK: Create Order Entity in CRM v1	To Do
CRMFB-122	Story	MOCK: Ability to update an existing Order record	To Do
CRMFB-121	Story	MOCK: Ability to Search for a Order in the Global search	To Do

---

CRMFB-138 > CRMFB-136

Story: CRMFB-136

Summary:	MOCK: Create a Patient via detailed entry
Description:	New records will start with an auto number higher than current system, due to 3.4 - Migrate Patient Data (max value + n)This does not include data migration, nor integrations.
Acceptance Criteria:	1. Verify that a Patient record can be created and successfully saved via the detailed entry screen.2. Verify the necessary fields for a Patient record are available via the detailed entry screen. (field specifcations)3. Verify the required fields are filled out.4. Verify any field validations are applied.

CRMFB-138 > CRMFB-135

Story: CRMFB-135

Summary:	MOCK: Quick-create a Patient
Description:	New records will start with an auto number higher than current system, due to 3.4 - Migrate Patient Data (max value + n)
Acceptance Criteria:	1. Verify that a Patient record can be created and successfully saved via the quick-create entry screen.2. Verify the necessary fields for a Patient record are available via the quick-create screen. (Field Specifications)3. Verify the required fields are filled out.4. Verify any field validations are applied.

CRMFB-138 > CRMFB-134

Story: CRMFB-134

Summary:	MOCK: Search for a Patient
Description:	+Search Fields:+First NameLast NameFull NameDOBView Columns:LIMS Patient IDFirst NameMiddle NameLast NameGenderDate of BirthCityState/ProvenceZipCountryFilter Criteria:All Patients (No filtering)
Acceptance Criteria:	1. Verify that a Patient entity can be searched from the patient entity view using the defined fields. (listed in the description)2. Verify that the Patient entity can be searched from the global search option.

CRMFB-138 > CRMFB-133

Story: CRMFB-133

Summary:	MOCK: Update an existing Patient record
Description:	
Acceptance Criteria:	1. Verify that an existing Patient record can be edited and saved successfully based on security roles.1.1. Verify that required fields validations are enforced upon save.

♻ CRMFB-138 > ⊡ CRMFB-132

Story: CRMFB-132

Summary:	MOCK: Set Patient entity status
Description:	• Jim wants clarification on Country Code and Primary Phone Number, as these are fields that are typically NOT provided to Client Services. Currently, Client Services is NOT responsible for gathering this data if it is missing.- Client Services IS currently responsible for gathering First Name, Last Name, DOB, and Gender.
Acceptance Criteria:	1. Verify that if data is missing from 'Business Required' fields and the Patient record is saved, the status reason of the Patient Entity is set to "In Progress".2. Verify that if data is entered in all of the 'Business Required' fields and the Patient record is saved, the status of the Patient Entity is set to "Active".

♻ CRMFB-138 > ⊡ CRMFB-131

Story: CRMFB-131

Summary:	MOCK: Ability to attach a document to CRM entity
Description:	
Acceptance Criteria:	1. Verify that a document can be attached and successfully saved to a CRM entity.1.1 Browse for document and attach

---

♻ CRMFB-138 > ⊡ CRMFB-130

Story: CRMFB-130

Summary:	MOCK: Ability to capture required metadata with attached documents
Description:	
Acceptance Criteria:	

♻ CRMFB-138 > ⊡ CRMFB-129

Story: CRMFB-129

Summary:	MOCK: Ability to select document Type when attaching it to Patient record
Description:	
Acceptance Criteria:	1. Verify that when attaching a document to a CRM entity, the Type of document must be selected from a list of values. (see specification)

♻ CRMFB-138 > ⊡ CRMFB-128

Story: CRMFB-128

Summary:	MOCK: Add new version to an attached document on CRM entity
Description:	
Acceptance Criteria:	1. Verify that a document can be attached as a NEW version of an existing attachment.1.1. Verify that version history is maintained.1.2. Verify that an audit trail is kept of version history.

