

REQUIREMENTS DOCUMENT

SDM WebApp Project | Team Okay Awesome

Phuoc Trinh | Lok San Gordon Chan | Thu Huynh | Sazedra Sultana | Lucy Cao

June 4th, 2020

TABLE OF CONTENTS

REVISION HISTORY	3
Introduction	4
Assumptions	4
Personas/Scenarios	5
Functional Requirements	7
Non-functional Requirements	9
UI/UX	10
Software Design	22

REVISION HISTORY

VERSION	DATE	DESCRIPTION / CHANGES MADE
2.5	06.04.2020	Added activity diagram, all information and diagrams finalized
2.4	05.18.2020	Added two storyboards + class diagram + activity diagram +communication diagram
2.3	05.05.2020	Changes to UI. Added a flow diagram of the Save/Update API Service for the backend under “Additional Diagram” section. Name: Backend: SDM Update/Save Data API
2.2	04.22.2020	Update bulk update design to be placed on the right. Took out one of the requirements: bulk update barcode sharing
2.1	04.10.2020	Add test flow diagram
2.0	03.06.2020	Final version with all sections expanded. Updated Sequence Diagram + Frontend Class Diagram
1.2	02.26.2020	Add Storyboards + Activity diagrams
1.1	02.21.2020	Add UI mock-ups + UML diagrams
1.0	01.24.2020	Initial version

Introduction

1.1 Team Introduction

Our team name is “Okay Awesome”. There are 5 members in our team that are going to work on the project for our sponsor, Paciolan. The team members include Phuoc Trinh, Gordon Chan, Thu Huynh, Sazedra Sultana, and Lucy Cao.

1.2 Project Description

The project is a web-based application that involves UX design and implementation of the design. We are tasked with the job of redesigning a previous model and adding additional features and functionalities. This application uses messaging architecture to manage metadata for events such as enabling near-field communication (NFC), event relations, etc. This project utilizes tools like React, JavaScript, NodeJS, Express, and Kafka. In this project, we will mostly focus on the Front End web application and managing metadata from events of the application. This project will be catering to customers/clients like the Pac8 collegiate teams and others that utilize the main tool. The tool we are developing will mostly focus on allowing our clients to modify or enable a few features for their events, and to display metadata.

Assumptions

In order for the project to move forward as expected, the team is working under the following assumptions:

- We are assuming that the users of the application will already have all the necessary information to access the tool.
- We are assuming that our team members will have at least a few hours per day to work on the project.
- We are assuming we will be able to contact our sponsor about the problems we face and communicate at least on a weekly basis.
- The majority of our users will be accessing the web-app via devices with a screen size over 900 pixels, according to Paciolan’s UX engineer.

Section 1: Requirements

Personas/Scenarios

Persona 1 - Robert:

PRIMARY PERSONA



"I want people to leave with a great experience and come back again."

AGE 35
TICKETING EXP 6 years

Director of Ticket Operations

SIMILAR TITLES & ROLES: Box Office Manager, Director of Event Setup

The Director of Ticket Operations spends a majority of his time setting up, maintaining, and scaling events so they run smoothly. He also frequently runs reports for his office, promoters, and other departments.

He is in charge of the box office's day-to-day operations and making sure everything is prepared for event days, onsales, and normal sales. He oversees the ticket sellers and troubleshoots any issues that they escalate.

GOALS

- Find new ways to improve box office efficiency.
- Train ticket sellers to sell and troubleshoot effectively.
- Successfully resolve customer service issues.

FRUSTRATIONS

- Having his workflow interrupted by questions.
- Ticket delivery, website, or fraud issues.
- Non-streamlined reporting.
- Having to develop workarounds in the ticketing system.
- Last minute configuration changes from upper management or promoters.

A TYPICAL EVENT DAY

- Confirm ticketing system, scanners and barcodes, and ticket scan report are set up for event
- Check seat statuses (opens, holds, etc.)
- ☞ Get call from Promoter for last minute seat holds (ARTS & Annex)
- Print will call tickets
- Print student athlete guest lists (Athletics)
- Walk through venue to check or fix setup
- Assign staff duties and brief Ticket Sellers on seat map availability
- Open windows 2 hours before event
- Supervise ticket sellers
- ☞ Troubleshoot customer barcode issue
- ☞ Troubleshoot ticket scan issue
- ☞ Troubleshoot resold ticket issue
- Prepare final reports for show settlement for senior management

Scenario:

Robert, as the director of ticket operations, wants to change the settings of his events so that they run smoothly.

He wants to enable NFC for all of his events in the spring 2020 season, so he goes to the search page and searches "spring 2020". His results are displayed and he selects all after hitting the update button. He then checks the NFC checkboxes so that they are enabled. He confirms his changes and sees the summary page.

Persona 2 - Sam:



SAM - The General Manager

"I oversee venue operations and make sure events are profitable"

As a leader of the venue, the General Manager works with the box office, marketing, promoters, and event staff to drive revenue generation and run successful events.

Profession	General Manager
Age	46
GM EXP	5 years
Venue Management EXP	10 years

GOALS

- **Ensure his venue is successful.**
- **Book events to maximize venue capacity and revenue.**

FRUSTRATIONS

- **Not being able to access real-time ticket sales data without contacting the ticket office or waiting for emailed reports.**
- **Having to quickly manage promoter expectations and market and artist volatility.**

Scenario:

Since Sam is rarely at his desk since being too busy, he wants easy on-the-go access to the KPIs for the event.

Sam can get information from an easy to use ticketing and reporting software such as SDM app which helps him to get information about events and gather sales performance data, particularly during on sales, to make strategic decisions that can maximize the venue's success.

Functional Requirements

Use Case Diagram

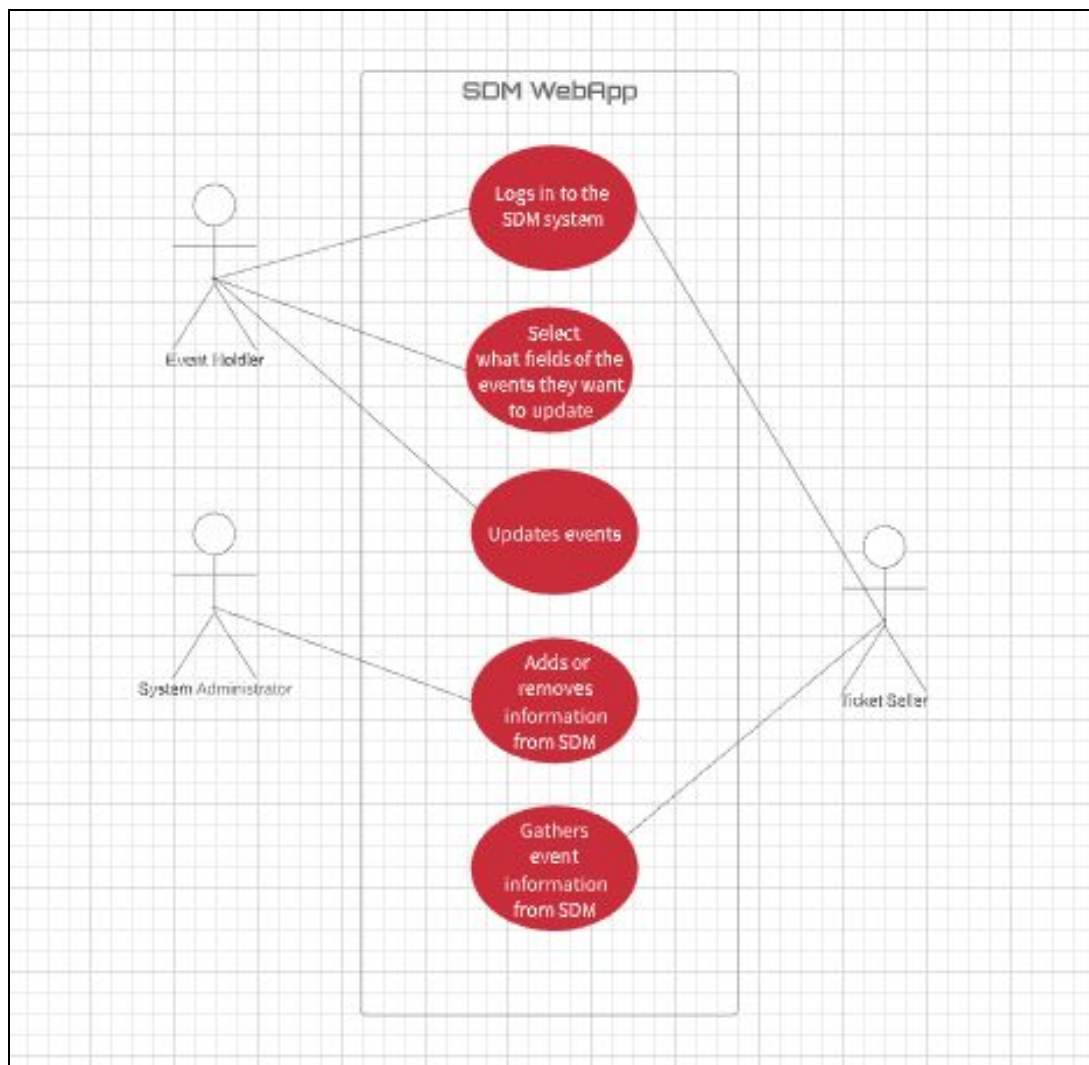


Figure 1.1. Use-Case Diagram for SDM Web App's Corporate Site.

USE CASE 1 Conduct freeform search in the SDM

Priority: **MUST HAVE**

Basic Flow

- 1 The user logs in to the SDM system.
- 2 The user types keywords into the search box.
- 3 The user hits enter.
- 4 The results matching the keywords will be displayed in the search result page.

USE CASE 2 Make a bulk update

Priority: **MUST HAVE**

Basic Flow

- 1 The user logs in to the SDM system.
- 2 The user types keywords into the search box.
- 3 The user hits enter.
- 4 The results matching the keywords will be displayed.
- 5 The user presses the update button.
- 6 Checkboxes will appear next to results.
- 7 The user checks events they want to update.
- 8 The user selects the fields of the events they want to update.
- 9 The user hits confirm.
- 10 The user is presented with a confirmation (summary) page.

USE CASE 3 Save changes to the metadata database

Priority: **MUST HAVE**

Basic Flow

- 1 The user hits confirm to confirm their changes.
- 2 Once the user makes changes to the event(s), the current metadata of those will be retrieved from MTD
- 3 The system compares the old data with the new ones and makes an update.
- 4 The metadata then will be saved to Cassandra.

USE CASE 4 Users should be able to enable integration for each of the events, (SMS, check-in, NFC)

Priority: **MUST HAVE**

Basic Flow

1. The user enters their search
2. Search results come up with the events related
3. Application displays enabled integration for SMS, check-in, NFC on each event
4. Users select a single event
5. User choose to enable or disable their preferred settings on the detail page (which already exists)

Non-functional Requirements

Security

Ensure that the network and connection are secured for the client. Secured databases for events' metadata with secured encryptions.

