Medi-Pal

A8: Siddhartha J, Douglas K, Nash Y

Team Name and Value Proposition

Medi-Pal

Discover. Heal. Thrive.

Why Medi-Pal?

- Conveys a sense of friendliness and companionship
- Designed to be a ‘pal’ that guides through complexities of Medi-Cal: an always-available assistant that feels approachable

Why this value proposition?

- Helps users *find*, *truly understand*, and *leverage* Medi-Cal resources to improve health and well-being
- Unlike catalogs or resource databases, Medi-Pal takes a *proactive* approach in users’ health

Our Team

Siddhartha J

Computer Science + Design

Douglas K

Computer Science

Nash Y

Biomedical Computation

Problem & Solution

Problem

- Low awareness, confusion, and inaction due to complex Medi-Cal information and lack of personalized guidance
- Millions of low-income Californians qualify for Medi-Cal benefits but don't know where to start or how to access/leverage them
- Existing directories are overwhelming, hard to act on, and unintuitive

Solution

- Platform that helps users instantly find nearby Medi-Cal resources with clear contact and access details
- Personalized recommendations and reminders based on each user's needs, eligibility, and location
- More targeted and proactive than existing platforms: Medi-Pal actively nudges users instead of waiting for them to search + leverages voice/avatar AI to provide a human-touch

Today's Agenda

01 Heuristic Evaluation Synthesis

02 Major UI Revisions

03 Current Progress on Prototype

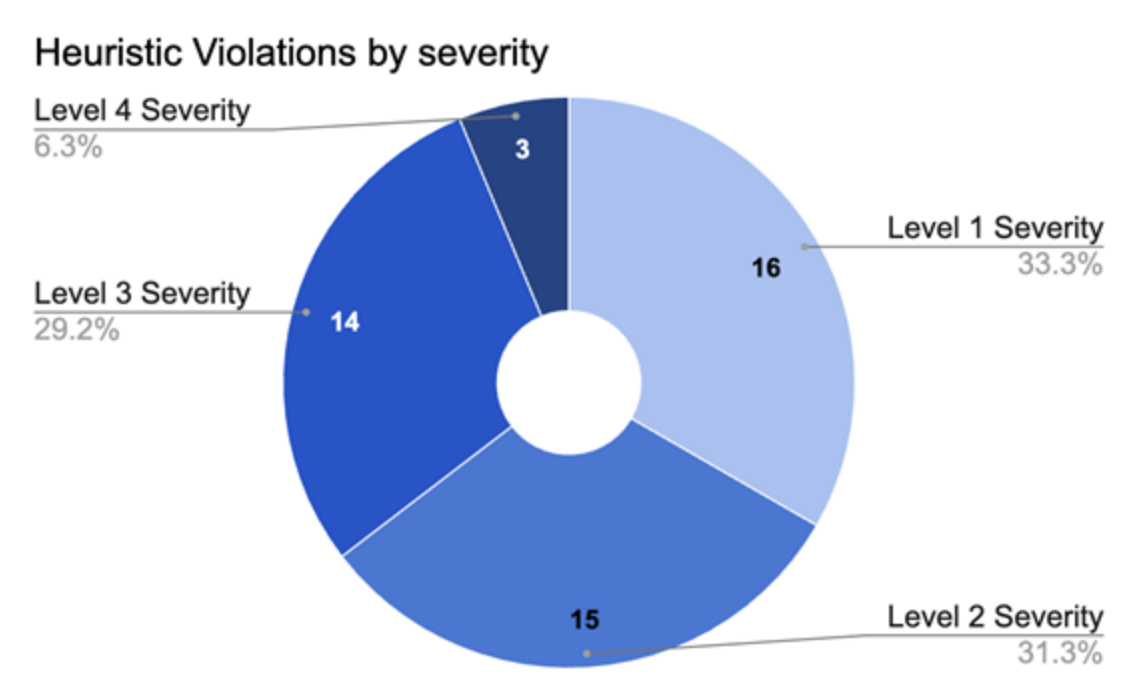
04 Live Demo!