⚡ CRMFB-138 > 🔖 CRMFB-127

Story: CRMFB-127

Summary:	MOCK: Create Business Units for Security
Description:	This work significantly impacts the Pharma team, as they are moving forward with their work. Not having the Business Units and Security roles configured properly, opens up the possibility that Pharma users will have unauthorized access to PHI data.
Acceptance Criteria:	1. Create the required business units to support our current user base.1.1 Create administrative security roles and define list of users (root level).1.2 Create client services business unit and roles.1.3. Ensure that units and roles are scaleable for future growth.2. Migrate existing users to the newly created business units.3. Document business unit/security role configuration on Confluence.

⚡ CRMFB-138 > 🔖 CRMFB-126

Story: CRMFB-126

Summary:	MOCK: Configure default views for Patient Entity
Description:	Default View : "All Patients" includes Active Patients AND Inactive PatientsFields on the view:External LIMS IDFirst NameMiddle NameLast NameGenderDate of BirthCityState/ProvenceZipCountry
Acceptance Criteria:	1. Define filter parameters and name of list - Active Patients2. Define filter parameters and name of list - Inactive Patients3. Define data elements (columns)4. Verify that the views are available only to the authorized users.

⚡ CRMFB-138 > 🔖 CRMFB-125

Story: CRMFB-125

Summary:	MOCK: Ability to view existing attachements
Description:	
Acceptance Criteria:	1. Verify that attached documents can be opened and viewed from within a CRM entity.1.1. Verify that previous versions of attached documents can be opened and viewed.

⚡ CRMFB-138 > 🔖 CRMFB-124

Story: CRMFB-124

Summary:	MOCK: Ability to share attached document across multiple entities
Description:	
Acceptance Criteria:	1. Verify that an attached document can be shared across multiple entities.

⚡ CRMFB-138 > 🔖 CRMFB-123

Story: CRMFB-123

Summary:	MOCK: Create Order Entity in CRM v1
Description:	
Acceptance Criteria:	1. Verify that the Case or Order entity has been enabled with the necessary fields. (see field specifications in linked page)2. Verify the required fields are filled out.

⚡ CRMFB-138 > 🔖 CRMFB-122

Story: CRMFB-122

Summary: MOCK: Ability to update an existing Order record

Description:

Acceptance Criteria: 1. Verify that an existing Order record can be edited and saved successfully based on security roles. (Client Services only)1.1. Verify that required fields validations are enforced upon save.

- -

⬦ CRMFB-138 > ⬚ CRMFB-121

Story: CRMFB-121

Summary: MOCK: Ability to Search for a Order in the Global search

Description:

Acceptance Criteria: 1. Verify that active Order records can be searched globally by: *Patient Name (first and/or last)* Order Number *Ordering Account* Ordering Clinician

- -

Test Cases and Evidence

Please see next page for document.

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-32	TR-83					Unexecuted

Test Case: TC-32

Summary: Step Line Breaks - v1.0

Step	Description	Expected Result	Actual Result	Status
1	This is an example of test step with line breaks. Line after a line break. Another line after a line break.	First line. Second line. Third line.		Unexecuted

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-33	TR-84					Unexecuted

Test Case: TC-33

Summary: Test to Keep Unexecuted - v1.0

Step	Description	Expected Result	Actual Result	Status
1	First step	First expected result		Unexecuted
2	Second step	Second expected result		Unexecuted

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-34	TR-85	Manual	01-25-2019 10:38:24	01-25-2019 10:39:07	Nathan McAlinden	Passed

Test Case: TC-34

Test Type: Manual

Summary: Test to Execute against STG environment - v1.0

Step	Description	Expected Result	Actual Result	Status
1	First step	First expected result		Passed
2	Second step	Second expected result	Testing Actual Result: <TEST> <create-by-workflow> <workflowname>Pathology</workflow-name> </create-by-workflow> </TEST>	Passed

© 2020 FOUNDATION MEDICINE, INC. V.3.1.0

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-35	TR-86	Manual	12-13-2018 00:34:54	12-13-2018 00:35:02	Ulises Castaneda	Passed

