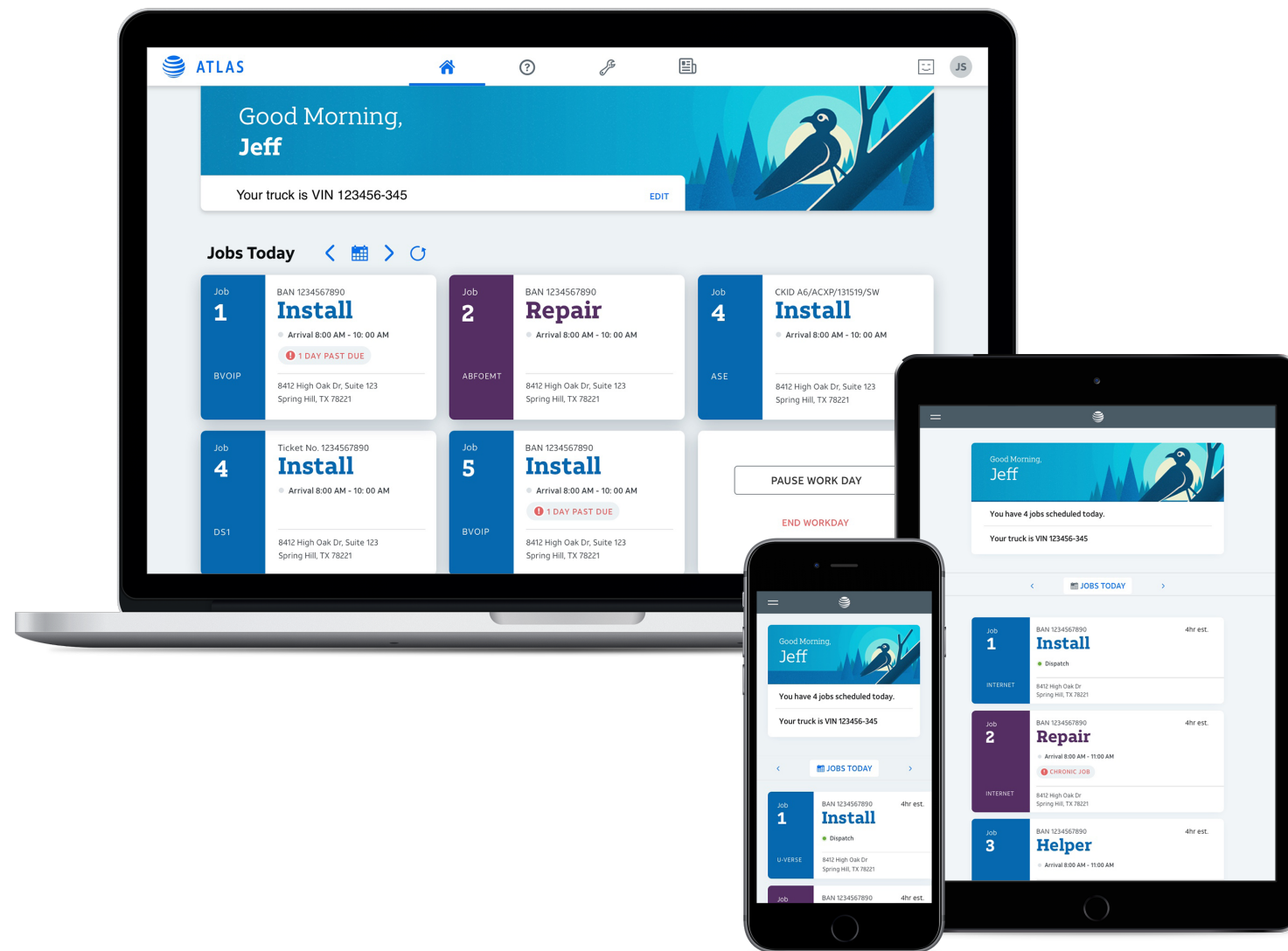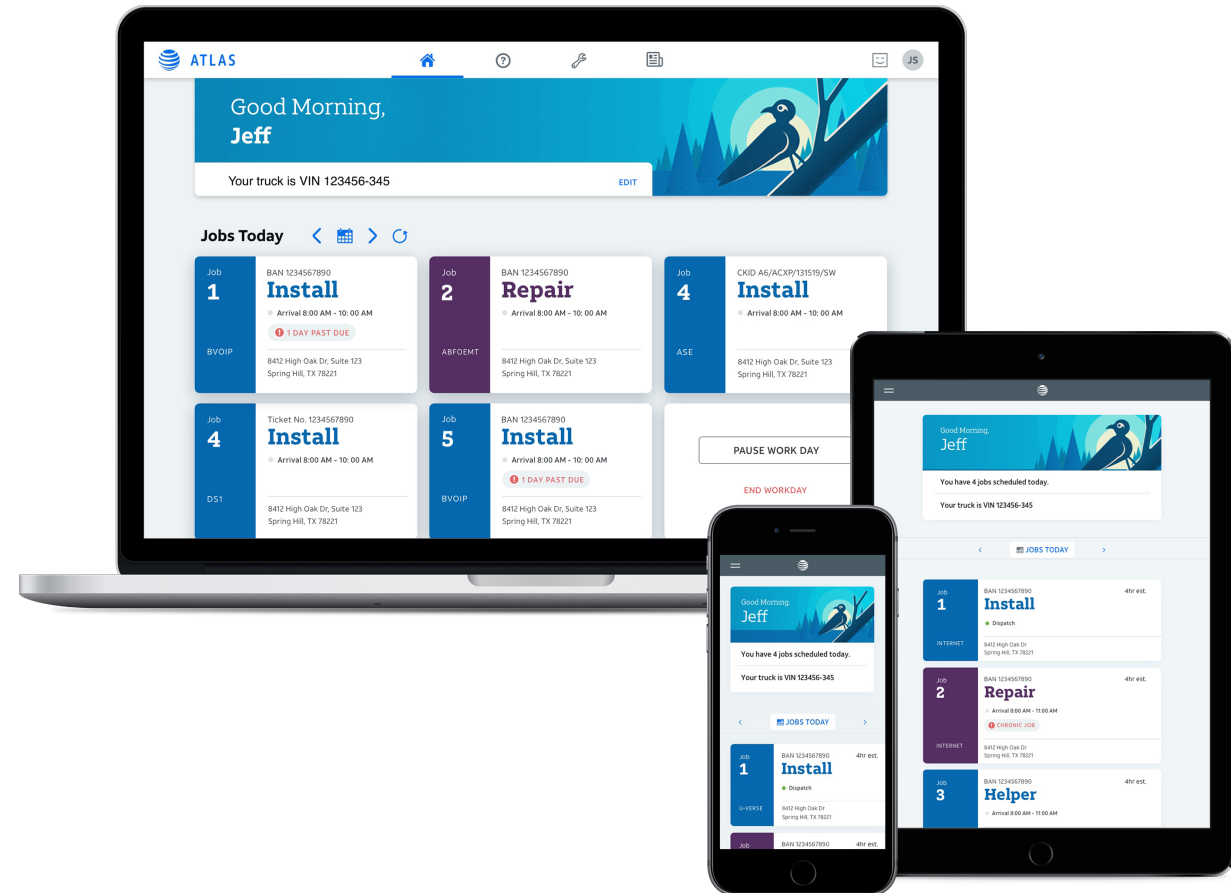# Design Mobile & Desktop Application For AT&T Technicians
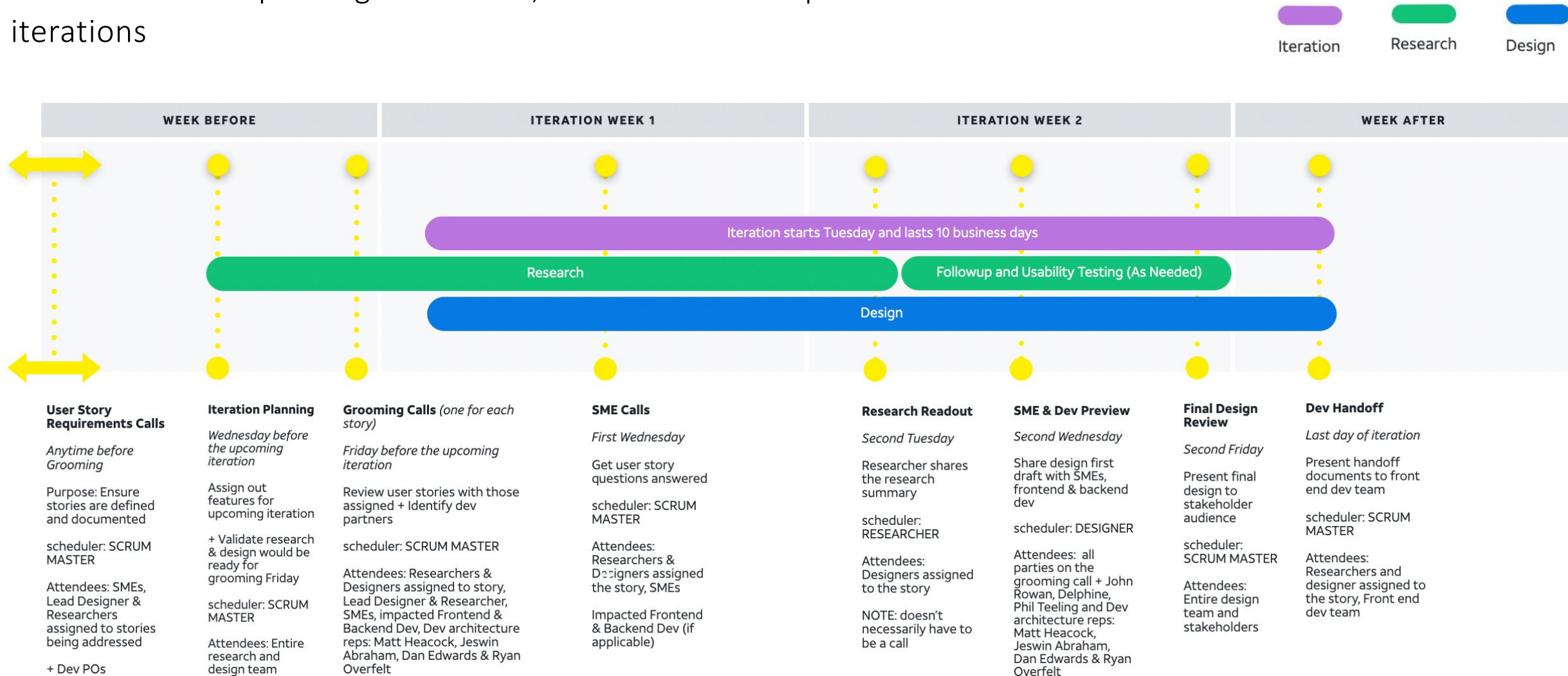
AT&T- ATLAS