Reliability

If a user makes changes and updated their events, these changes must be accurately reflected on the database once the website retrieves this information back to the webpage to be displayed.

Usability

Our web-app will become a sub-system within Paciolan's existing software, therefore our web-app shall provide a familiar interface such that users would not be confused by a sudden shift in design language.

Performance

Our web-app has to be able to retrieve and update the bulk of metadata from the database in a timely manner. Since repetitive access to the database would require significant time that could diminish the overall user experience.

Maintainability

Ensure that the search page is more freeform than the existing page. Implement an event-driven architecture.

Correctness

The design should follow the Paciolan Pac8 outline for the UI/UX design. Some design elements (such as drop-down menus) should follow the ADA guidelines.

Section 2: UI Design

Search Page

In the old design, the search function was conducted only using the event code and season code. Following below is the new design that aims to convert it into a freeform search where the user can enter any keyword relevant to the event into the search box, and the system will return them a list of matching events.

Supplemental Data

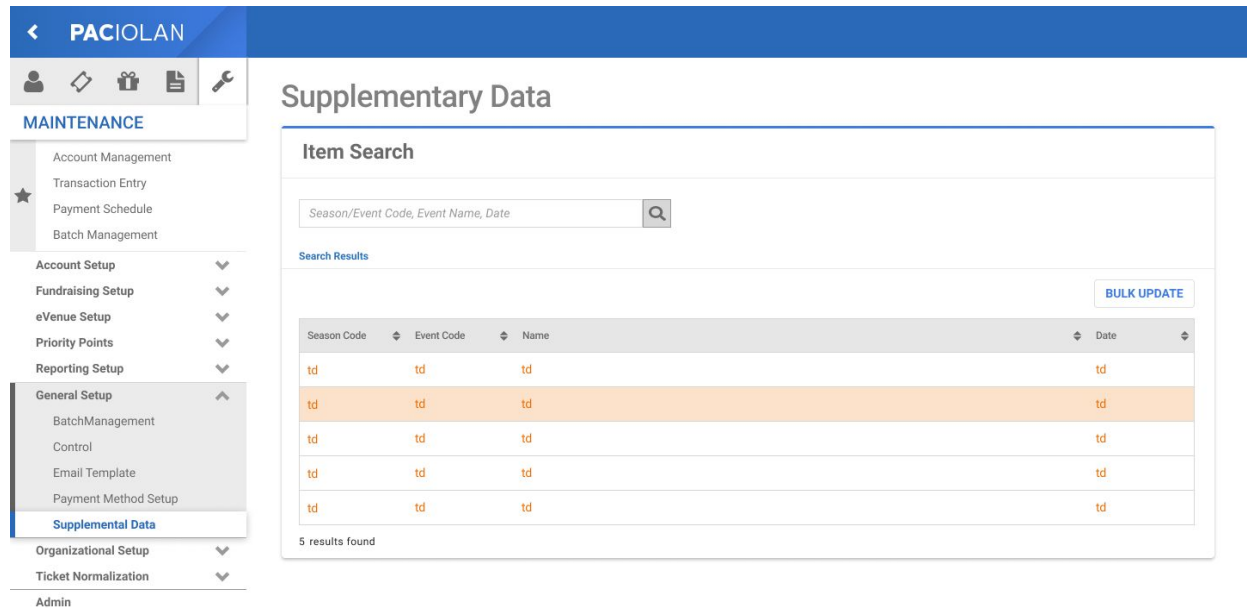
Item Search

Event Code, Event Name, Season Code, Season Name

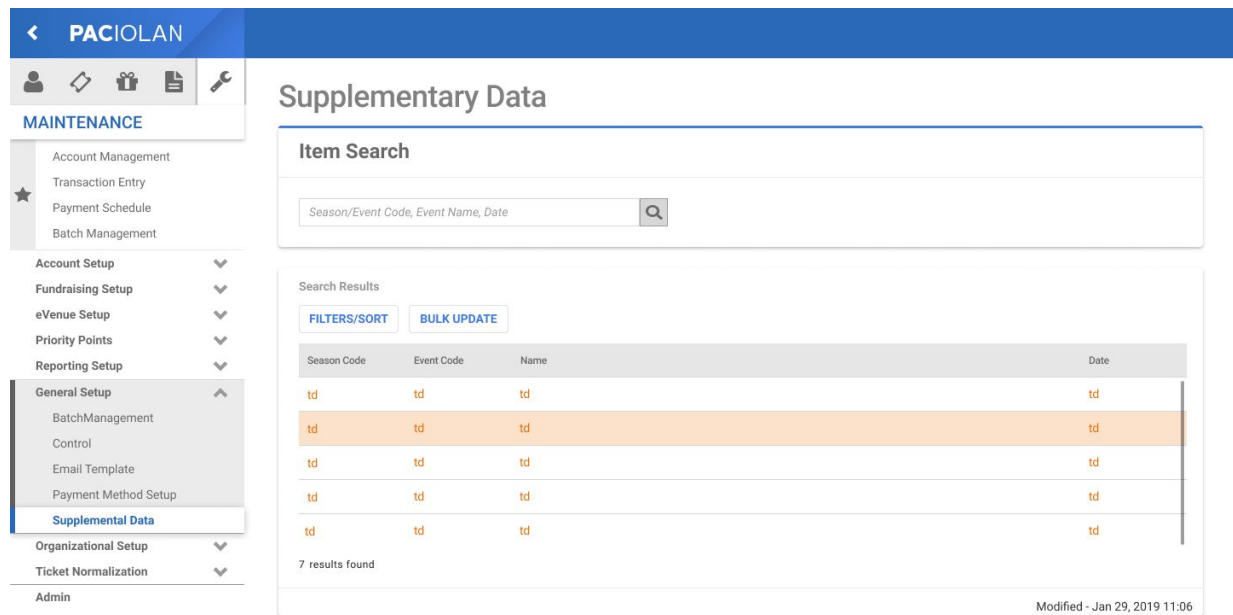
Search Result Page

We have developed two initial designs for the search result page. The 3 key functions that the search result page has to accomplish are to lead the user to an individual item detail page, to give a brief visualize of the state of each item, and to lead the user to the bulk update page.

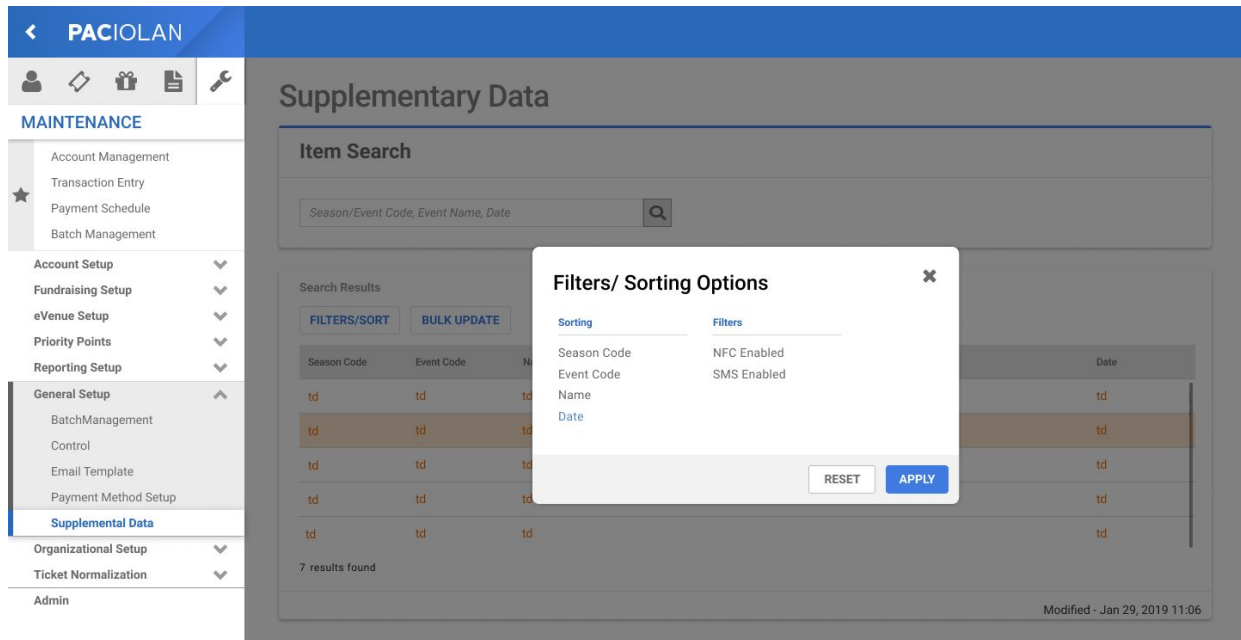
(Search Result Mockup Version 1: Search results will be displayed within the same panel as the search box, which require the panel to change size dynamically.)



(Search Result Mockup Version 2: Displaying search result in a separate self-contained panel.)

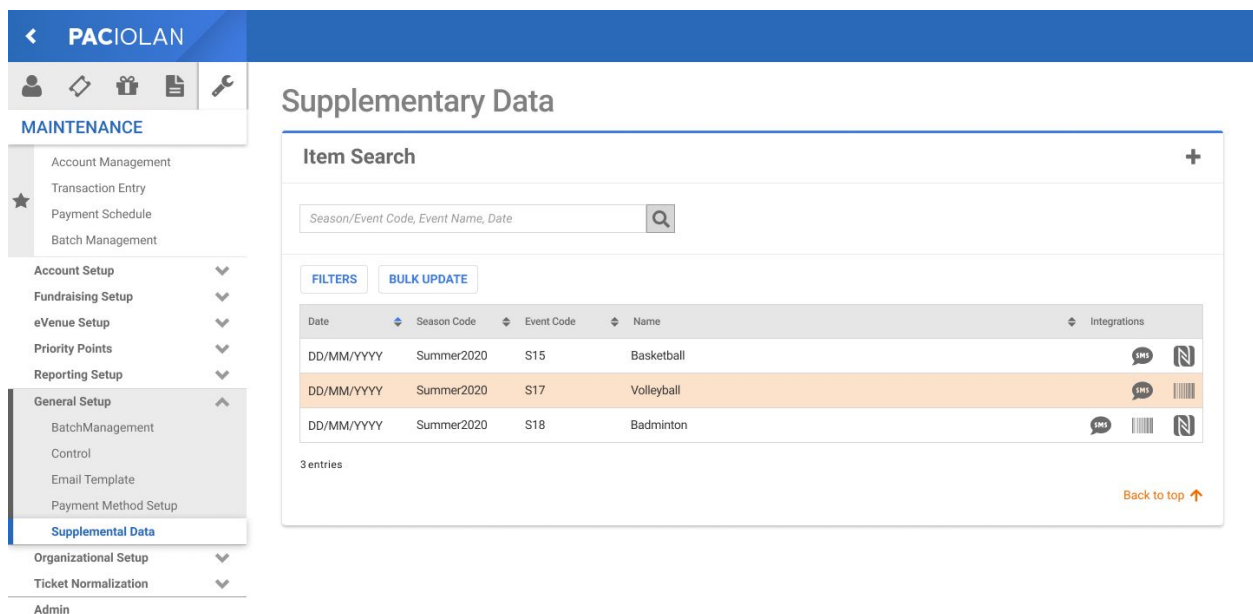


(Search Result Mockup Version 2: Using modal to present different filtering options for the results.)