Heuristic Evaluation Synthesis

01

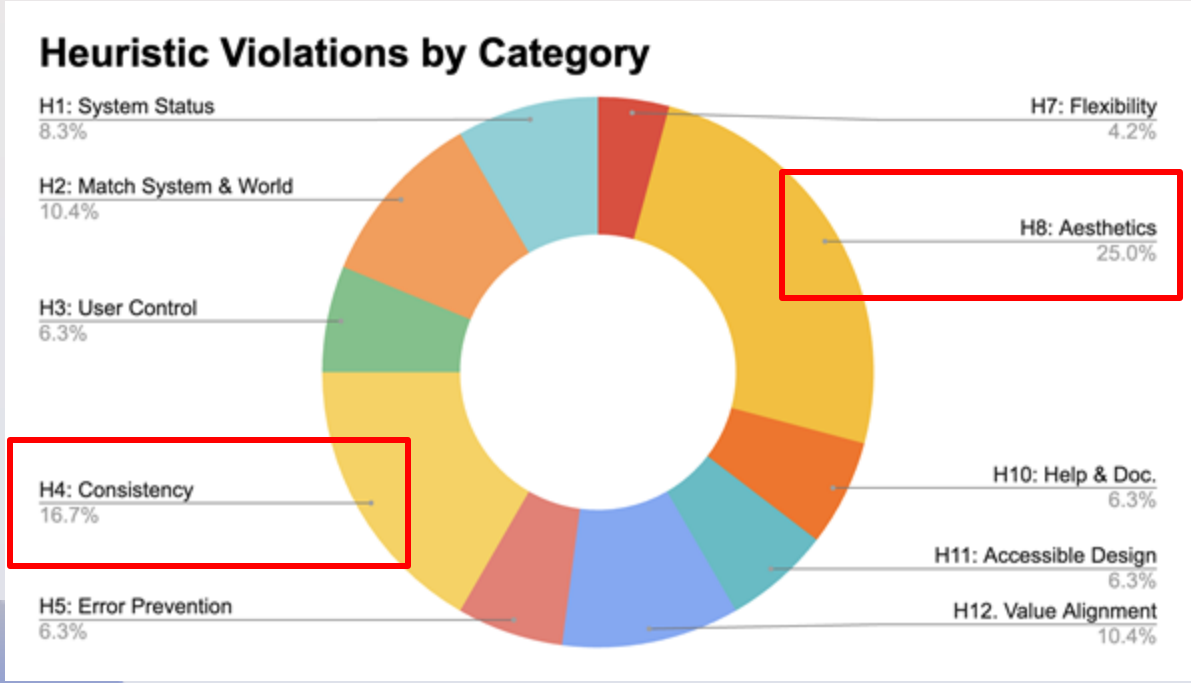
Violations: Grouped by Severity

48 violations. **17** severe violations.



Violations: Grouped by Category

48 violations. **17** severe violations.



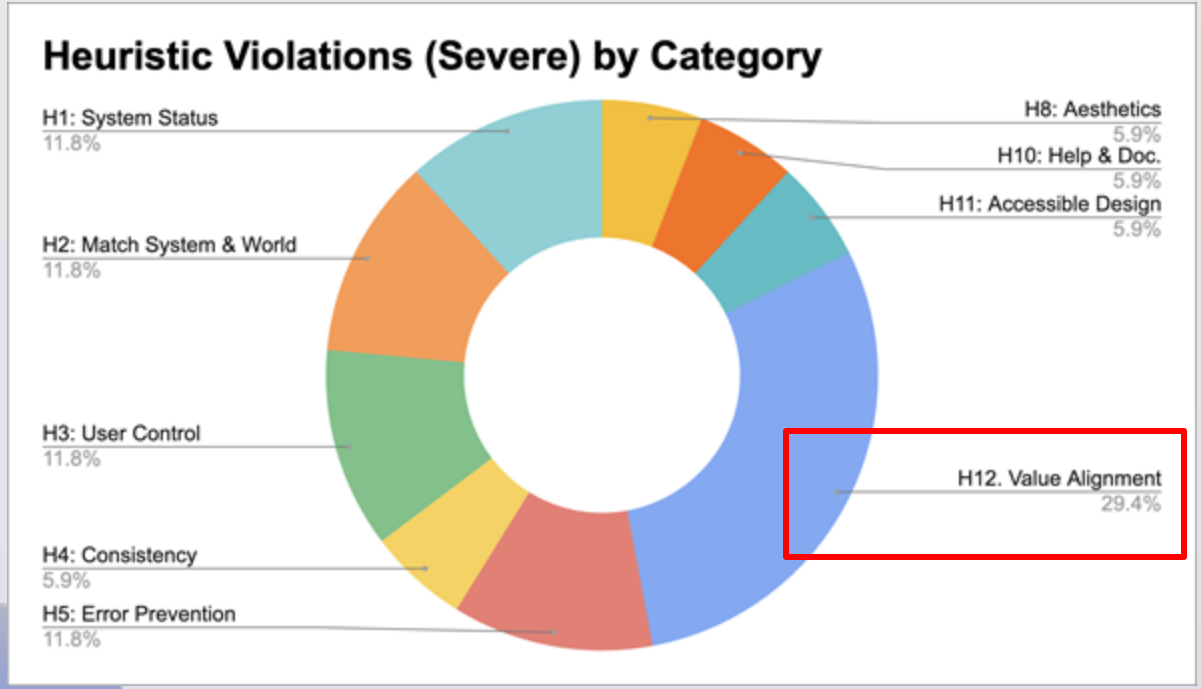
Most of our heuristic violations were about **inconsistent layouts** and **clutter**.
Here are some examples:

4 buttons with same purpose

Inconsistent search bar

Violations: Grouped by Category

48 violations. **17** severe violations.



A considerable proportion of severe violations were due to **lack of consideration** of marginalized groups:

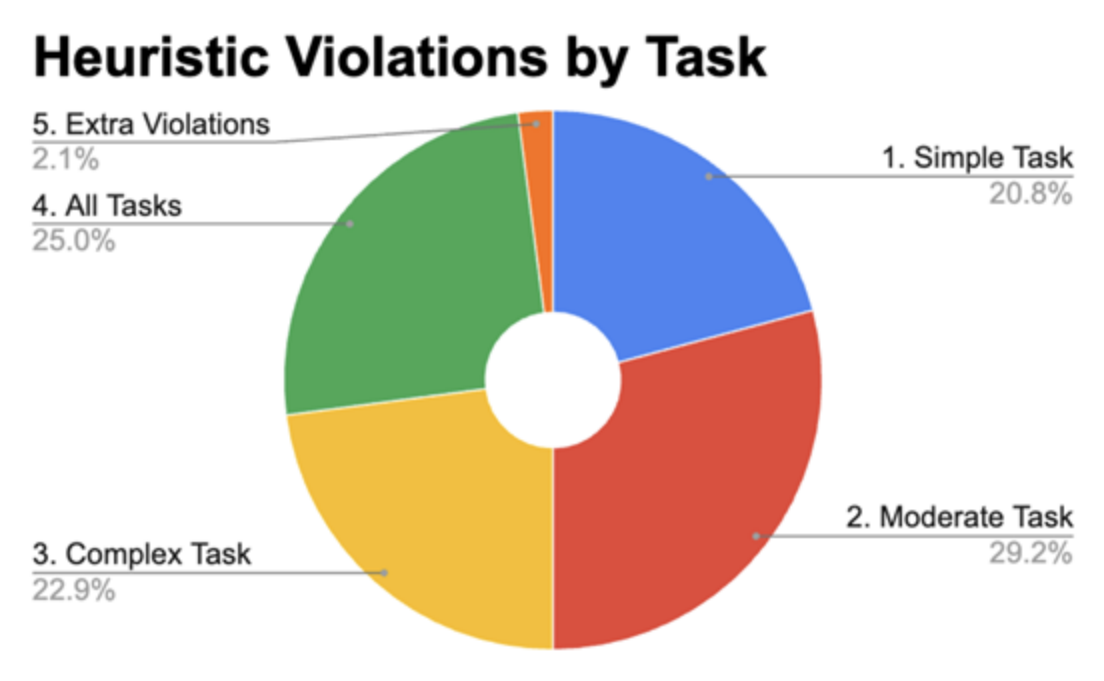
Lack of multi-language support

Inaccessible for users living in rural areas

Intimidating use of specialized language

Violations: Grouped by Task

48 violations. **17** severe violations.