ATLAS is an internal app for AT&T technicians. Currently, techs using a lot of outdated legacy systems, ATLAS is set to replace those systems & let

technicians use 1 app to perform multiple tasks, so it can be a toolbox of some sort for all digital tasks.

ATLAS team is adept at the agile process, where each iteration takes 2 weeks. Depending on the size, feature can take up 1-4 iterations
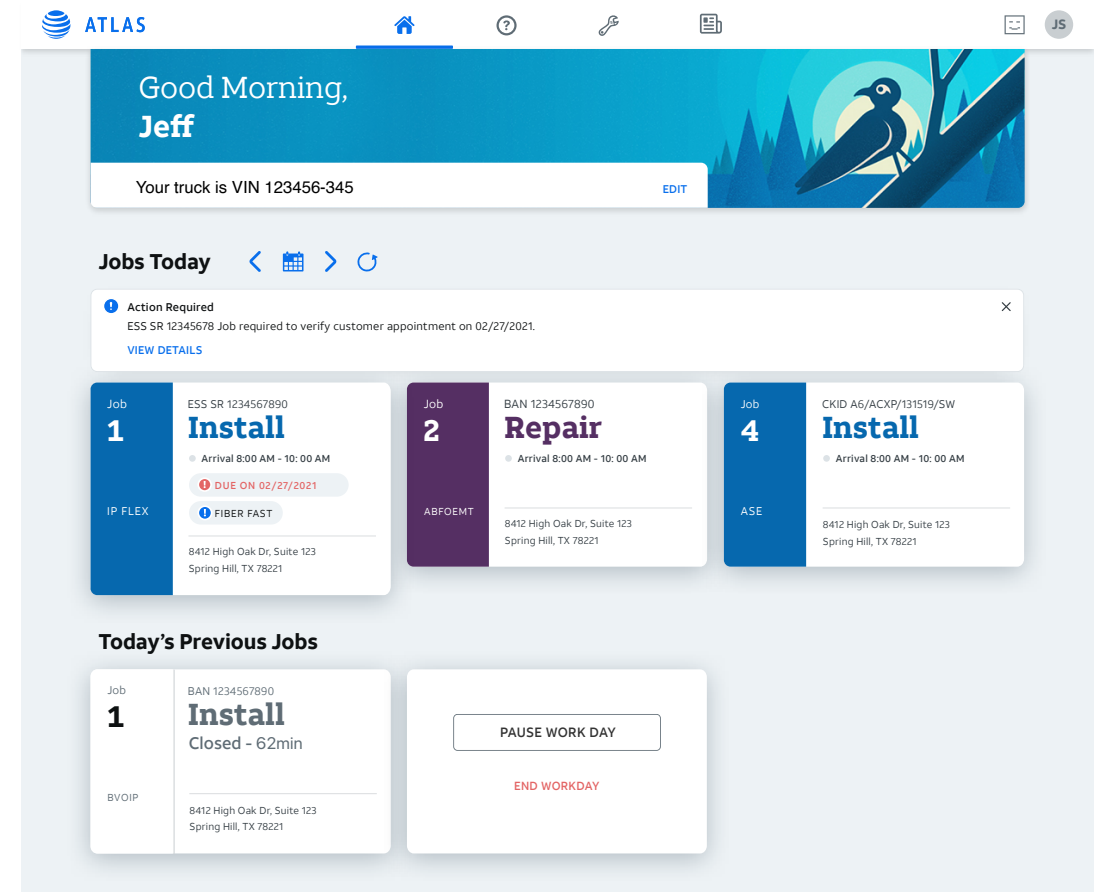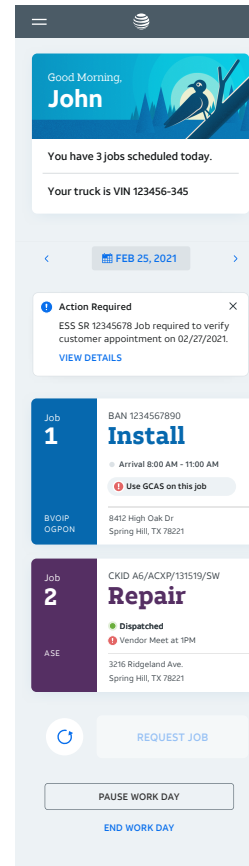
Iteration   Research   Design

WEEK BEFORE	ITERATION WEEK 1	ITERATION WEEK 2	WEEK AFTER

Iteration starts Tuesday and lasts 10 business days

Research | Followup and Usability Testing (As Needed)

Design

User Story Requirements Calls

Anytime before Grooming

Purpose: Ensure stories are defined and documented

scheduler: SCRUM MASTER

Attendees: SMEs, Lead Designer & Researchers assigned to stories being addressed

+ Dev POs

Iteration Planning

Wednesday before the upcoming iteration

Assign out features for upcoming iteration

+ Validate research & design would be ready for grooming Friday

scheduler: SCRUM MASTER

Attendees: Entire research and design team

Grooming Calls *(one for each story)*

Friday before the upcoming iteration

Review user stories with those assigned + Identify dev partners

scheduler: SCRUM MASTER

Attendees: Researchers & Designers assigned to story, Lead Designer & Researcher, SMEs, impacted Frontend & Backend Dev, Dev architecture reps: Matt Heacock, Jeswin Abraham, Dan Edwards & Ryan Overfelt