Test Case: TC-35

Test Type: Manual

Summary: Test to Execute against non STG environment - v1.0

Step	Description	Expected Result	Actual Result	Status
1	First step	First expected result		Passed
2	Second step	Second expected result		Passed

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-36	TR-87					Unexecuted

Test Case: TC-36

Summary: XML/JSON code in steps - v2.0

Step	Description	Expected Result	Actual Result	Status

1	This is an example of xml code:
	
	`<create-by-workflow>`
	`<workflow-name>H and E Slide_V2_DR</workflow-name>`
	`</create-by-workflow>`
	`<EXTERNAL_REFERENCE>TRF000151.01_HE</EXTERNAL_REFERENCE>`
	`<create-by-workflow>`
	`<workflowname>IHC</workflow-name>`
	`</create-by-workflow>`
	`<TEST>`
	`<create-by-workflow>`
	`<workflowname>Pathology</workflow-name>`
	`</create-by-workflow>`
	`</TEST>`
		XML code has to be properly formatted.		Unexecuted

2	This is an example of JSON code:
	
	`{`
	`"execution": "PRE",`
	`"status": [`
	`"progress", "accepted"`
	`]`
	`}`
		JSON code has to be properly formatted.		Unexecuted

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-37	TR-88	Manual	01-16-2020 10:56:16	01-16-2020 10:56:25	Nathan McAlinden	Passed

Test Case: TC-37

Test Type: Manual

Summary: Step formatting and styling - v3.0

Step	Description	Expected Result	Actual Result	Status
1	This is a test step with format: • Normal text, **bold text**. • *Italic text*. **Bold text**. • **Bold text**, normal text.	Formatting and styling is correct.		Passed

Test Case	Test Run	Test Type	Executed Start	Executed End	Tester	Status
TC-38	TR-89	Manual	01-31-2020 11:11:56	01-31-2020 11:12:11	Michael Mackin	Passed

Test Case: TC-38

Test Type: Manual

Summary: Actual Result values - v1.0

Step	Description	Expected Result	Actual Result	Status
1	Verify this step contains an "Actual Result" value.	"Actual Result" has a value in execution.		Passed

TC-38 > TS-1 > EvidenceInInlineReportAtt2.PNG

REPORT   TRACEABILITY   GENERATE REPORT

Type	Key	Summary	Status
▸ Epic	CRMFB-44	Mock: Node Upgrade	Epic To Do
▸ Epic	CRMFB-49	MOCK: Critical Automation	Epic In Progress
▸ Epic	CRMFB-52	MOCK: Payer metadata for Billing Team	Epic To Do
▸ Epic	CRMFB-55	MOCK: Move to DevOps pipeline	Epic Requested
▸ Epic	CRMFB-56	MOCK: Account Notification	Epic Requested
▸ Epic	CRMFB-57	MOCK: Formulary Ordering	Epic Requested
▸ Epic	CRMFB-58	MOCK: Secure Email Integration	Epic Requested
▸ Epic	CRMFB-99	MOCK: QTest Attachments	Epic To Do
▸ Epic	CRMFB-103	MOCK: Text Formatting	Epic Requested