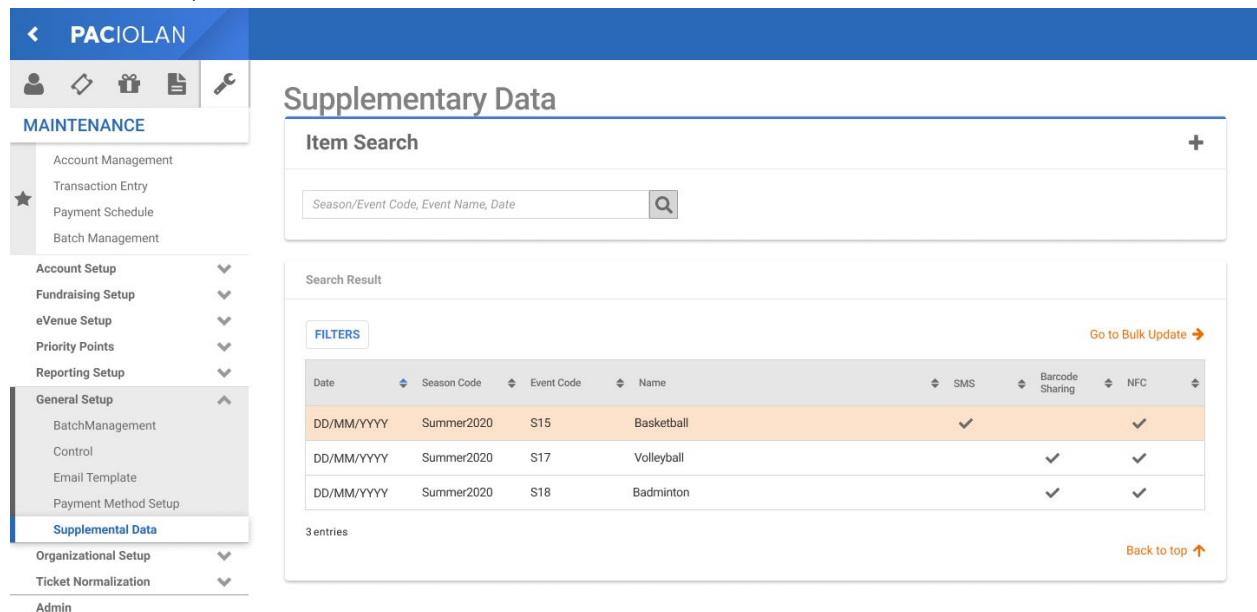
UPDATED Mockup:

(**Search Result Mockup Version 3:** New columns to show what supported technologies was used for each events)

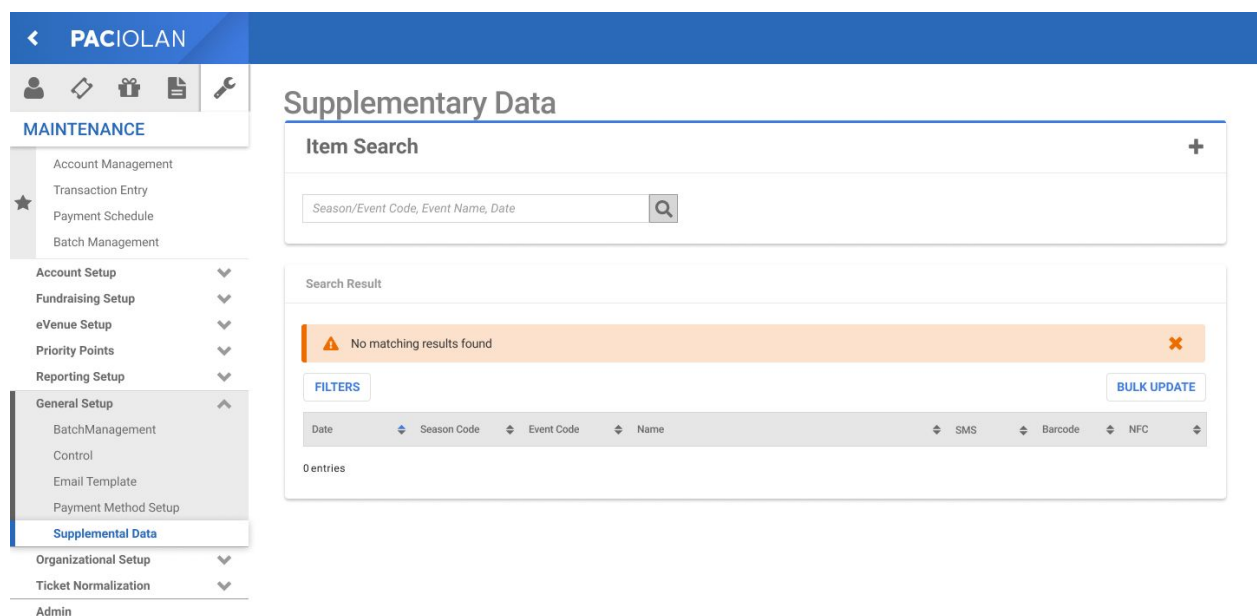


- Using unique icons to represent each particular technologies(e.g. NFC enabled, SMS, Barcode Sharing)
- Updated the color of sorting arrows on each column header
- Added a link at the bottom for user to navigate back to the top of the page

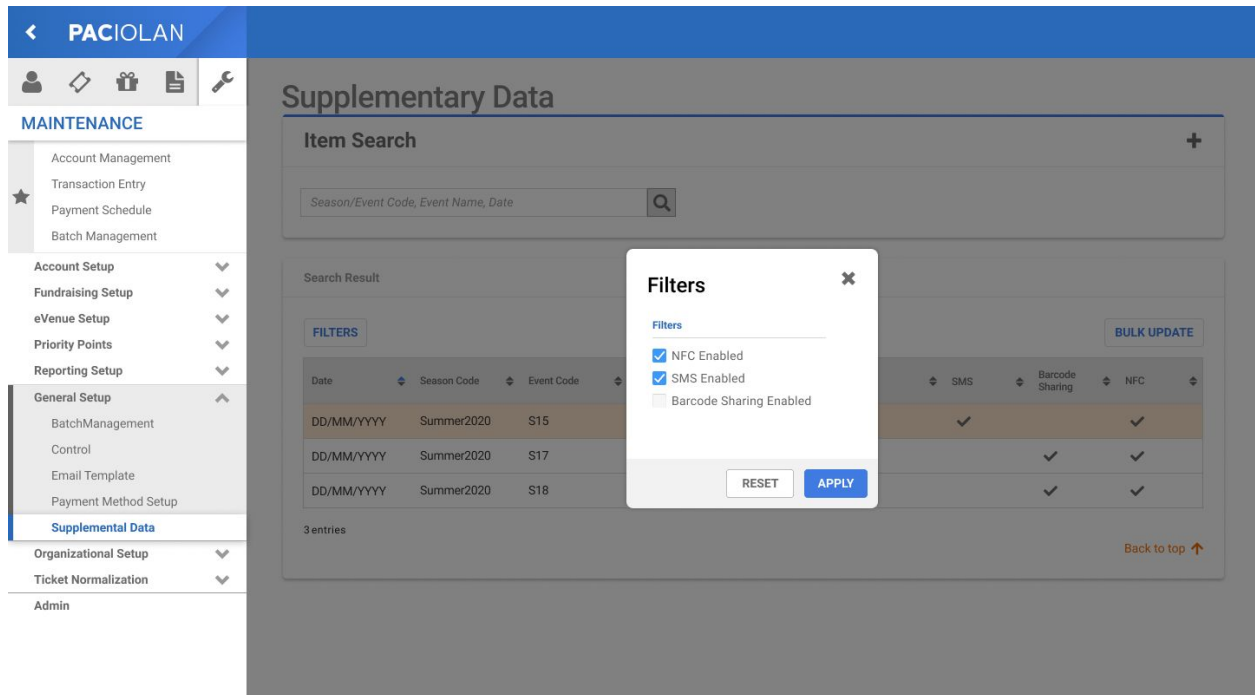
(Search Result Mockup Version 4: Using check marks instead of icons for each supported technologies for each events)



- Changed the icons from version 3 to checkmarks for easier recognition
- Changed the bulk update button to an orange link that matches our new bulk update page. Hopefully, the different styling would improve signaling for users to recognize that control of the page will be transferred to a different page when the link was pressed.



- New error notification that matches Paciolan's UX/UI design guidelines



- Simplified filtering options
- Changed from radial buttons to checkboxes to indicate that multiple options could be selected at once.

Bulk Update Page

Below is the SDM Bulk Update Page mockup which displays a list of events taken from the search result page for users to choose from to be bulk updated. It shows different bulk update options and to enable or disable SMS, NFC.

We have developed several different versions of the UI, which varies slightly in the presentation of the table, but the user flow (including pop-up windows) are finalized.

(Bulk Update Mockup 1: Shows the original state of events taken from the search result page.)

Bulk Update Mockup 1 - Version 1:

<

PACIOLAN

ACCOUNTS

Account Management

Transaction Entry

Payment Schedule

Batch Management

Processing

Account Setup

Account Parameters

Institutions

Regions

Custom Fields

Import Configurations

Zip/Postal Codes

Priority Points

Tools

Supplementary Data

Go back to Search Results

Event Name	Event Code	Season Code	SMS	NFC	Check-In
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable

10 entries 4 changed

DONE

Bulk Update Mockup 1 - Version 2:

- The enable feature columns (with checkboxes) are moved to the left, which could be more intuitive to users when dealing with checkboxes.

Bulk Update Mockup 1 - Version 4:

PACIO LAN

ACCOUNTS

- Account Management
- Transaction Entry
- Payment Schedule
- Batch Management

Account Setup

- Account Parameters**
 - Institutions
 - Regions
 - Custom Fields
 - Import Configurations
 - Zip/Postal Codes
- Priority Points
- Tools

Supplementary Data

[Go back to Search Results](#)

<input type="checkbox"/>	SMS	NFC	Barcode Sharing	Season Code	Event Code	Event Name
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	F19	F01	Football vs Washington State
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	F19	F01	Football vs Washington State
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	F19	F01	Football vs Washington State
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	F19	F01	Football vs Washington State
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	F19	F01	Football vs Washington State

10 entries 4 changed

[DONE](#)

- The checkboxes are also left aligned within the columns, which is also more intuitive than the previous version and has a cleaner look.
- This version is preferred by our sponsor.

