DISPATCH

A Case Study In Improving User Experience

DISPATCH TODAY

DSP

- Select Stocks
- Map and Route Stocks
- Review Batch

DSPM

- Reverse Dispatch
- Dry Run
- Login
- Logout
- Change Destination
- Release Problem
- Review Batch



WHO DID WE TALK TO

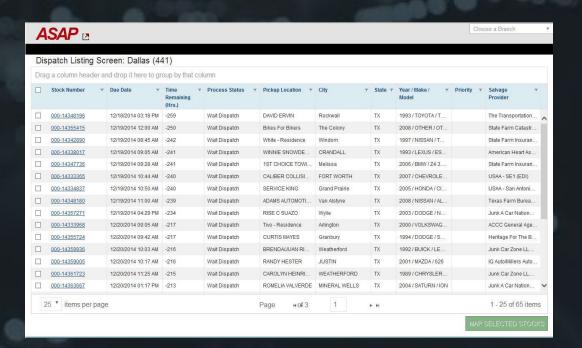


- Trainers
- Branch Managers
- Branch Dispatchers



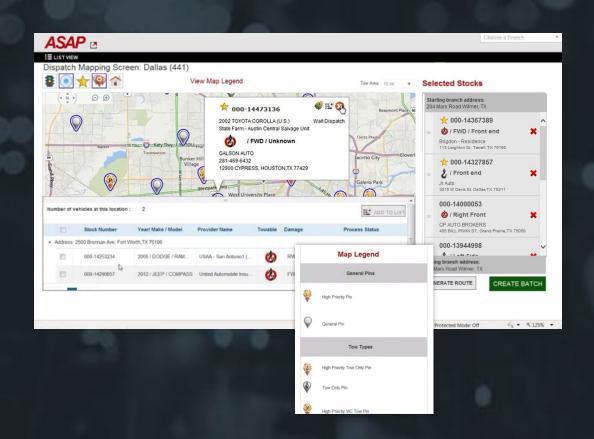
LISTING SCREEN

- Consolidate Listing and Mapping screens
- Make all stocks visible in the listing table; remove pagination
- 3. Ability to add/remove columns
- 4. Table functionality virtually unknown to users, ie. sort, filter, group
- Add Automatic Refresh to see stocks real time



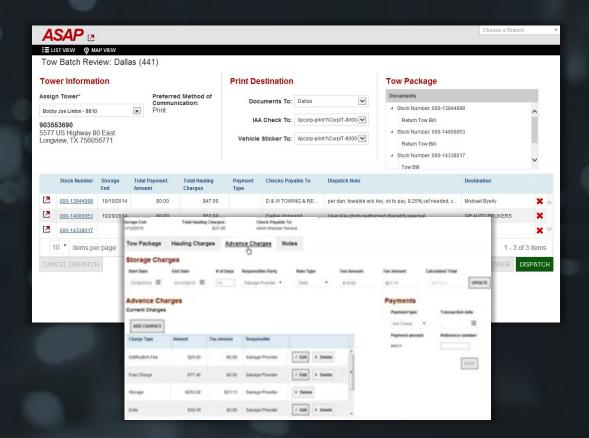
MAPPING SCREEN

- 1. Icons don't mean anything
- 2. Page is too busy; too much information
- 3. Most don't use generate routes feature
- Once stock is selected, it would be nice if it would be grayed out
- 5. Map is buggy



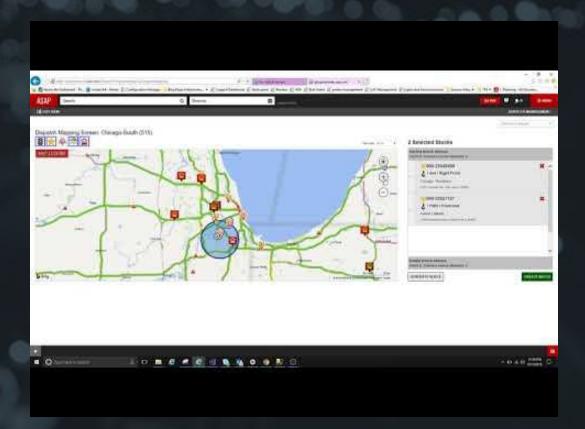
TOW BATCH REVIEW SCREEN

- 1. Would like to manage Logout functionality (edit, undo, bulk).
- Legibility is difficult (color, text-size, etc.)



PERFORMANCE ISSUES

1. All around performance is slow



WHAT ARE THE KEY AREAS OF CONCENTRATION

- Eliminate Redundancy
- Responsive, Modern Layout Design
- Optimize Performance
- Architect new workflow for DSP and DSPM

WHAT DOES FUTURE DISPATCH LOOK LIKE?

 Exploring Design using mobile first methodologies Improving Desktop Performance with a modern tech stack

MOBILE PROTOTYPE

KEY FEATURES

- Combines List and Map Screens
- Entire application reduced to
 3-Step Form w/ validation
- Eliminates need for tables/ pagination while retaining filter, sort data

- Seamless DSP to DSPM Integration
- Added "Settings" for user to configure repeatable steps
- Lists Batches by created date
- Map pins more discernable

DESKTOP ANGULAR APPLICATION

KEY FEATURES

- Handles automatic refresh through HTTPClient/Observable
- Local Storage Memory settings for Branch Properties

- DB: Firebase Cloud Firestore
- Google Maps VS Bing Maps
- Material Design VS Kendo