SME Calls

First Wednesday

Get user story questions answered

scheduler: SCRUM MASTER

Attendees: Researchers & Designers assigned the story, SMEs

Impacted Frontend & Backend Dev (if applicable)

Research Readout

Second Tuesday

Researcher shares the research summary

scheduler: RESEARCHER

Attendees: Designers assigned to the story

NOTE: doesn't necessarily have to be a call

SME & Dev Preview

Second Wednesday

Share design first draft with SMEs, frontend & backend dev

scheduler: DESIGNER

Attendees: all parties on the grooming call + John Rowan, Delphine, Phil Teeling and Dev architecture reps: Matt Heacock, Jeswin Abraham, Dan Edwards & Ryan Overfelt

Final Design Review

Second Friday

Present final design to stakeholder audience

scheduler: SCRUM MASTER

Attendees: Entire design team and stakeholders

Dev Handoff

Last day of iteration

Present handoff documents to front end dev team

scheduler: SCRUM MASTER

Attendees: Researchers and designer assigned to the story, Front end dev team

Add and design Experience Advance call notification feature within the ATLAS app.

Company: AT&T   **User:** AT&T Technicians   **Role:** Sr. UX Designer

Project Summary

I designed the experience of ACN feature for ESS jobs, which helped to optimize technicians' time and efforts.

Responsibilities

- Experience Design
- Prototype & Usability Testing

Timeline & Team

4 weeks start to finish (2 iterations). With a collaboration of at least 7 people from different teams (Research, Design, Business, Architecture, Front-end, Back-end).

Results

The feature is currently been developed, but we've got some really good feedback from primary users during the usability testing. All technicians that participated in testing were very excited for this feature to come to life.

I **designed the experience** of decrementing parts for ESS jobs, which helped to **optimize technicians' time and efforts**.

Working within the **agile 2-week sprint system**, I have collaborated with business stakeholders, researcher and development teams, have **designed 2 flows** for Close Job &Standalone process, **done usability testing & optimized overall experience** for ESS technicians.

Understanding Requirements	Research & Discovery	Outlining Strategy	Experience Design	Prototype, Test & Iterate	Hand Off
• Meeting with business stakeholders • Going through the User Story	• Collaborating with researcher • Attending user interviews & making notes • Going over the discoveries	• Combining some findings and coming up with the recommendations • Collaborating with Architecture & Development teams, understanding API limitations	• Coming up with several potential solutions (3-5) • Presenting some of the solutions at the Initial Design Review • Selecting 2 potential versions	• Creating 2 flows in a pdf format • Usability testing with 5 techs • Finalizing 1 flow and creating a prototype in Figma • Final usability test with 3 techs	• Presenting final flow for a large group of stakeholders during Final Design Review • Preparing files for handoff: Sketch, pdf, Zeplin

We realized that overall the existing tool **fails** on **at least 5 usability heuristics**, which also gave us ideas for potential improvement
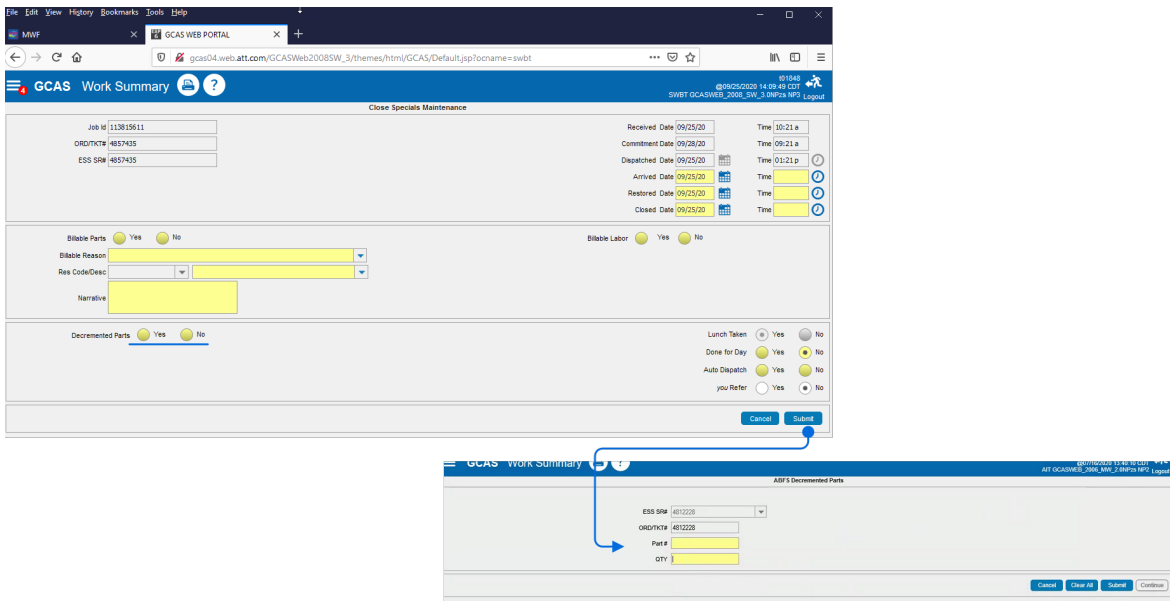
EXISTING TOOL

User Control & Freedom
The submission form for decrementing parts appeared only after closing the job & tech can only submit 1 part at a time, without being able to remove or edit it.

Error Prevention
There was no confirmation screen for submission or validation messaging.

Recognition Rather Than Recall
No available record of previously submitted parts, tech's either need to keep everything in mind or contact their manager to find out.

Help Users With Errors
There was no way to fix errors inside of the tool, technician had to contact the dispatcher in order to get help and solve it.

Help and Documentation
Prior to submitting parts techs should know their inventory on the truck, there's no way to check if inventory in the system accurately mirrors their actual on-hand inventory or not. And there's no upfront information for the dispatcher, who could help with sorting out some of the errors.

Design Process Flow

Feature : 112643/131403 MVP 2 – UI Job Details Enhancement to link Managed Data with Telco Dispatch data - wireframe & template (ACN)

Step 1. Get ACN Notification

Today's Work List

Tech will get the notification seven days a head to verify customer appointment.

ACN Notification → Future Worklist

Select ACN Job card → Job Tab → Customer Tab

Customer Contact → Verified appointment

UX recommended approch for better user experience

UX : Blue dash line indicates UX Recommended approach for ACN notification but due to some technical constraints this solution is not very much feasible in terms of backend and architecture perspective.

Step 2 : After successfully confirming the customer appointment how the tech would confirm the action in ATLAS?

Here would again go back to the customer tab where we incorporated toggle functionality which captures the tech's response. Based on that response backed API would dismiss the notification from the worklist permanently.

Step 2. Manual Confirming appointment

Manual ACN Confirmation

Customer Tab → Mark Completed

Backend process
API Identified ACN → Dismiss ACN Notification

UX : What if tech would accidentally hit "X" icon in that case tech would lose the notification the only way he can recovered the notification when the tech would Log out and log back in the application. (i.e)Session based notification.

Hit " X " icon for dismiss the Notification (Temporary)

Today's Work List → Hit "X" → Dismiss ACN Notifiaction → Refresh page

Notification will not appear again.