(Bulk Update Mockup 2: After making changes, the orange highlight indicates modified events.)

PACIOLAN

ACCOUNTS

Account Management

Transaction Entry

Payment Schedule

Batch Management

Processing

Account Setup

Account Parameters

Institutions

Regions

Custom Fields

Import Configurations

Zip/Postal Codes

Priority Points

Tools

Supplementary Data

Go back to Search Results

Event Name	Event Code	Season Code	SMS	NFC	Check-In
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable

10 entries 4 changed

DONE

(Bulk Update Mockup 3: Warns the user when they disable previously enabled features.)

PACIOLAN

ACCOUNTS

Account Management

Transaction Entry

Payment Schedule

Batch Management

Processing

Account Setup

Account Parameters

Institutions

Regions

Custom Fields

Import Configurations

Zip/Postal Codes

Priority Points

Tools

Supplementary Data

Go back to Search Results

Warning

You have disabled feature(s) that were previously enabled. Are you sure you want to do this?

☐ Don't show me again

NO

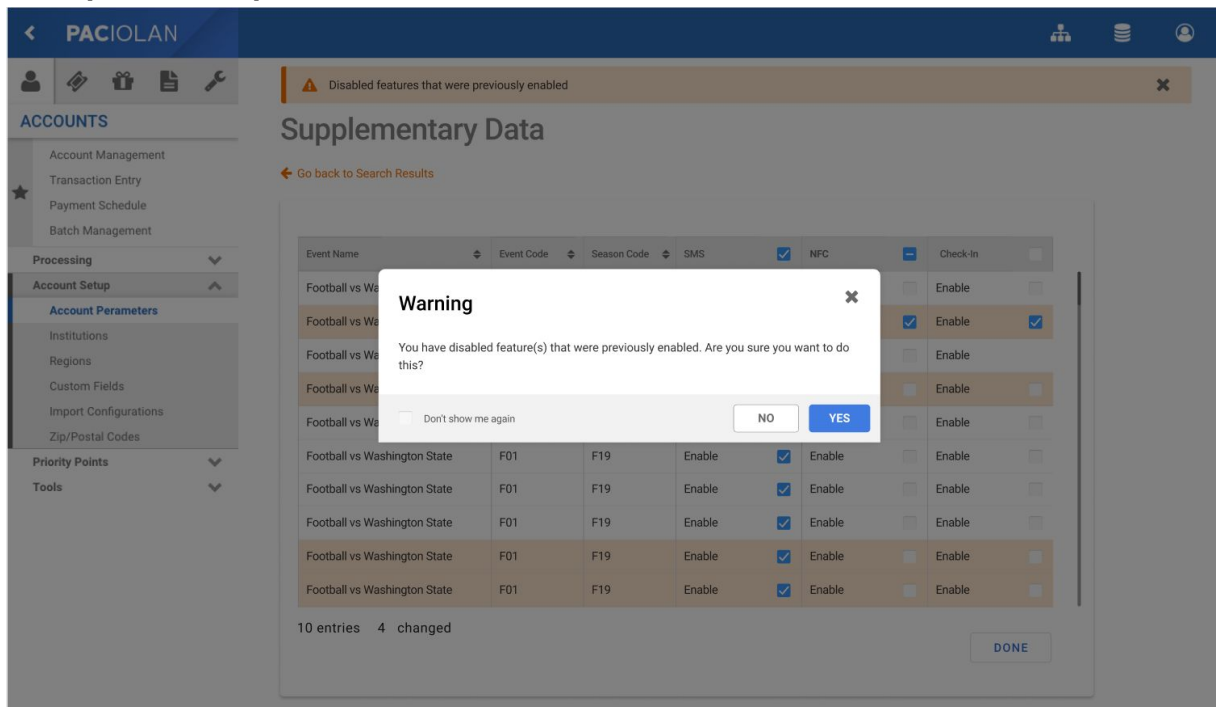
YES

Event Name	Event Code	Season Code	SMS	NFC	Check-In
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable
Football vs Washington State	F01	F19	Enable	Enable	Enable

10 entries 4 changed

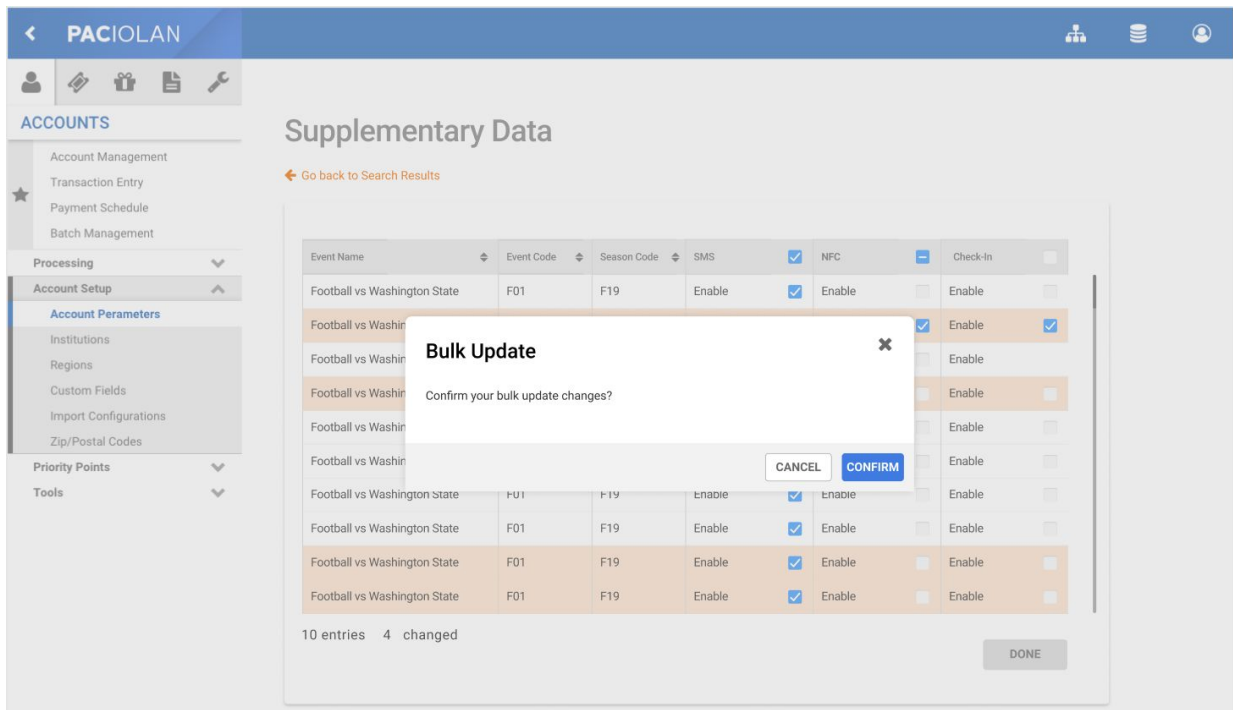
DONE

Bulk Update Mockup 3 - Version 2:

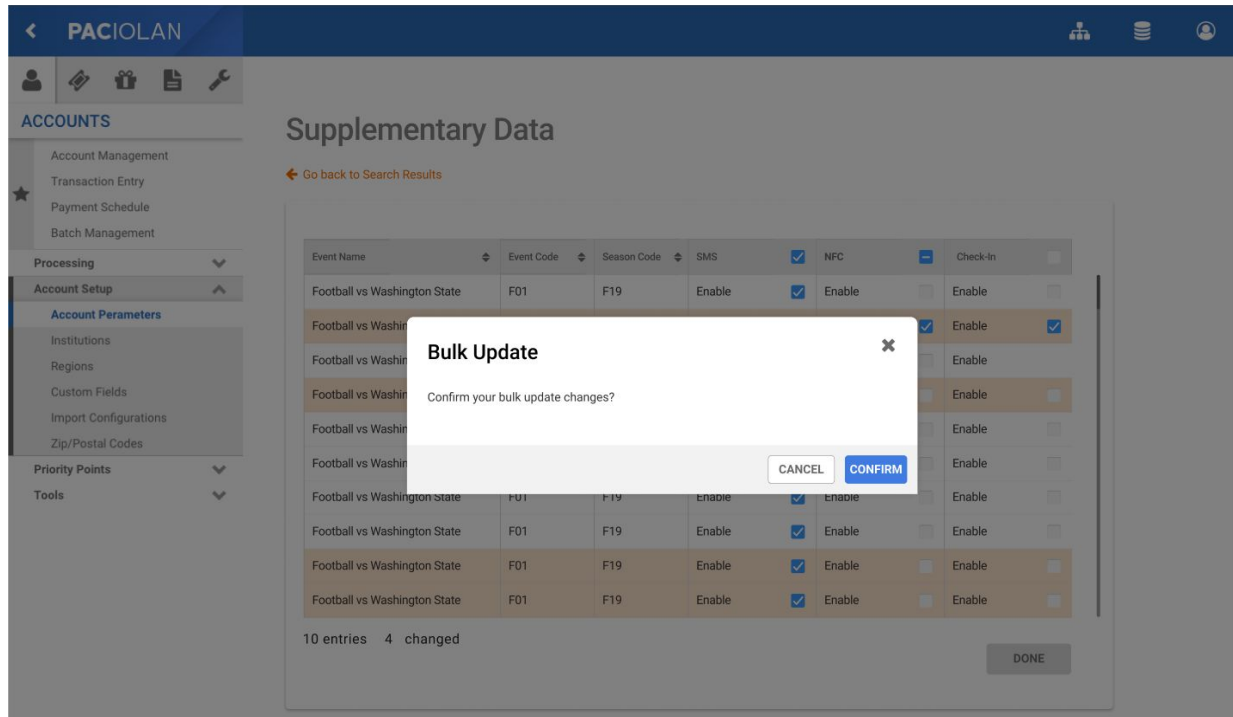


- Updated to match Pac8 style guide.

(Bulk Update Mockup 4: Prompts the user to confirm their bulk update changes.)