TC-38 > TS-1 > EvidenceInInlineReportAtt1.PNG

PROJECT DETAILS

Select Project

CRM FooBar ✕ ⇅

Select Version

1.2.1 ✕ ⇅

TEST DETAILS ☐ Exclude Test Details from report

Select Test Cycle
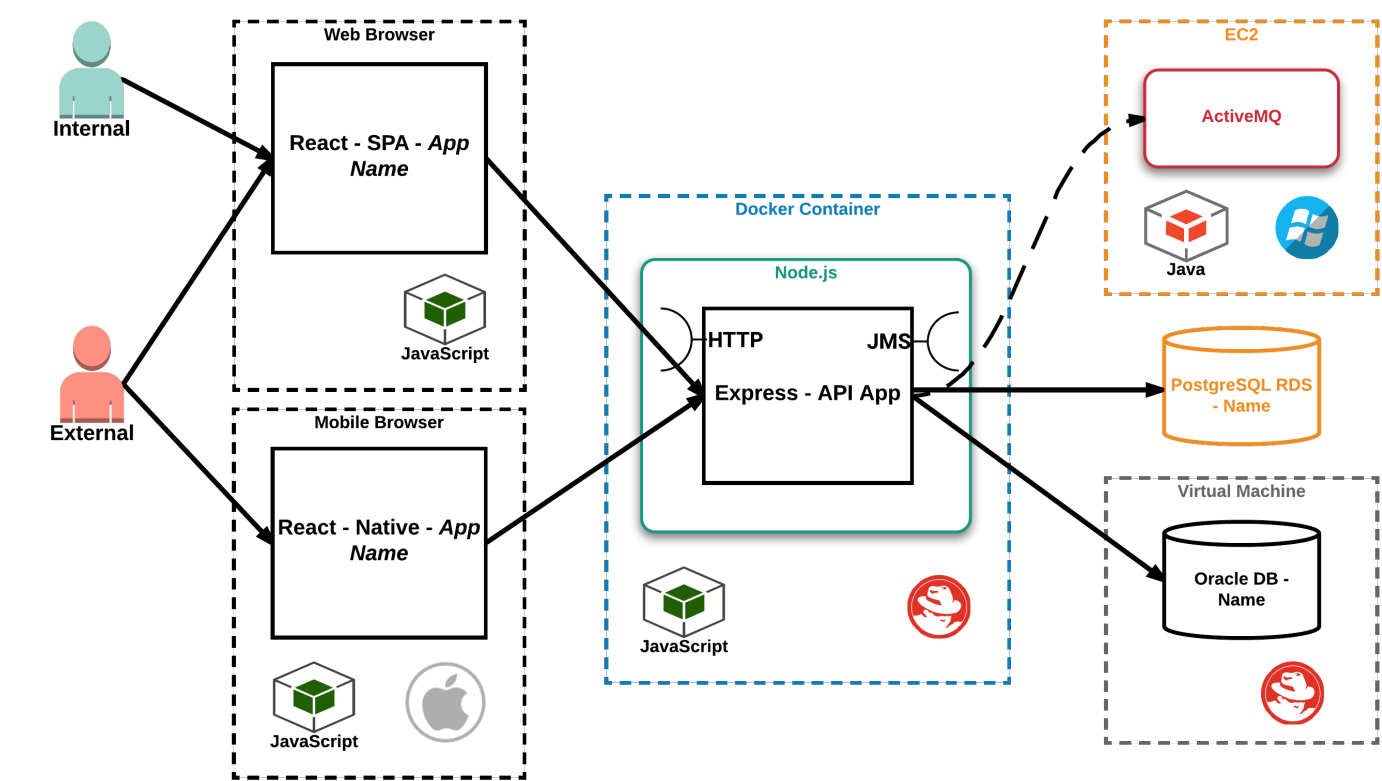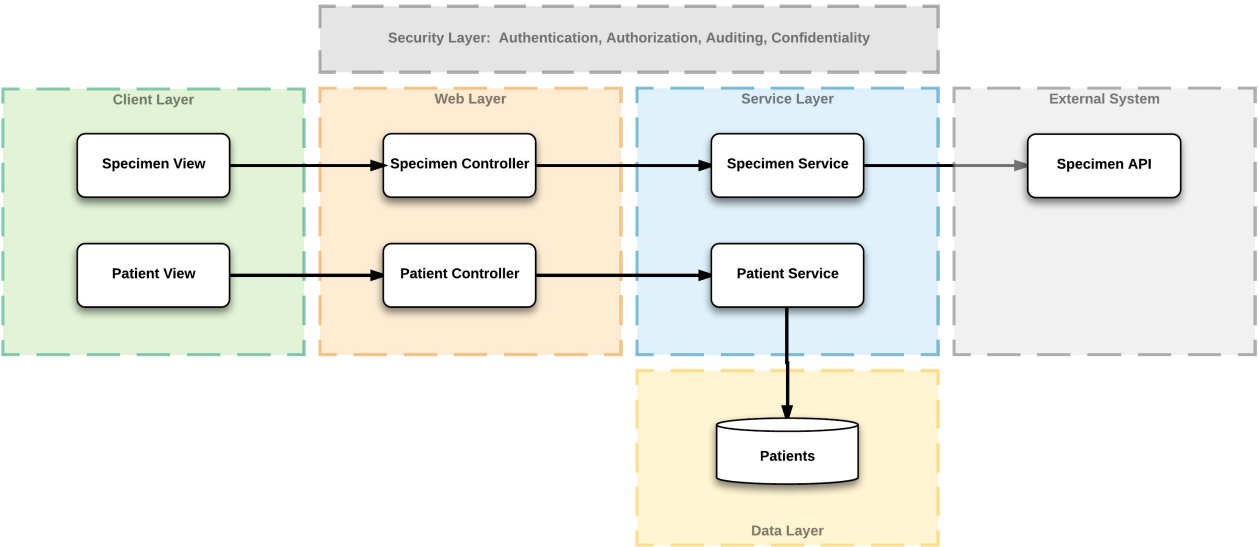
AG General Test Data ✕ ⇅