Initial Design
Solution

First, I mapped out the **experience of confirm customer availability prior to visit customer location.**

This approach is **different from how techs using it now,** but it would be **more consistent with other features,** that needed to be done prior to closing (like submitting billable items, youRefer credits, and others).

orova
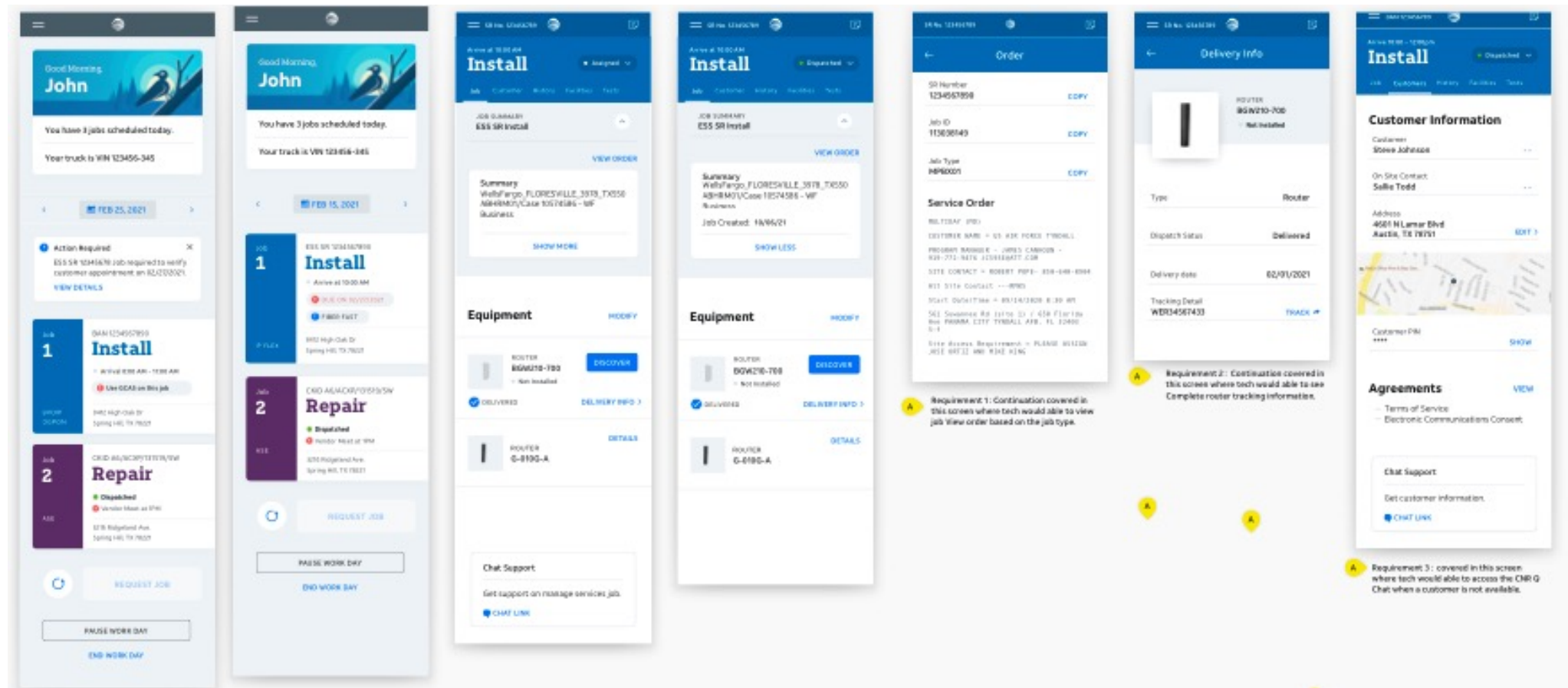
Next, I created **another version** to see **how would (ACN feature) works**, if it was **submitted after closing the job**.

We had doubts if our initial recommendation would work for techs' needs & we could predict **potential API limitations**, so wanted to **test with techs** first, to understand if we should push on our first solution.
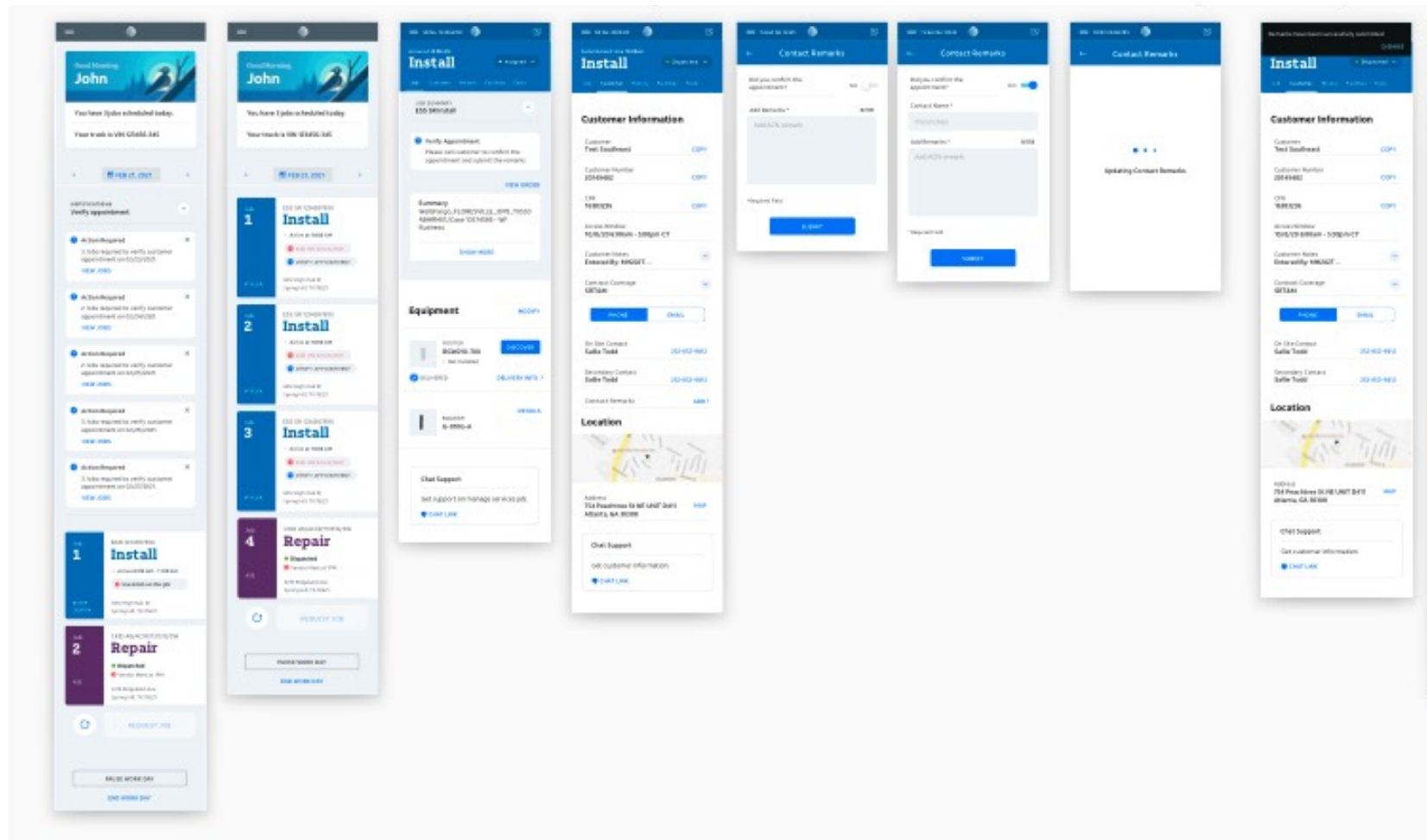
Conducting first round of **usability tests** helped us **learn**, that:

- Techs don't have preference over submission **before** or **after** closing the job. So we have decided to go with consistency and kept the first flow.

- Techs **love** the idea of **submitting multiple parts** at a time & **having** some sort of **record of already submitted parts**.

- Techs **wish** they could **have** a similar **record for standalone tool** as well, however, it's not their priority, as they **would not use** standalone tool **as often**.