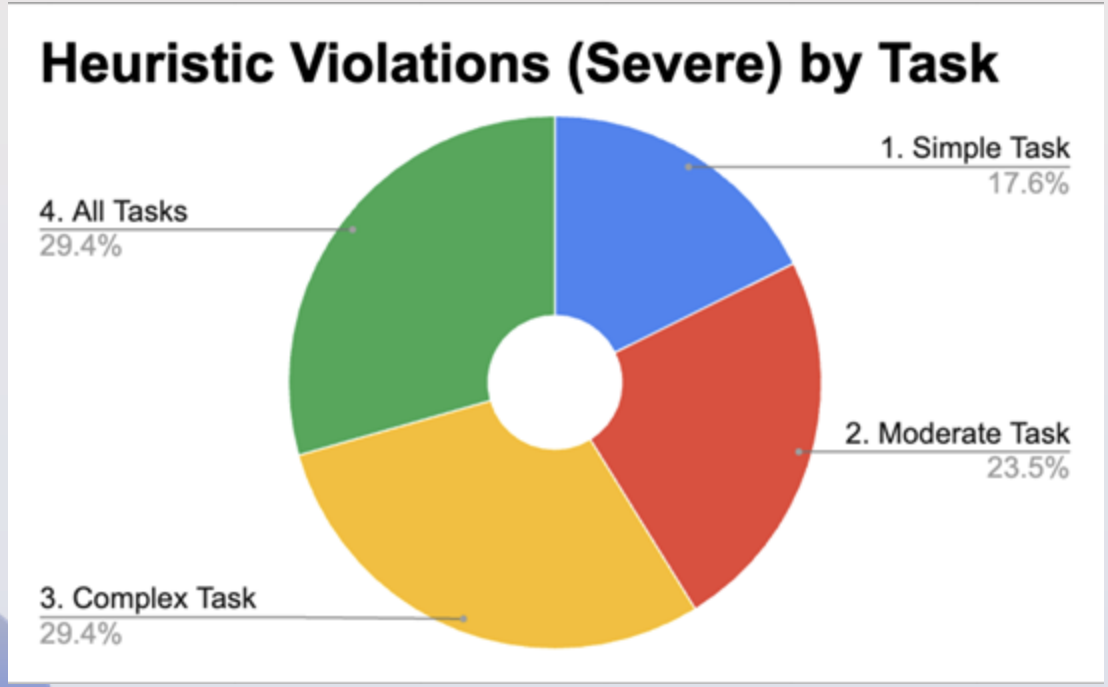


Violations were distributed evenly across different tasks, and **many (25%)** were **recurring across all tasks**. Here are some interesting observations:

- Most **aesthetics-related** violations occurred in our **Resource Page** for Task 2 (50% of all H8 violations).
- **Inconsistent design of chat interface** was the most frequent violation in Task 3 (27% of all Task 3 violations)
- **Value alignment** was a **recurring issue** across ALL tasks (25% of all “All Tasks” violations).

Violations: Grouped by Task

48 violations. **17** severe violations.



Severe violations were **focused especially towards Task 3** (29.4%), and there were a **large number of repeated violations** (29.4%) across all tasks.

For Task 3, the most prevalent severe violation was the **lack of user control (H3)** in the chat interface:

- Lack of support for finding additional resources.
- Some pages (i.e., Checklist) cannot be accessed before chatting with AI.

Violations: Insights for Fixes

Problem	Fix
Clutter and unneeded redundancy, especially in Resources Page	**Focus users' attention on important details** by including more white space
Inconsistent layout, especially for chat interface	**Enforce consistency** using Design Systems & Stylesheets
Chat interface **forces a linear navigation path** on users • We argue this is a **product critique,** not a heuristic violation - explanation provided in later slides.	Anticipate frequent user actions and **support multiple task flows**
Lack of consideration for marginalized groups	Use **simple vocabulary** throughout the app, provide **multi-language support** in high-fi prototype

Violations: Detailed Overview

Level 3 Severity
(Major Usability Problem)

Heuristic Violations - to be addressed:

- Schedule/calendar is first read on Home Page (H1).
- "Learn More" pop-up does not help users understand & leverage their resource (H10).
- No back button after getting search results (H5).
- 4 buttons with same function on home page (H8).
- Minimal visual hierarchy in Checklist page (H1).
- "Checklist" is not the most intuitive word for the feature (H2).
- AI automatically edits the checklist (H5).
- Texts & links are in blue font (H11).
- Intimidating use of specialized language (H12).