Execution

| PRE | POST |

RUN REPORT

DOCUMENT ID   PROJECT

FMI-2472352   CRM FooBar

VERSION

1.2.1

TEST CYCLE

AG General Test Data

DATE GENERATED

22 APR 2020 06:52 EDT

Jira Web Link Attachments

Please see next page for document.

DOCUMENT ID   PROJECT

FMI-2472352   CRM FooBar

VERSION

1.2.1

TEST CYCLE

AG General Test Data

DATE GENERATED

22 APR 2020 06:52 EDT

© 2020 FOUNDATION MEDICINE, INC. V.3.1.0

CRMFB-44 > CRMFB-60 > Example Image

CRMFB-137 > CRMFB-120 > JPEG Image

⚡ CRMFB-44 > ⬚ CRMFB-60 > Example PDF

Please see next page for document.

⚡ CRMFB-44 > ⬚ CRMFB-60 > Example PDF

© 2020 FOUNDATION MEDICINE, INC. V.3.1.0

← BLOG

Informing precision treatment approaches for breast cancer with FoundationACT liquid biopsy

Jon Chung, PhD
Senior Clinical Development Manager at Foundation Medicine

f   🐦   G+   in

OCTOBER 4, 2017
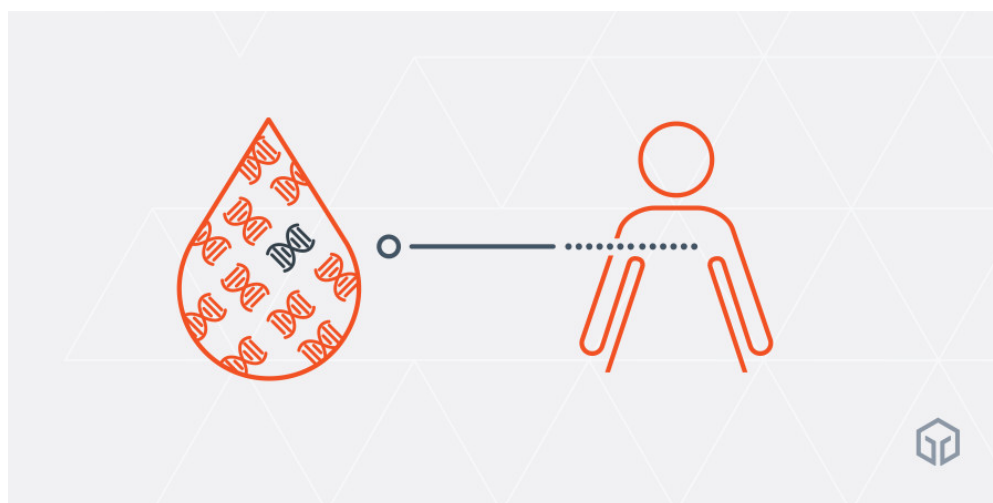
Breast cancer is an important example of how a better understanding of cancer biology can lead to more personalized and precise treatments that improve patient care and prognosis. Surgery, hormone therapy, and chemotherapy were the standard of care for decades, but research into the *HER2* gene as a potential biomarker for certain types of breast cancer ultimately led to the first FDA-approved molecularly targeted therapy for breast cancer. Since then, we have continued to learn more about the various molecular subtypes of this disease. Finding new biomarkers for breast cancer could aid in the development of more therapies precisely targeted to those genomic alterations. The ultimate goal for all of us in oncology is to have personalized treatment options available for every person diagnosed with breast cancer.