There were couple additional flows in between, where I was collaborating with developers, however, for testing purposes we have landed only on 2 flows from previous slides.

After some additional **iteration on visual design**, and using feedback from the testing, I have come up with the **final flow** for **the Close Job** scenario & a **prototype** that we presented to 3 techs.
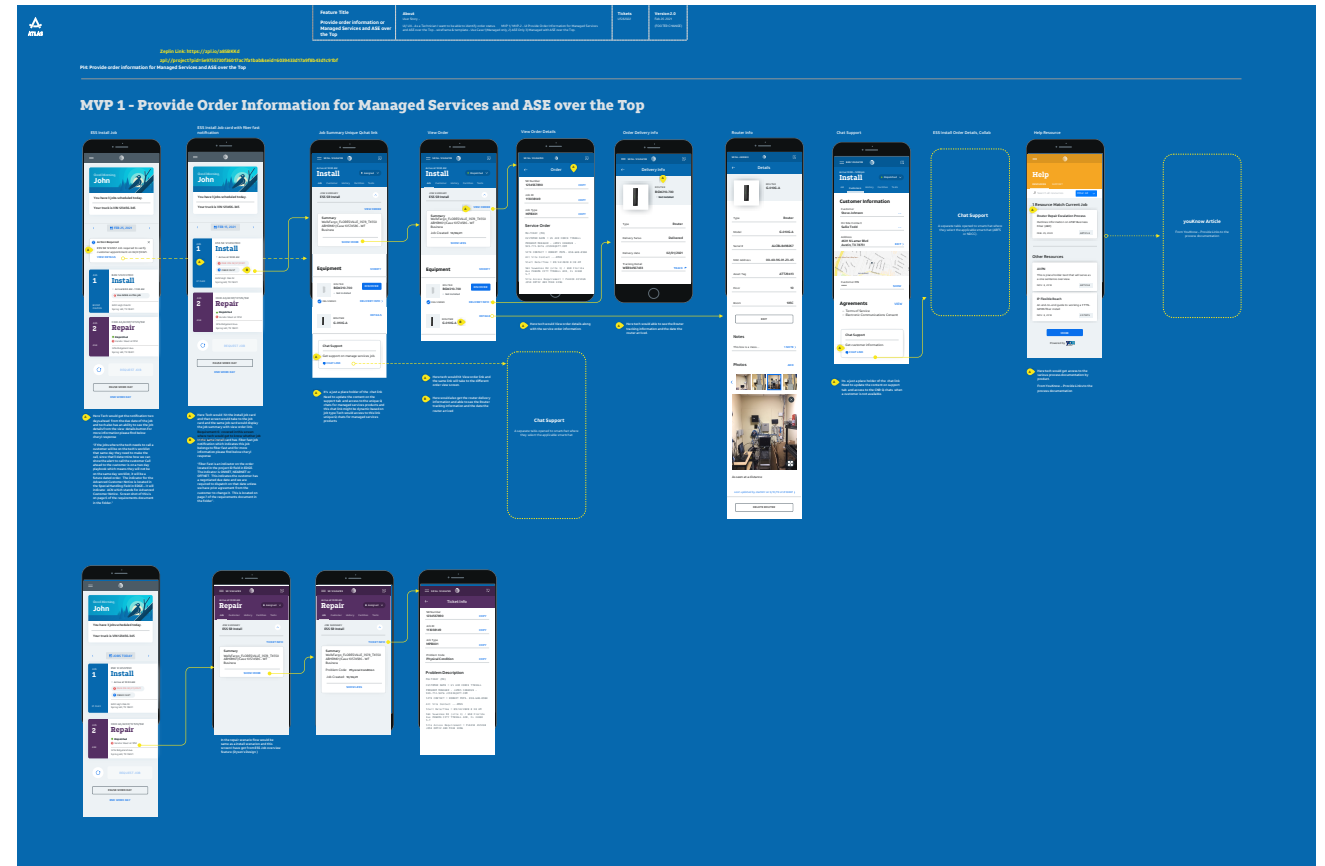
View Prototype

I have also created a **flow** for **standalone tool**, so we could finalize our Usability Tests & **prepare for the final handoff**.

View Prototype
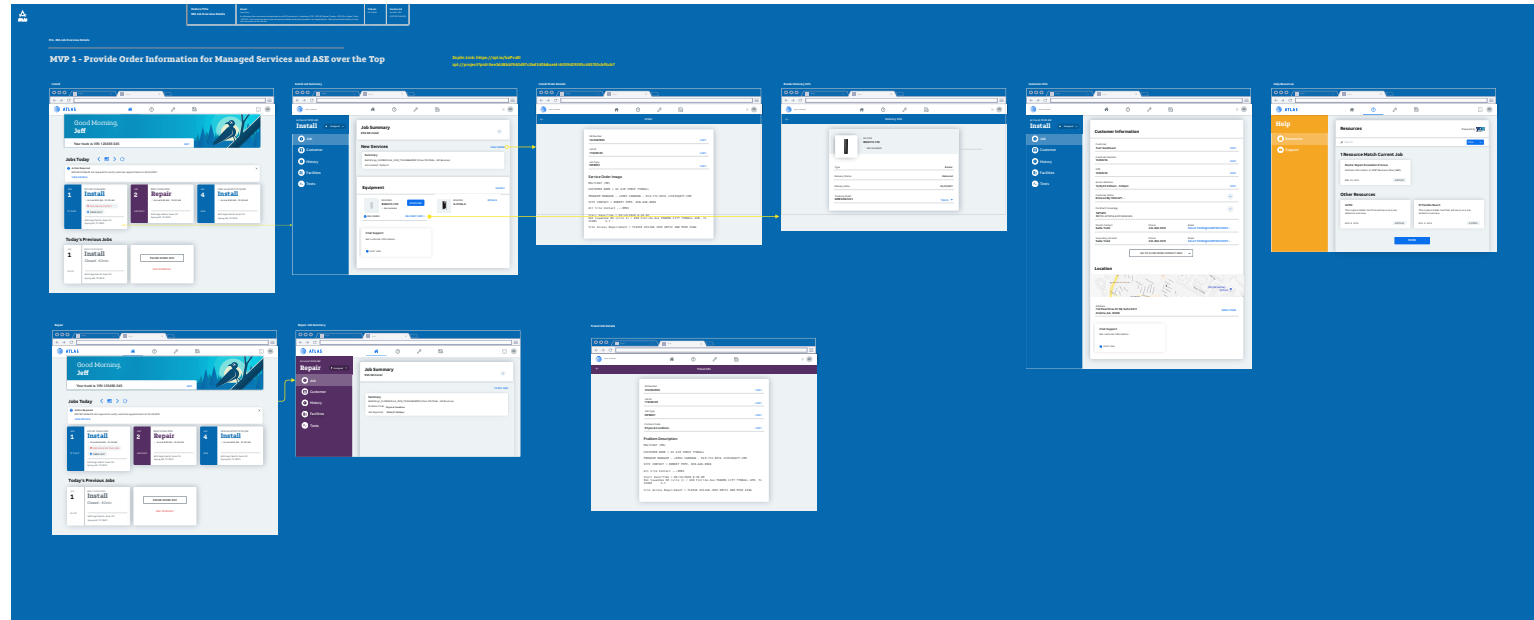
While handing off the feature I have provided my recommendations for future improvement. Such as **providing a way** for techs to r**eview their past submissions** and potentially be able to **add more parts** to them, that will require API upgrade.

Closing a job is an **integral part** of a technician's work, **being able to do it efficiently** & **without** any **pain points** will **reduce amount of friction** they have throughout the day.

Creating a more **optimized way to submitting decremented parts**, while closing the job, will help tech to save some time in their day & will **let them focus on other, & more important tasks**.

Key Outcomes & Results

- Because of close collaboration with back-end team, we were able to create a better and more efficient solution for techs.

- All interviewed technicians (8 from usability tests) were happy to see the improvements we made from the initial solution & were looking forward to the implementation of this feature into ATLAS.