Product Critiques - addressed in high-fi:

- Lack of consideration for people living in locations with sparse healthcare resources (H12).
- No chat history (H12).
- Checklist page cannot be accessed before chatting with AI (H3).
- Lack of multi-language support (H12).
- White male doctor avatar may reproduce existing stereotypes (H12).

Level 4 Severity
(Usability Catastrophe)

Heuristic Violations - to be addressed:

- No "Take Me There!" button for nearby resource after user search (H4).
- Clicking "X" button takes user to an irrelevant screen (H2).

Product Critiques - addressed in high-fi:

- Users cannot start a new chat to find an additional resource after completing their first conversation (H3).

Major UI Revisions

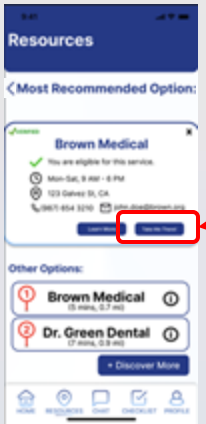
02

Revision #1: Enforcing Consistency

When searching in the search bar in the "Resources" page, there is no "Take Me There!" button similarly to the first pop-up. (H4: Consistency & Standards, Task 2)

Before

After



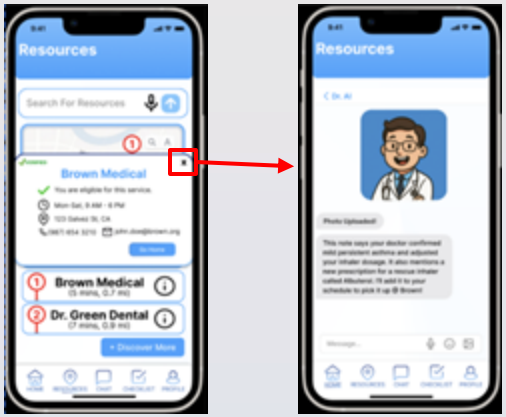
"Take Me There" button is added for the search result.

Justification: We have decided to use the same buttons used in our "Nearby Resource pop-up" to increase consistency. The "Take Me There" button has the same feature and brings users to a screen that shows step-by-step directions on how to reach the resource's location.
Progress on Usability Goals: Increases **usefulness** of our app, as users can use our app directly to navigate & get directions towards the desired resource, instead of searching on another maps app.

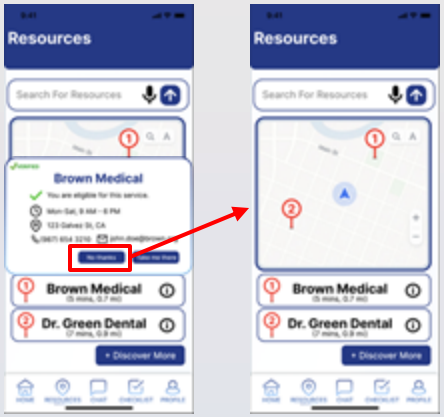
Revision #2: Wrong Hyperlink

When clicking the "X" button on the top right corner of the "Brown Medical, "YMCA Gym", and "Dr. Green Dental" pop-ups, it takes you to the middle of a conversation with Dr. AI. (H2: Match between System & World, All Tasks)

Before



After



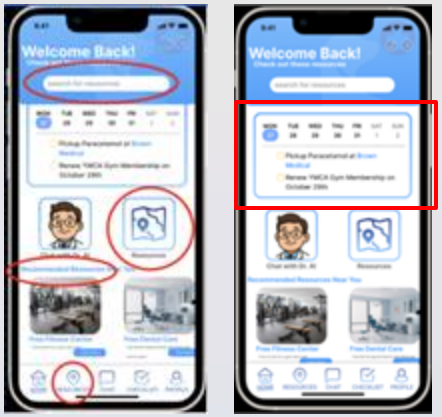
We fixed the hyperlink on Figma so users can return to the original page after existing.