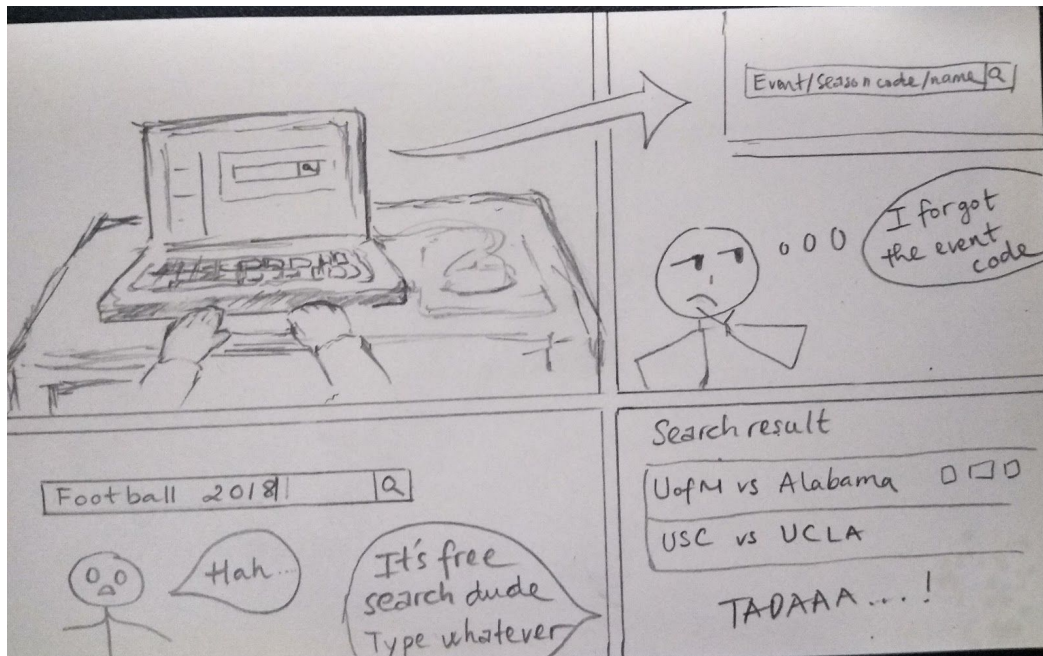
Bulk Update Mockup 4 - Version 2:



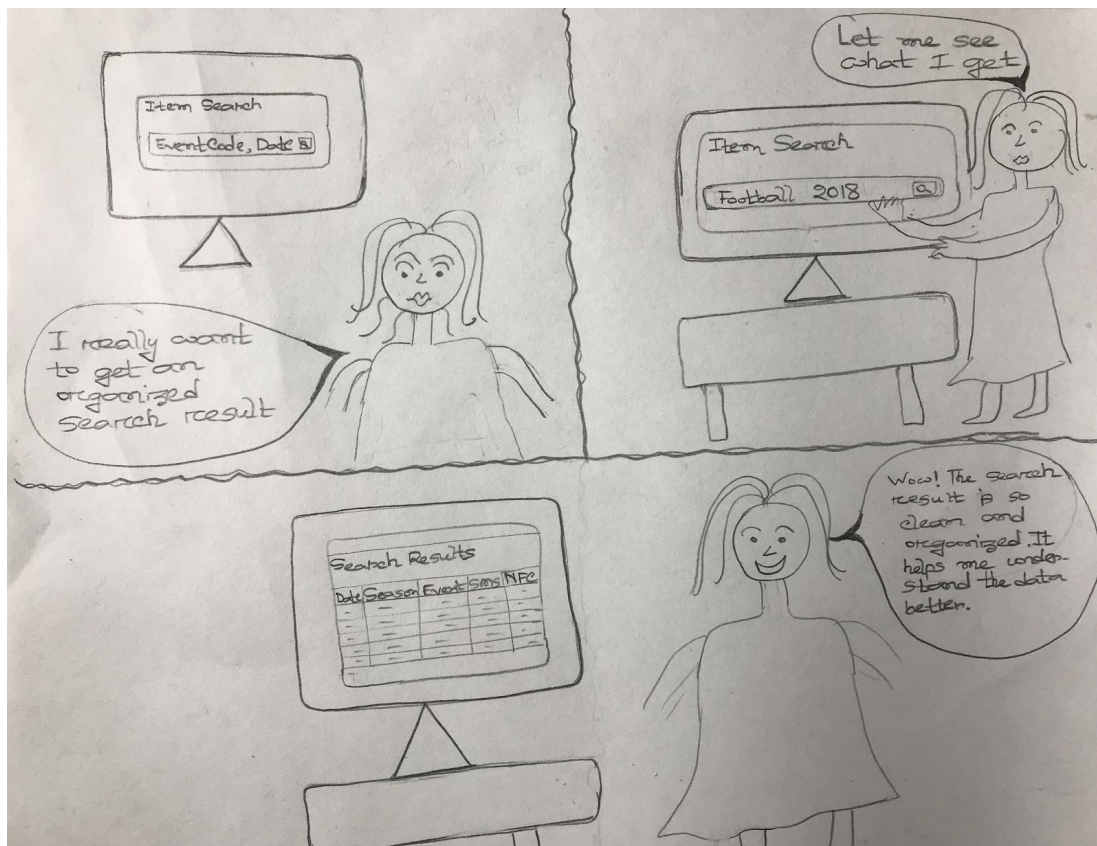
- Updated to match Pac8 style guide.

Storyboards

Use case 1: Free search function



Use case 1: Free search function (updated for this sprint)



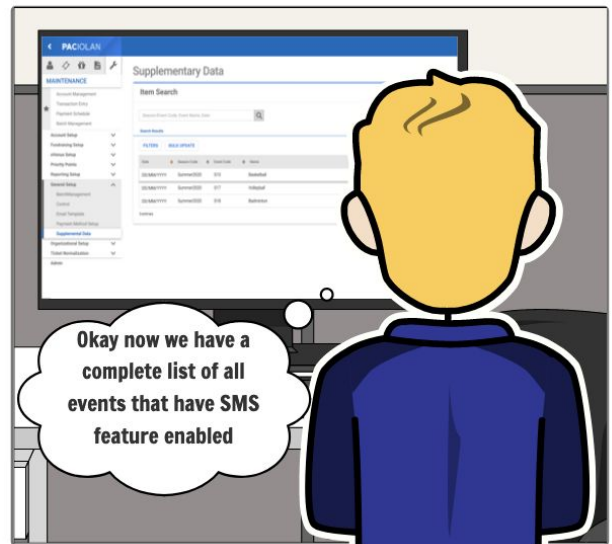
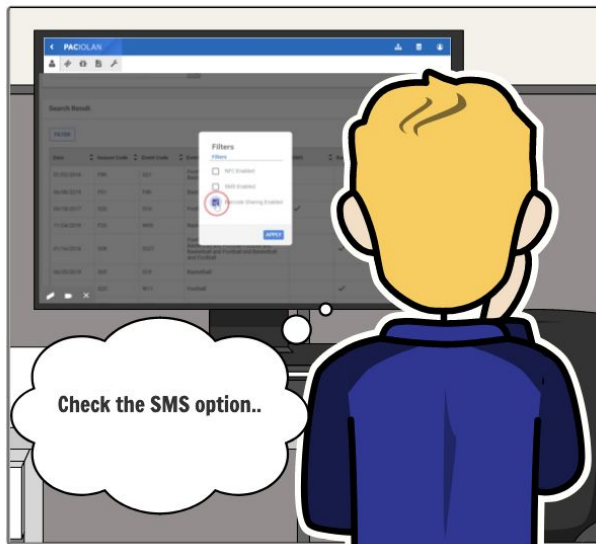
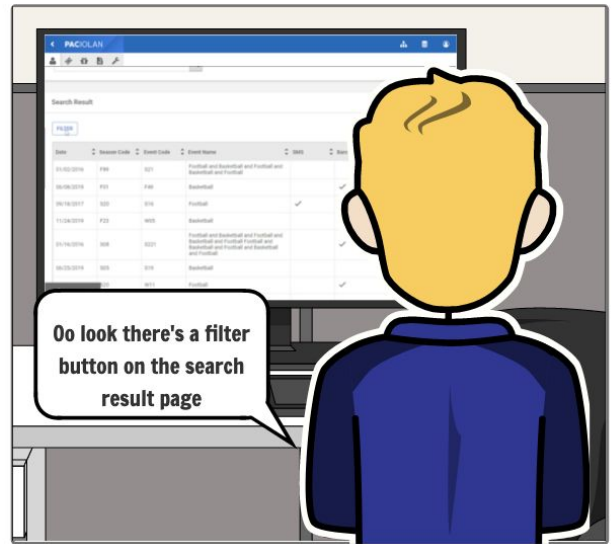
Use case 2: Bulk update



Use case 2: Bulk update (pop-up modals)



[Use case 3: Search result page \(updated for this sprint to include filtering feature\)](#)



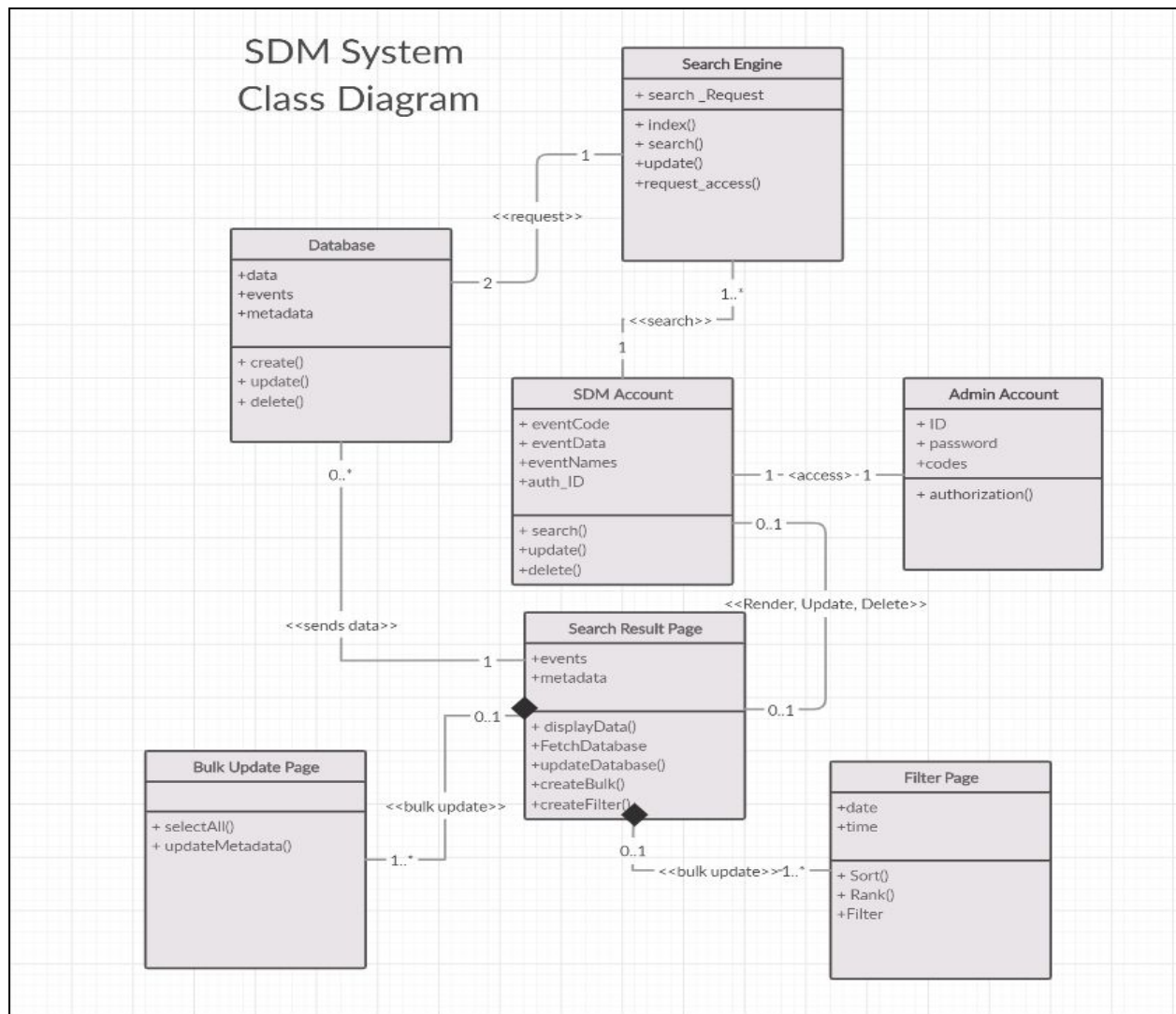
[Use case 4: Search result page \(updated for this sprint to include sorting feature\)](#)

Section 3: Software Design

SDM System Class Diagram

Below is a class/object diagram of our system that we are working on so far. Since the technical requirements for this project deal more with objects, requests, responses, and API interactions, we focused more on the behavioral aspects of how the different components of our system interact with each other.