What I Learned

- Collaboration and communication between different teams is the key to a successful result.

- Heuristics evaluation of the problem or already existing solution is highly valuable in the early stages of the work, as it allows to better focus on certain aspects, that could've been overlooked otherwise.

Audit, Design & Maintenance Work for ATLAS Design System

Company: AT&T   **User:** Designers & UI Developers

Project Summary

I worked with Sr. Designer and on my own to audit, create, maintain components & outline patterns for the existing design library, & collaborated with the Front-End team to create consistency & order within the Design System.

In addition, I also was just a team "librarian" & was giving suggestions on some patterns that would be more consistent or appropriate for specific needs.

Responsibilities

- Visual Design
- Experience Design
- Content Governance

Timeline & Team

There was no specific timeline, as we didn't have the budget and time reserved for this, it was an ongoing project in my free time.

Team: UX Designer, Sr. Visual Designer, UX Team Lead, 2 UI Developers

When I just joined the team, one of my first tasks was to **make an audit for the design system**, to **find and to notate inconsistencies between Design and Development libraries**. Later that became an **ongoing effort** on our path to creating a highly functional and consistent system.

Date	Issue (describe the issue that you came across)	Design Preview Screenshot (provide a screenshot of how the screen/component looks in design)	Dev Preview Screenshot (provide a screenshot of how the screen/component looks in development)	Notes (provide additional notes, if you have any)
1/26/2021	Create Ticket (desktop) Design uses modal with Line items and Headline. Dev replicates mobile view.			Design approach is also inconsistent between Desktop and Mobile screens
1/26/2021	Tests Tab (inconsistencies in color and text)			For Design this was the latest desicion designed for UTG Testing. Original design was similar but without chevron icons (Test tab previously designed by J

Dev component library			Design component library			
Dev CL Details	Dev CL Screenshot	Dev CL Name & (token name)	Design CL Name & ($Proposed token)	Design CL Screenshot	Design CL Details	Additional notes
		Icons	*Alert*			
Use this icon in an alert when the alert is only informational		Info <atlas-alert-info>	Low		Blue Exclamation: Used for informational alerts	**Dev vs Design:** - Dev size - 25.61x24.8 | Design size: 24x24 - Icon itself looks slightly different in Dev vs Design (Design -
Use this icon in an alert to provide a warning to the technician		Warning <atlas-alert-warning>	Med		Yellow Exclamation: Used for important warnings	
Use this icon in an alert to inform the technician of a dangerous situation		Danger <atlas-alert-danger>	High		Red Exclamation: Used for critical problems	
Use a brighter red background		Brighter Red Danger (same token as above)	~	~	~	DEV issue: - Danger & Brighter Red Danger have very similar colors, what is the need to have them separate? (what each of them are covering and can we combine them OR change colors?) - Shares same token name as Danger
Use this icon in an alert when providing a success state of an		Success <atlas-alert-success>	Check		Blue Check: Used for confirmation that something	
Use a green background		Green Success (same token as above)	~	~	~	**DEV issue:** - Success Green does not exist in Design CL, what is the purpose of it? Should we remove it? - Shares same token name with Blue Check
Use this icon in an alert to convey that an operation in the context of the alert is currently being		Loading <atlas-alert-loading>	Working Mixins/Icons/One-dot Loader/Working	~ (couldn't find details)	**DEV & DESIGN Issue:** - At a largest state loading icon is smaller than any other Alert Icon, which probably is inconsistent
~	~	~	Working Mixins/Icons/One-dot Loader/Working		~ (couldn't find details)	

While doing the audit, I also came across some **ADA compliance issues**, specifically within the color for the overall Help section.

I have brought this observation to my Team Lead and to our bigger group of designers and developers. However, it was decided that, because **we didn't have time allocated** for fixing this issue, we potentially **should come back to it at a later time**.
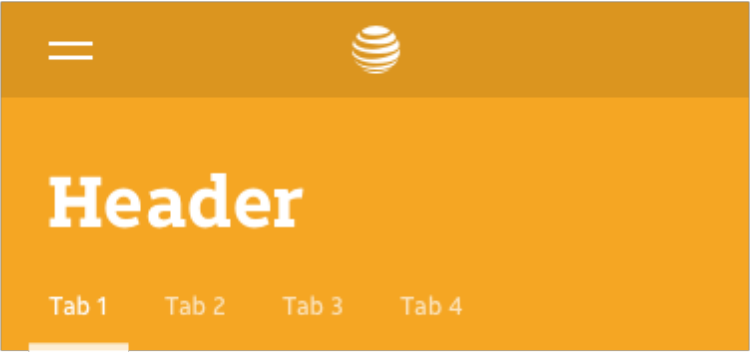
HELP HEADER

HELP NOTIFICATIONS

The contrast for help color fails on all WCAG parameters & is especially hard to read for inactive tabs, which already have 70% opacity.

DS AUDIT & FINDINGS

Another finding that we came across - this system **wasn't using any existing DS as a guide** but was **borrowing** different c**omponents from multiple systems**, without any particular reasoning behind it.

That created multiple **inconsistencies** in patterns and **confusion** for designers and front-end developers.

One of the examples of these inconsistencies was having **2 different grid layouts** for desktop pages. 1 was used for main job pages, with side navigation, & another for child pages, without side nav.

The **number of columns** &**different widths** created a lot of confusion among designers, and basing on my observations, I doubt the front-end development team even followed those grids, which later created a lot of **misalignment between our teams**.