The evolution of each breast cancer

Comprehensive genomic profiling (CGP) is one path toward achieving this vision of more personalized treatments. CGP has not only provided a better understanding of the genomic markers of breast cancer, but is helping to drive new research directions and new types of treatments.

This enhanced look into a tumor's molecular underpinning has shown us that breast cancer – like other cancers – can be highly dynamic and evolve over time in response to therapy in each individual. Unfortunately, for some patients with advanced or refractory disease, obtaining a new sample may not be medically feasible.

Liquid biopsy has emerged as a non-invasive genomic profiling option that assesses circulating tumor DNA (ctDNA). FoundationACT®, our ctDNA assay, requires only a blood sample to probe for genomic markers that could help guide treatment. Importantly, this assay allows us to measure these markers both when a metastatic cancer arises and at different stages of its evolution.

Informing breast cancer treatment with liquid biopsy

We recently published new findings that support the value of this technique. In the study, Foundation Medicine and our collaborators across the country showed that FoundationACT was able to identify genomic alterations in the majority of a group of patients with estrogen receptor-positive (ER+) metastatic breast cancer. These potentially actionable alterations could inform treatment decisions as well as complement tissue-based testing.

Study results showed that 84% of ER+ breast cancer patients had evidence of ctDNA in the blood, and at least one genomic alteration was found in 78% of cases. In addition to *ESR1* mutations, which are associated with resistance to aromatase inhibitor therapy, we identified genomic alterations that have been associated with response to a targeted therapy in metastatic breast cancer and may represent a new direction of treatment for patients refractory to standard therapies.

Furthermore, a subset of patients had matched tissue and ctDNA samples from approximately the same time period, which allowed for concordance analyses. This comparison showed that 89% of mutations detected in tissue were also detected in ctDNA, demonstrating that FoundationACT liquid biopsy could provide a complementary approach to tissue-based genomic testing.

More options for more patients

FoundationACT has already been shown to identify potentially actionable alterations in other types of cancer, including lung, gastrointestinal and prostate cancer. These new findings support the ability of FoundationACT to capture genomic alterations that may drive recurrence of disease or treatment resistance in ER+ breast cancer, helping to inform next steps for treatment, even for patients who may not be able to undergo a tissue biopsy. For patients who need a non-invasive genomic profiling option, FoundationACT could provide important information to navigate the complex dynamics of cancer.

See the full publication here: Chung et al. Hybrid capture-based genomic profiling of circulating tumor DNA from patients with estrogen receptor-positive metastatic breast cancer. *Annals of Oncology*. 2017.

Genomic Testing

Insights & Trials

Partnerships

About Us

Blog

Press & Investors

Careers

Contact Us

GENOMIC TESTING

Testing Overview

Financial Services & Support

Order a Test

INSIGHTS & TRIALS

Overview

FoundationSmartTrials™

FoundationInsights™

PARTNERSHIPS

Overview

Institutional Partnerships

Biopharma Partnerships

10/20/2017   Informing precision treatment approaches for breast cancer with FoundationACT liquid biopsy - Foundation Medicine

Page 44 of 44   5/5