UML Class Diagram (Overall)



3.1 Admin Account

This is the account that is specific to the person that is going to access the SDM tool.

3.2 SDM Account

In order to access the SDM tool, the Admin Account must have the authorization to access the SDM account and its tools by having an authorization ID.

3.3 Search Engine

Search Engine service allows the user to access its functionalities through its API to query the database for information. Since it is a CRUD API, it can also update the database, create data, read, and delete.

3.4 Database

The database stores all the necessary information, metadata, events, etc.

3.5 Results Page

The results page contains 2 other subpages, the filter page, and the bulk update page. Through the server, this page will fetch the data and display it.

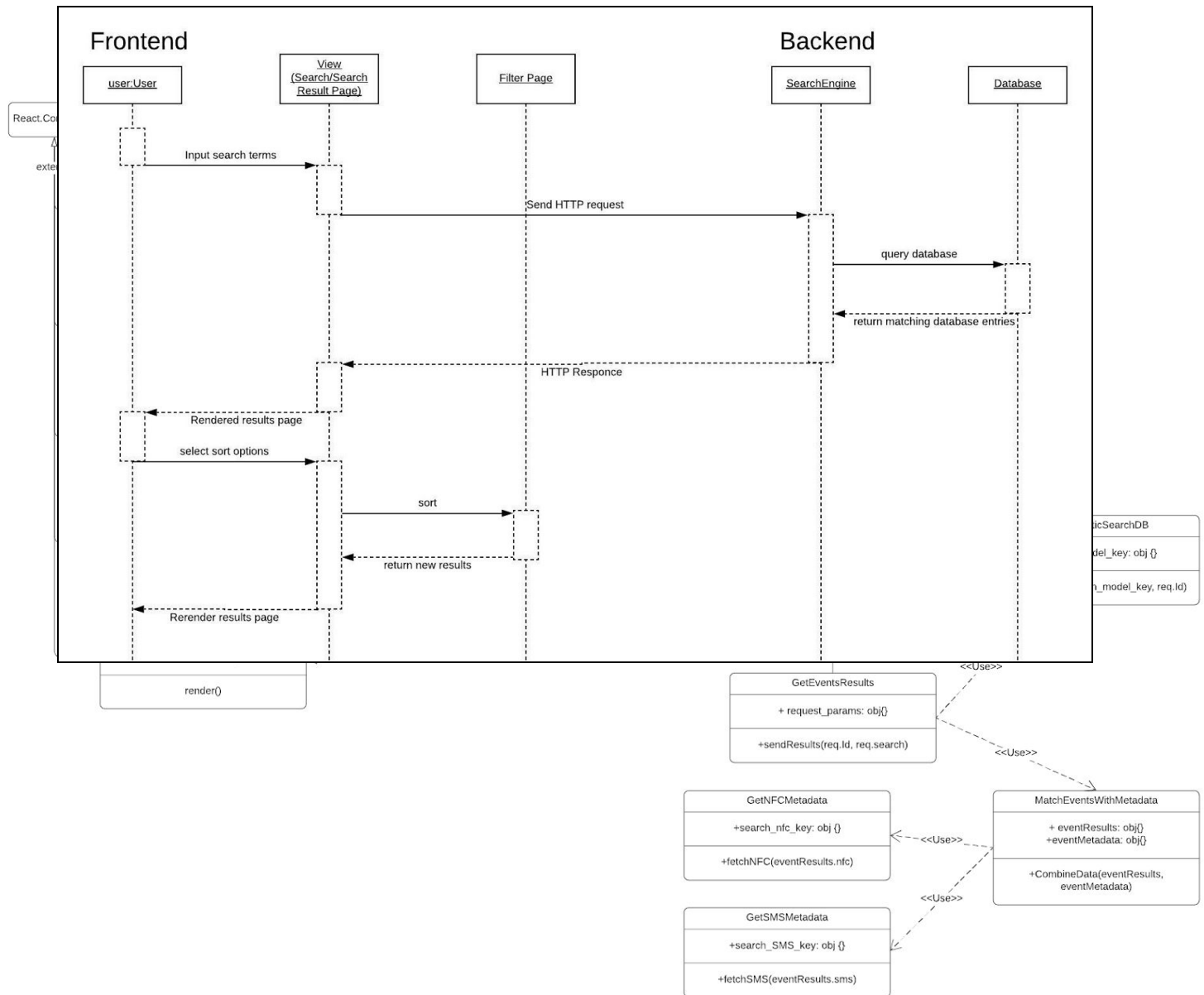
3.6 Bulk Update Page

This page will allow the user to edit contents in mass quantity, and once it's saved, it will update the original page to match.

UML Class Diagram (Updated for this sprint)

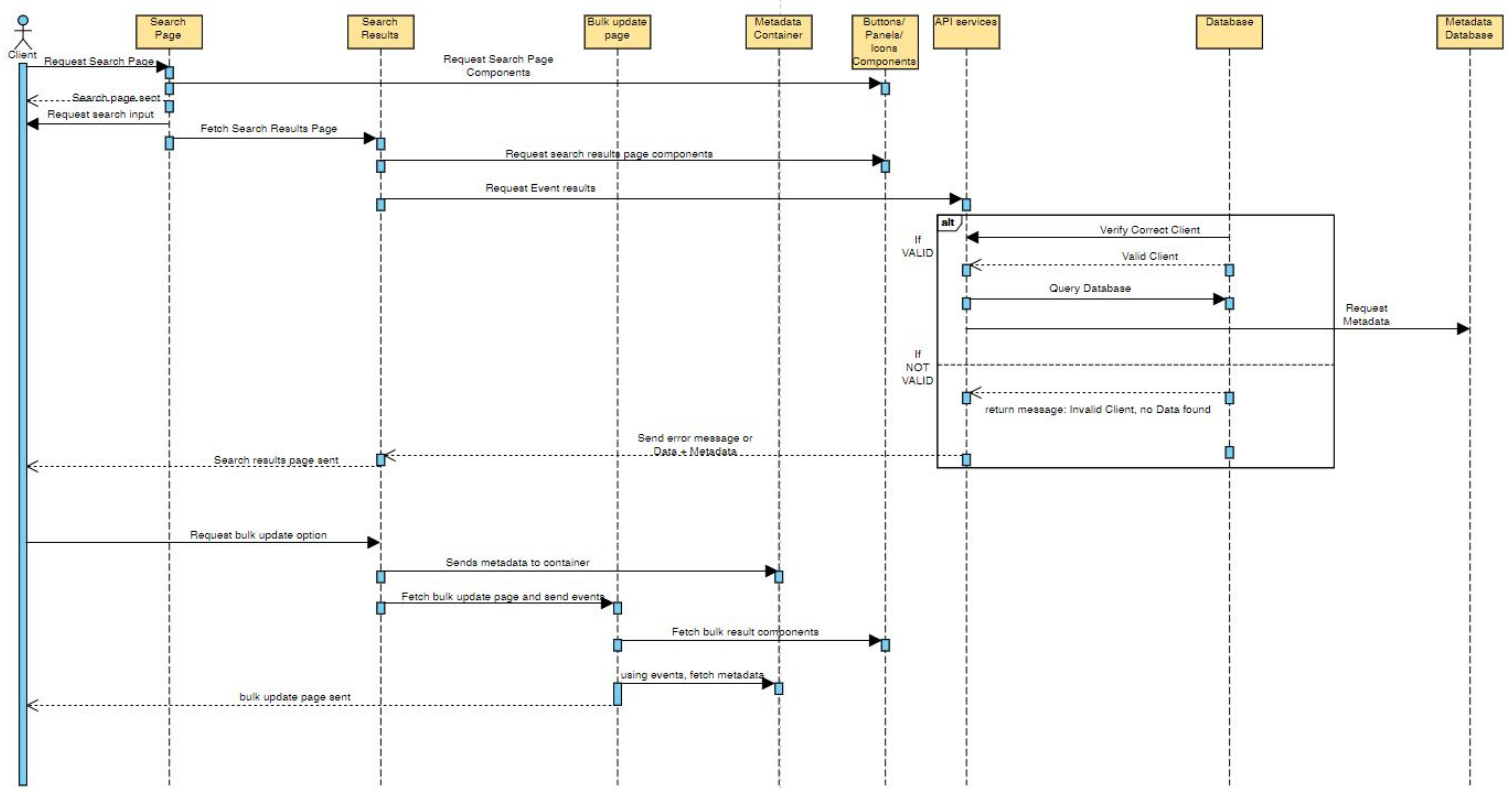
UML Sequence Diagrams:

1.0 Conduct Free-form Search



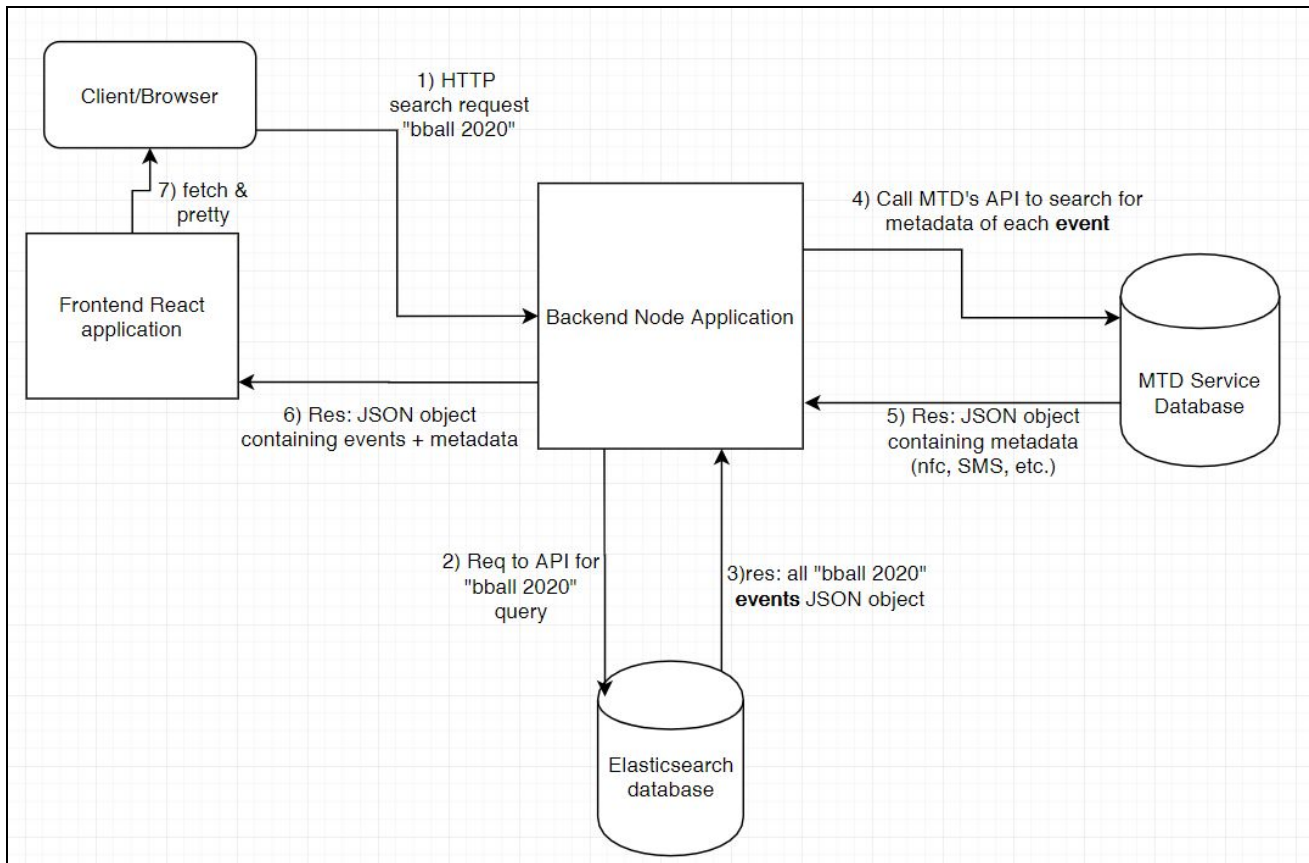
2.0 Conduct Search/Request data/Display results/Bulk update feature

(full image here: <http://prntscr.com/rcv4ge>)