DESIGN SYSTEM'S GRID LAYOUTS
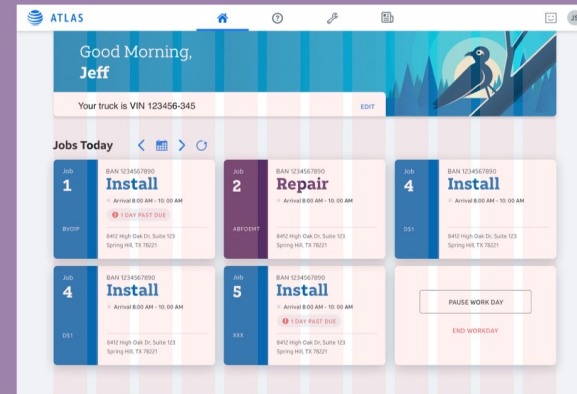
CENTER ALIGNED GRID FOR CHILD PAGES
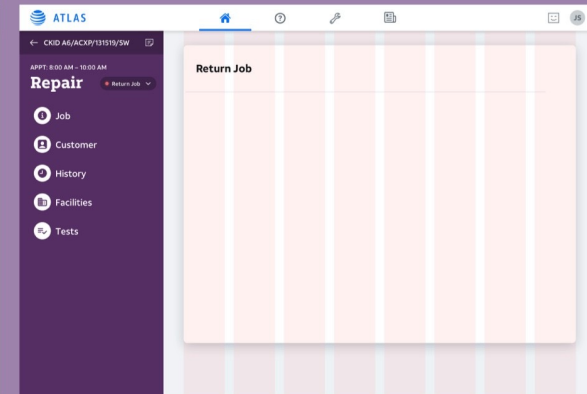
RIGHT ALIGNED GRID FOR MAIN PAGES
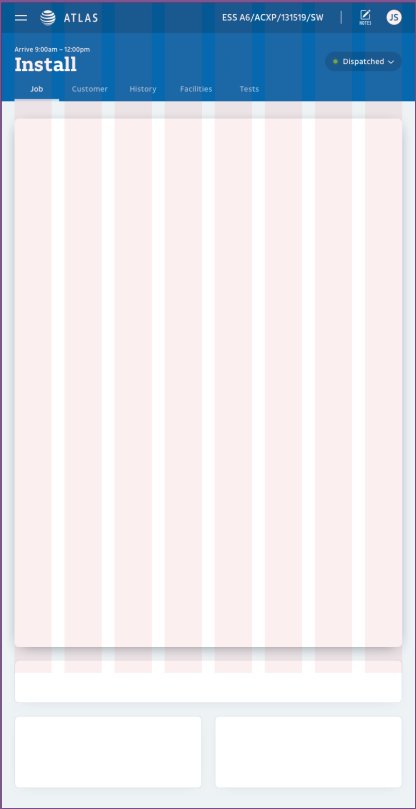
GRIDS IN PRODUCTION
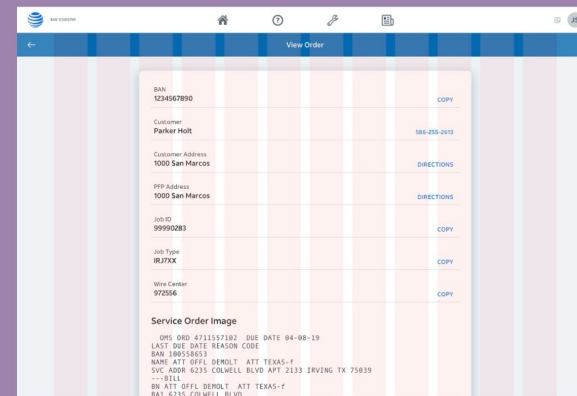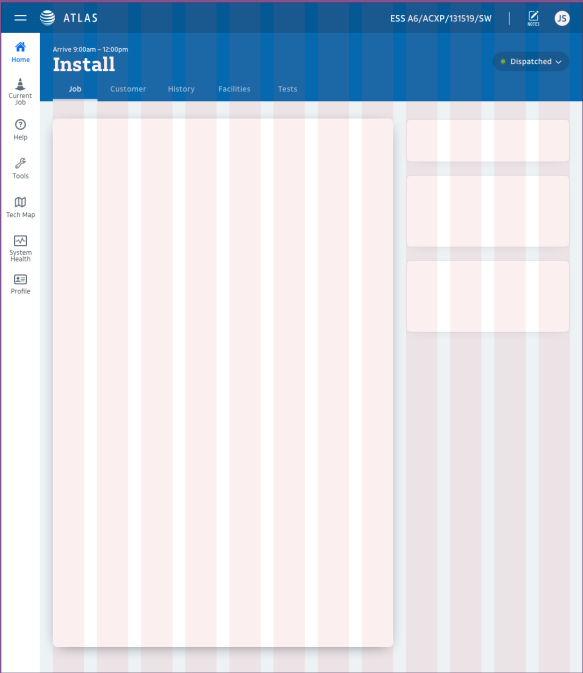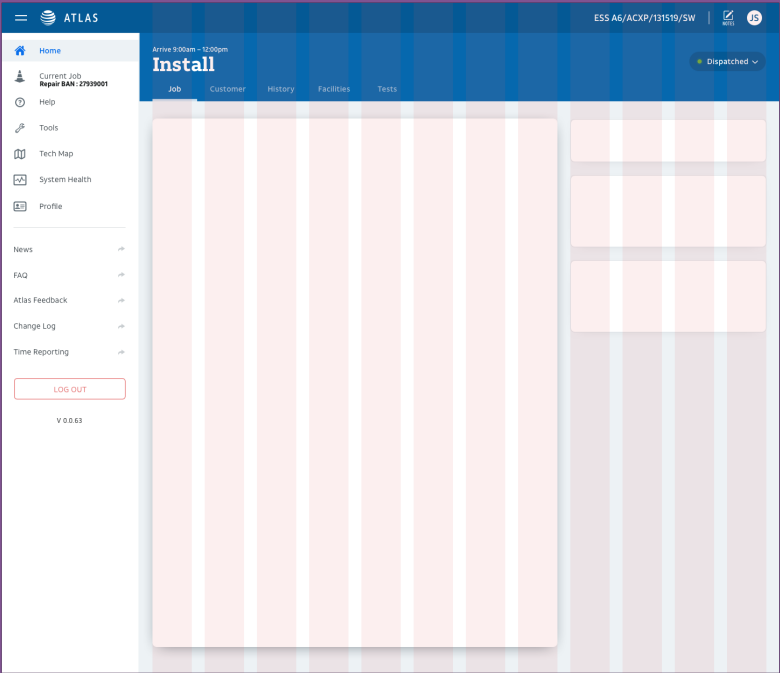

CHILD PAGE BASING ON DESIGN LIBRARY
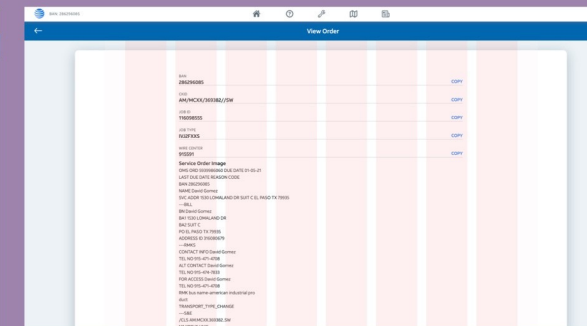
SAME CHILD PAGE EXISTING IN PRODUCTION

NEW GRID PROPOSAL
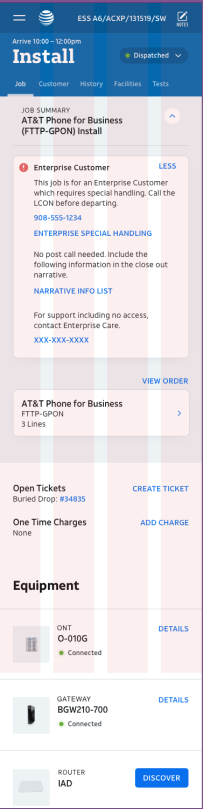
	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	Extra large ≥1200px
.container	100%	540px	720px	960px	1140px
.container-sm	100%	540px	720px	960px	1140px
.container-md	100%	100%	720px	960px	1140px
.container-lg	100%	100%	100%	960px	1140px
.container-xl	100%	100%	100%	100%	1140px
.container-fluid	100%	100%	100%	100%	100%

Basing on the learnings from DS Audit, we have decided to propose a new grid layout. ATLAS uses bootstrap at it's core, so I proposed responsive layout basing on the standard bootstrap grid.

One of my efforts while working on DS for ATLAS, was collaborating with Front- End Developers on creating new components & cleaning up issues in the existing ones.
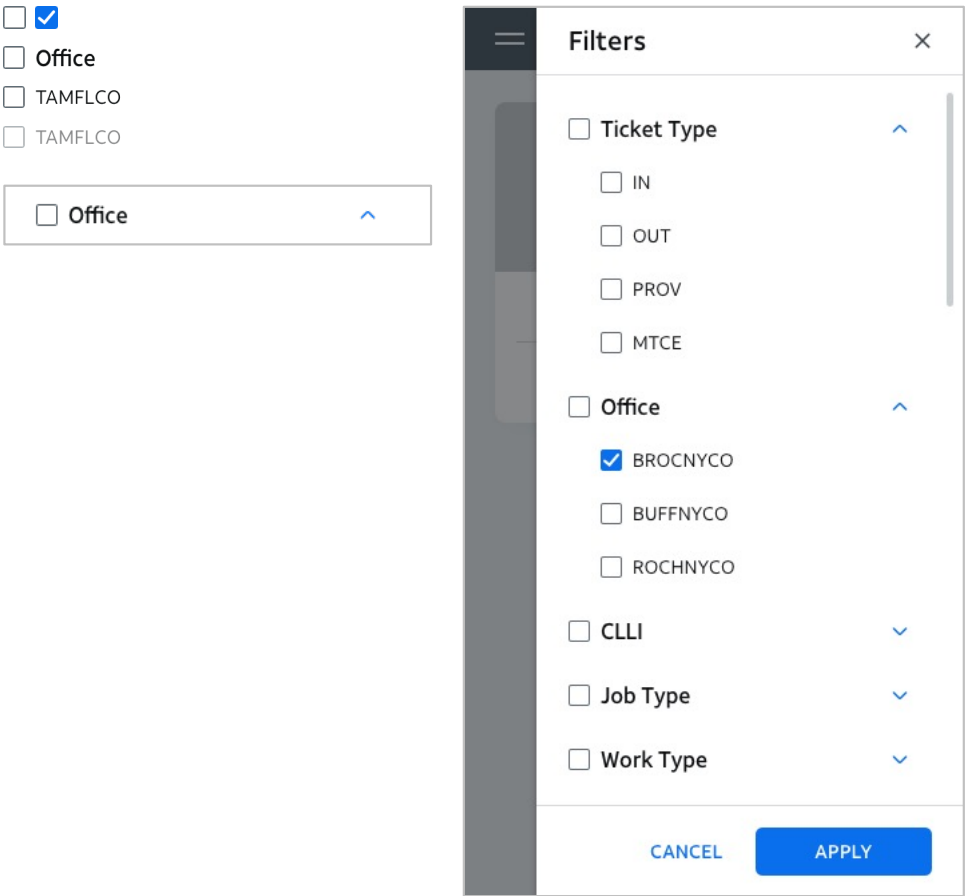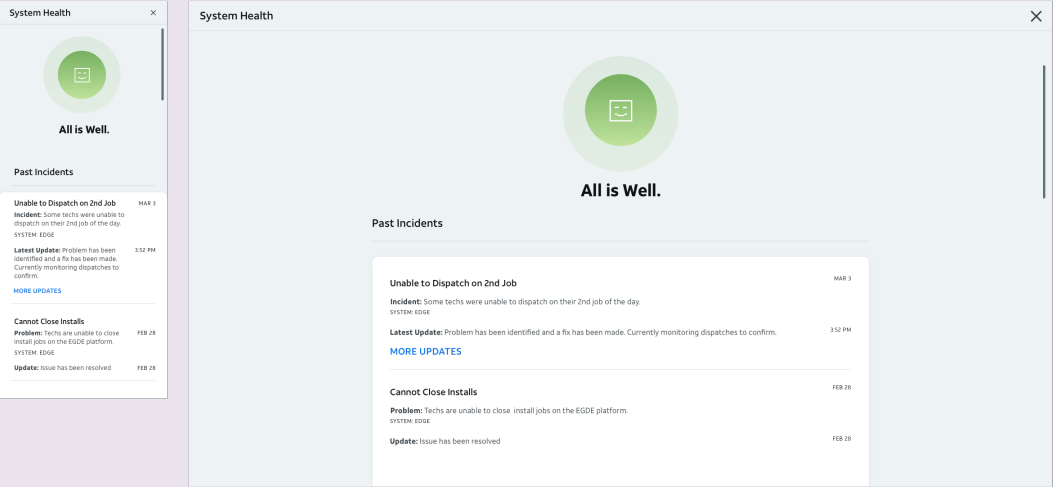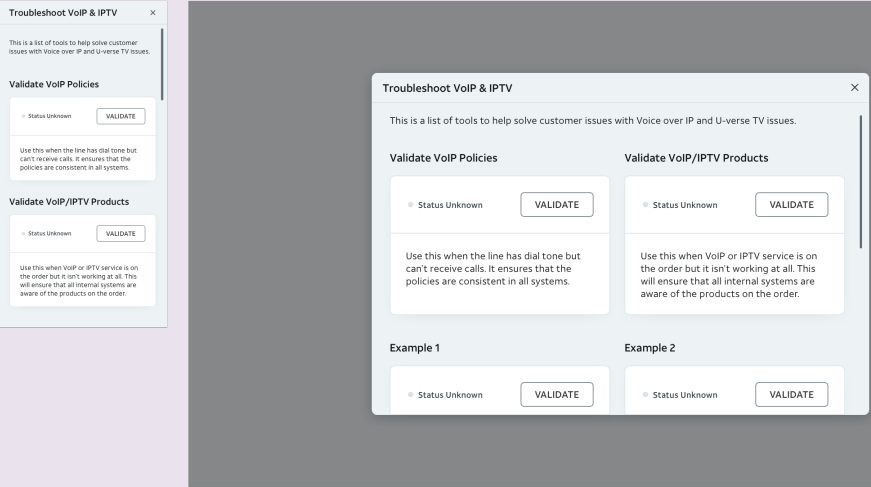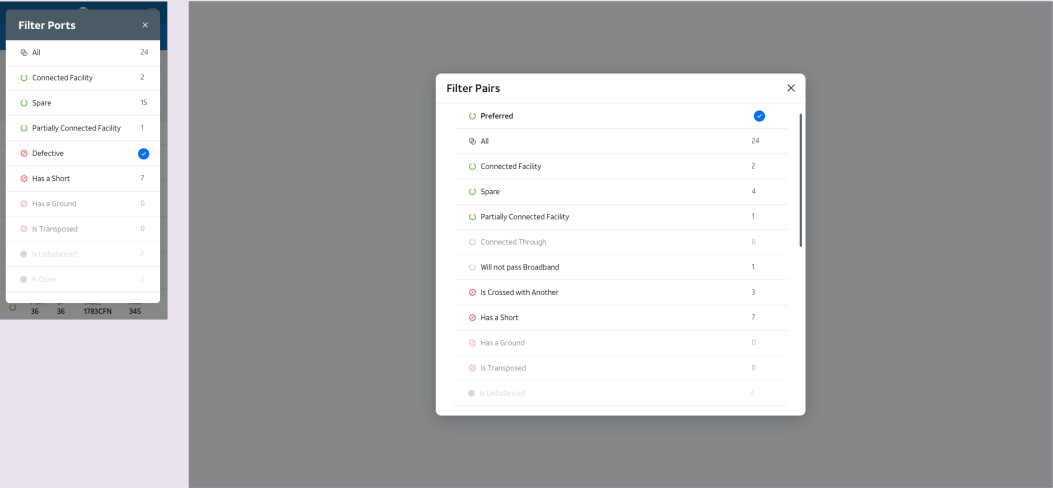
- Side-sheet

- Tooltip
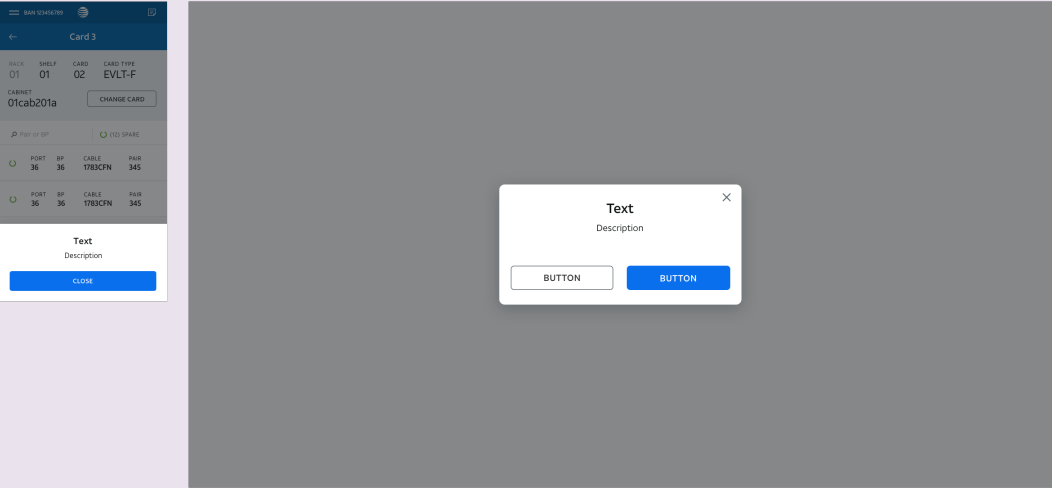
- Bottom action-sheet

NEW COMPONENTS FOR DS - MODALS

- Full-screen

- 8-column

- 6-column

- 4-column

Thank You!

Download Resume

E-mail-maciupui6@gmail.com   Cellular-765-398-2878