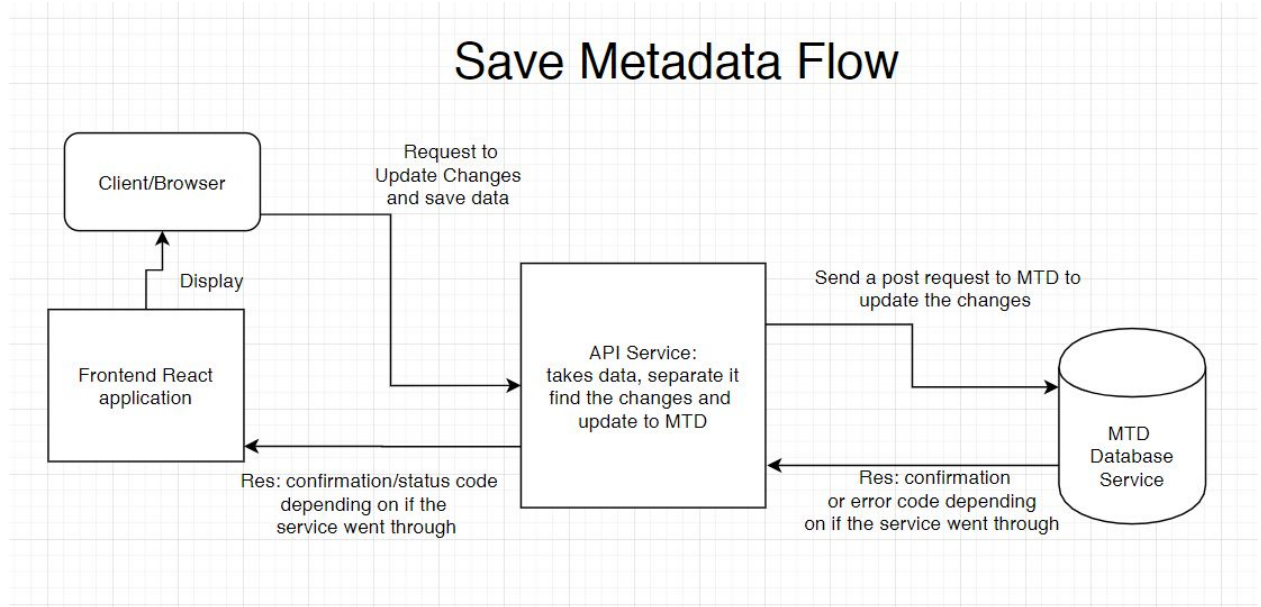
Additional Diagrams

1. Backend: SDM Search Flow Diagram



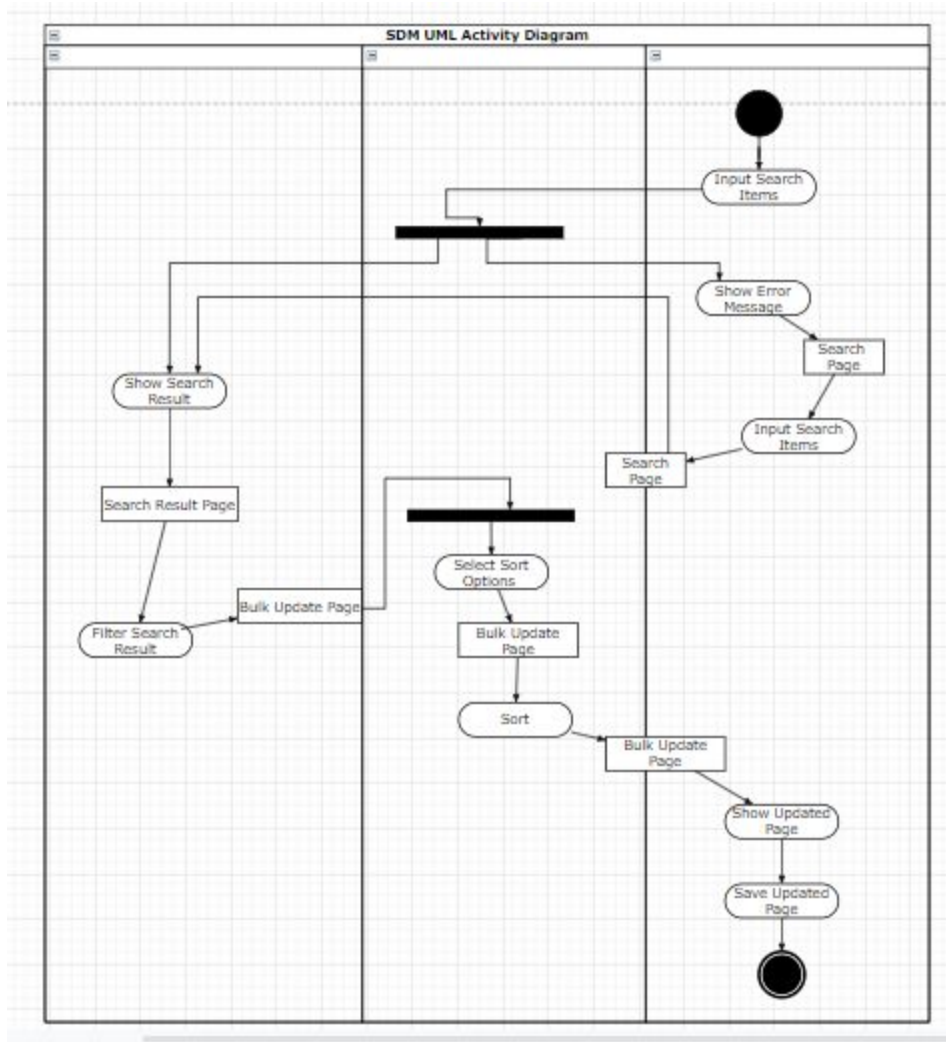
2. Backend: SDM Update/Save Data API

a. For last sprint



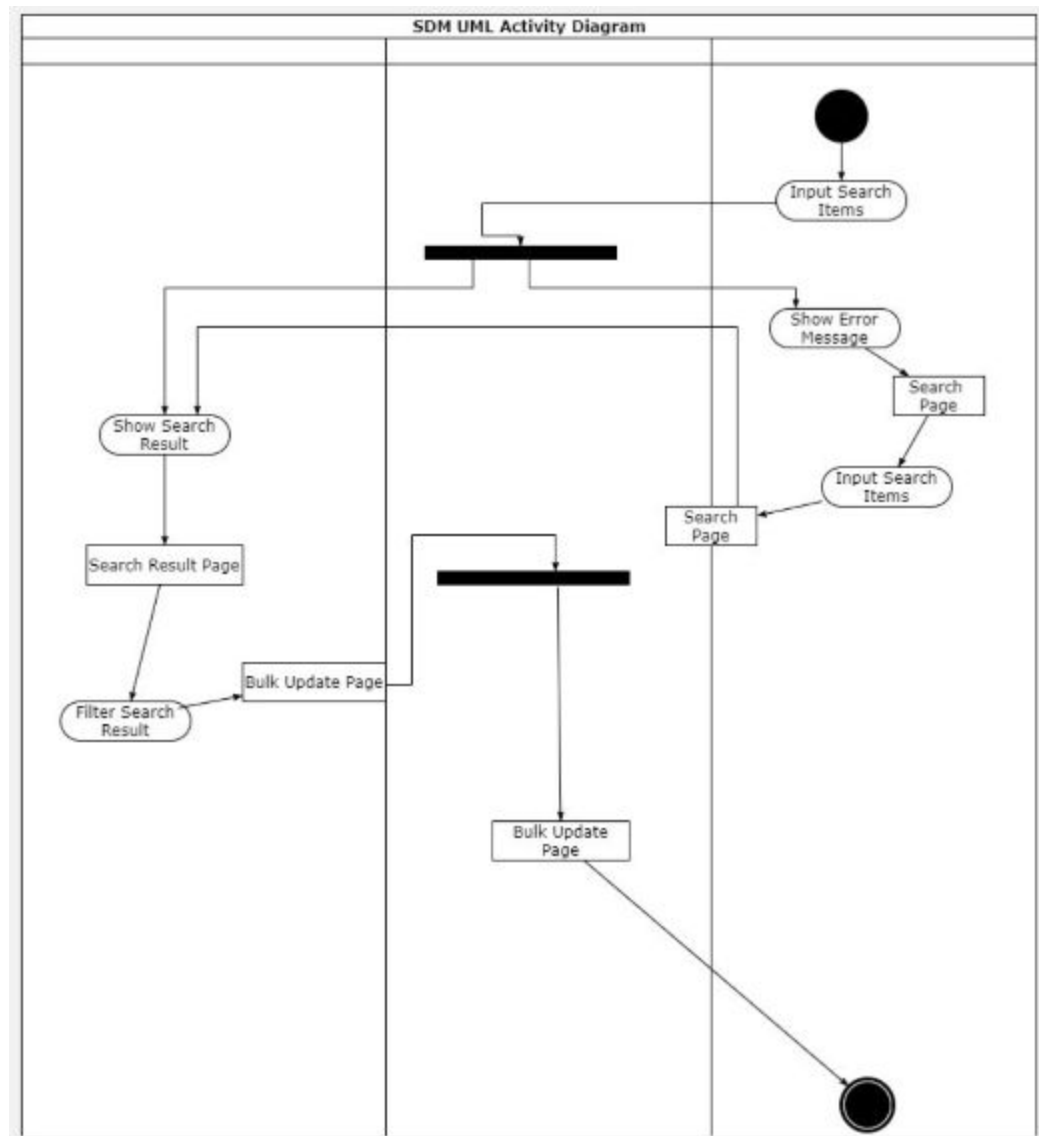
3. SDM UML Activity Diagram

a. Overall



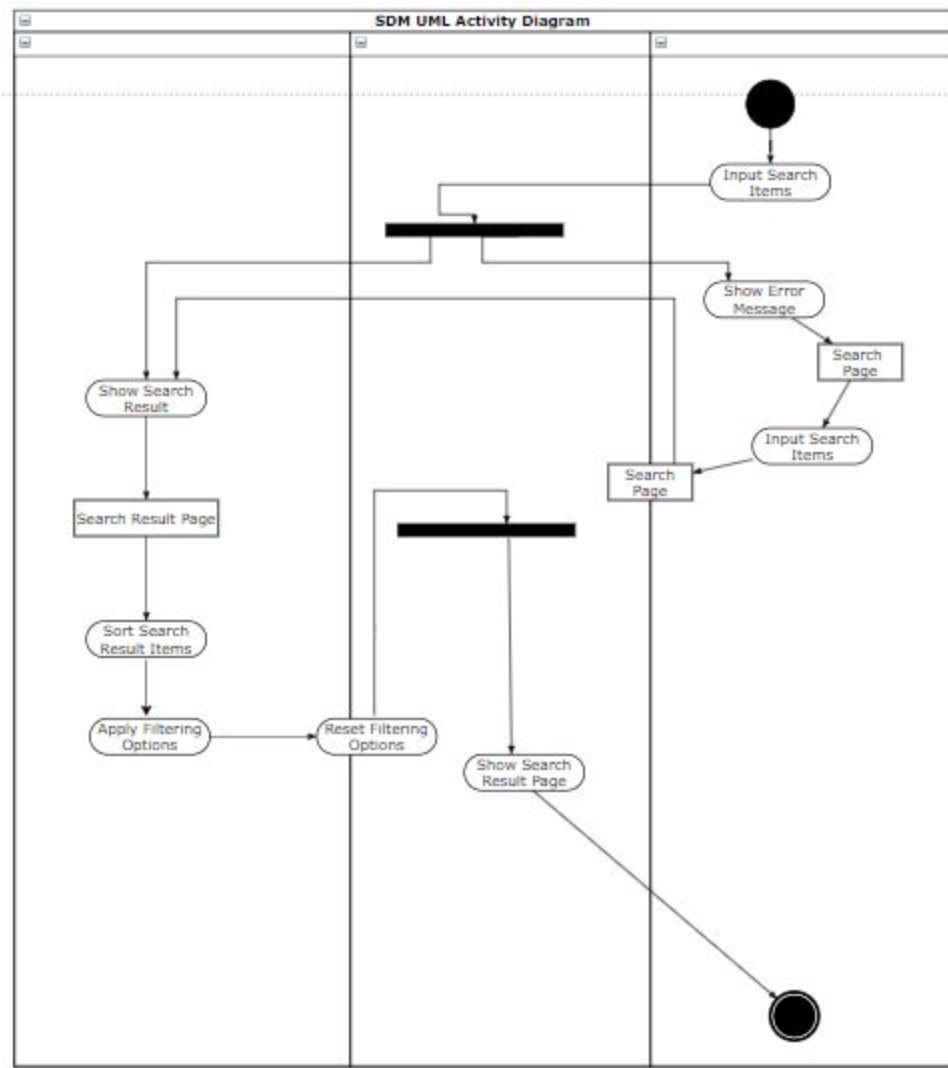
4. SDM UML Activity Diagram

a. From last sprint



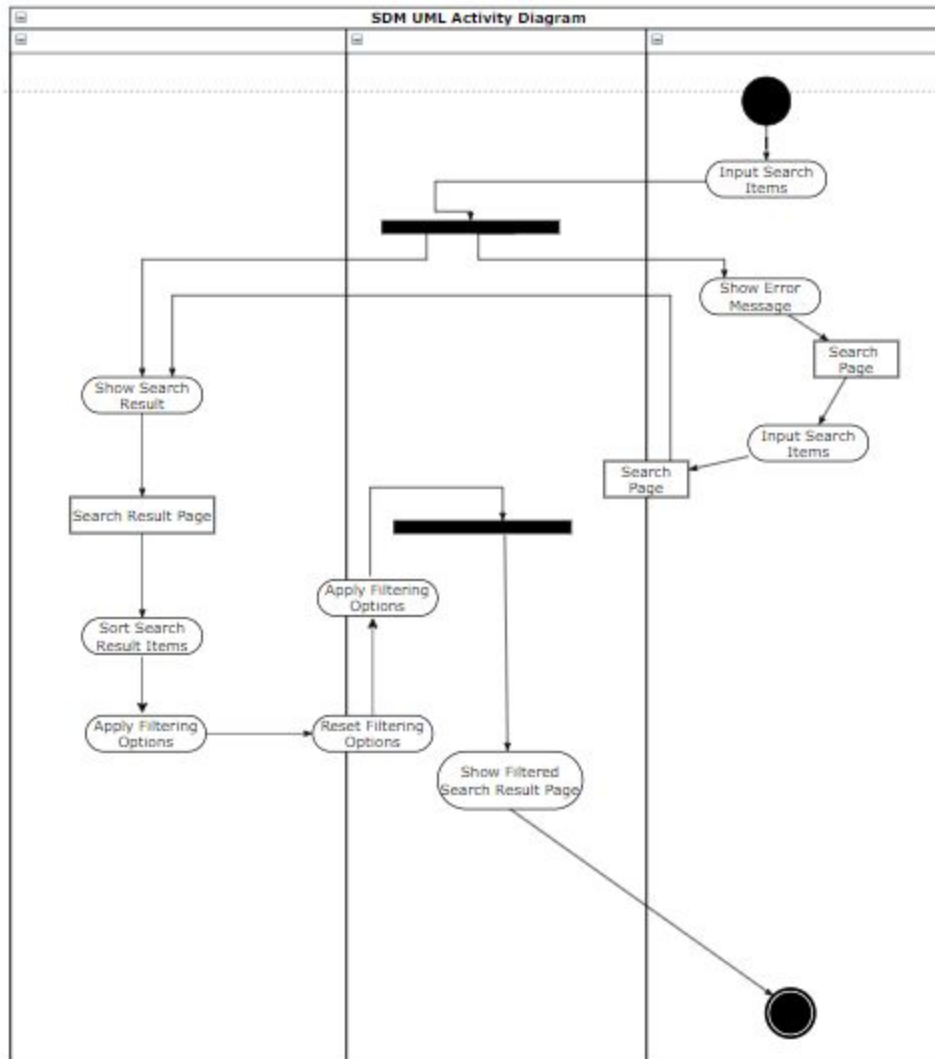
5. SDM UML Activity Diagram

a. From last sprint

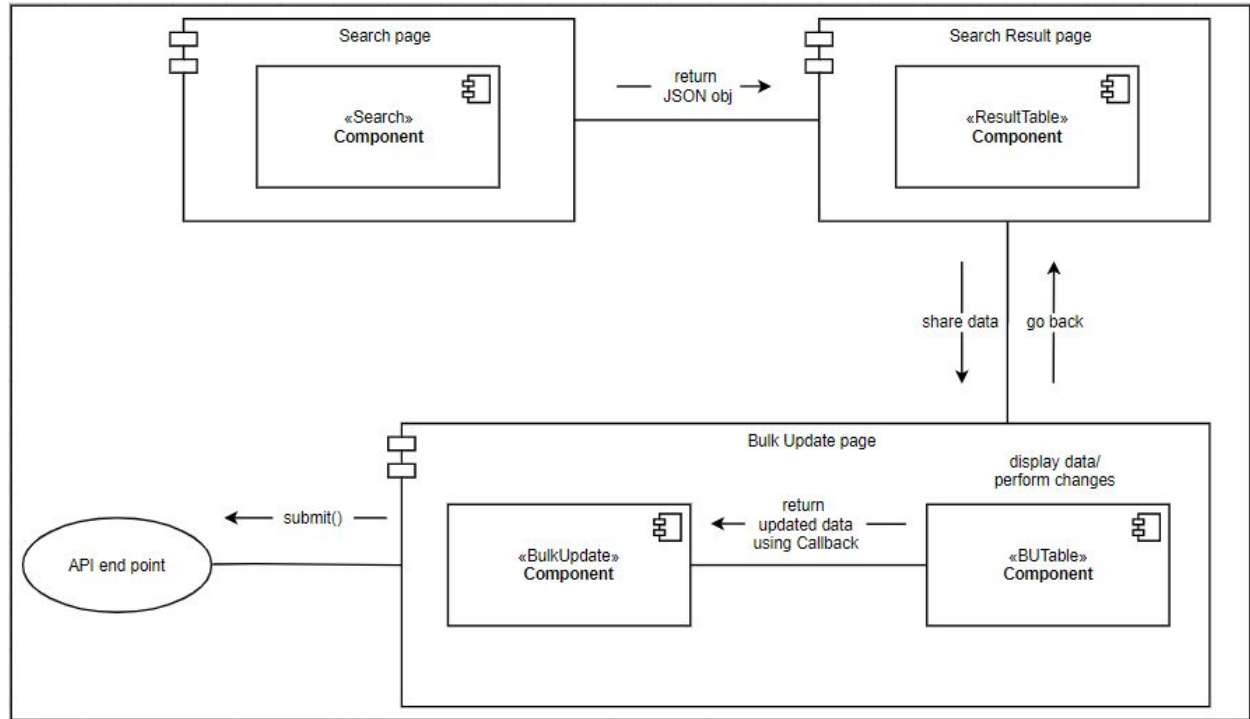


6. SDM UML Activity Diagram

a. Updated for this sprint



7. SDM App Communication diagram (updated for this sprint)



8. Simple test flow diagram for Microservices

a. From Last Sprint