Justification: This was a bug with our hyperlinks in the original medium-fi prototype, and we fixed this by changing the hyperlink to the correct page. We also changed the "X" button to the "No Thanks" button in our new design because some of our target users lack digital literacy and may not understand the "X" button on the top right corner.
Progress on Usability Goals: Increases **efficiency** and **learnability**. If the bug was still there, users would have to return to the home page and go back to the "Resources" page to truly exit the pop-up. Meanwhile, the "No Thanks" button uses text instead of icons as an affordance to help users without digital literacy understand how pop-ups work.

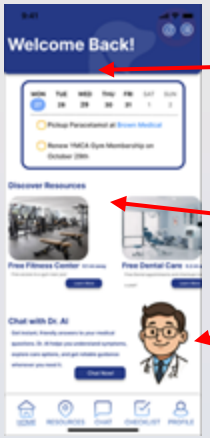
Revision #3: Enhancing Home Screen

On the "Home" page, there are 4 different features that reference resources. Similarly, there are 2 different ways to access the "Chat with Dr. AI" feature. (H8: Minimalist & Aesthetic Design, Task 2)
The schedule/calendar with daily tasks is the first-read on the Home page. (H1: Visibility of System Status, Task 1)

Before



After



Removed search bar

Removed buttons on home page and increased screen size for Resources page and chatting with AI.

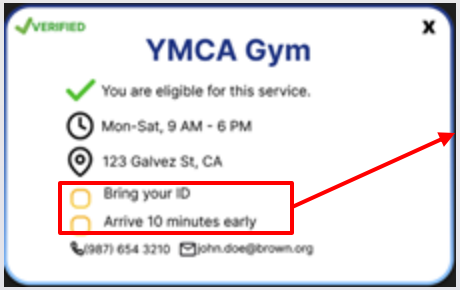
Justification: Our new design incorporates more whitespace and reduces clutter. Since we've removed the search bar and home buttons, there are now only two ways to access the "Resources" page. For each action, we let users have 2 different ways to access the page because we want to support flexibility. Regarding the "first read" issue, users were not able to notice features such as "Chat with AI" due to the small size of the home buttons. We fixed this by dedicating at least one-third of the screen to show each feature of "Discovering Resources" and "Chatting with AI". We believe that the user's first read should be their immediate action items of the day as it informs them about their plans, but we still draw user attention to other features by using large images of avatars and resources.
Progress on Usability Goals: Increases **learnability**. Users may be confused and would have to learn a lot of new controls if there are 4 different ways to navigate to the same page. So, we have decided to reduce the learning cost by reducing clutter on the home page. We have also added explanations of how "Chat with Dr AI" works, so users can understand what this feature does instead of guessing. Our consideration of first reads also act as visual cues for new users of our app.

Revision #4: “Learn More” Pop-Up

Your "Learn More" pop-up is limited to hours, address, and contact information. (H10: Help & Documentation, Simple Task)

Before

After



Added checklist items to help users prepare before navigating towards the resource.

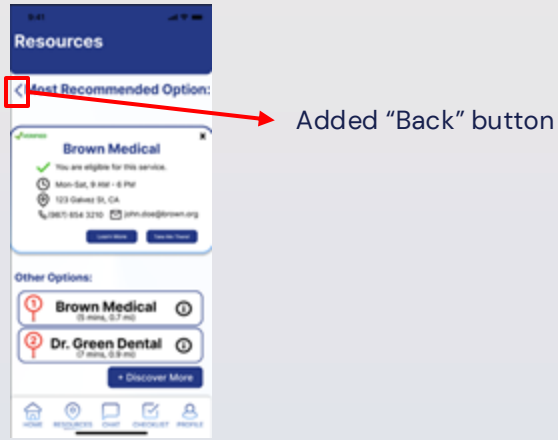
Justification: Most of the information in the original pop-up can be found on Google Maps, and does not provide actionable guidance for users to access the resource - they still do not know what they need to bring or what they need to sign up for. Having a checklist of preparation tips would provide users with more useful guidance on how to use the resource that is suggested.
Progress on Usability Goals: Increases **usefulness**. These tips are valuable for users as they may not know what to bring to their appointment, especially if they are accessing a new resource introduced by Medi-Pal. Having these tips would make sure that the users are actually able to benefit from the resources suggested by Medi-Pal.

Revision #5: Adding “Back” Button

After a user gets their search results, they can only move forward (reading more about other options). There is no back button/clear way to return to previous page. (H5 Error Prevention, Moderate Task)

Before

After



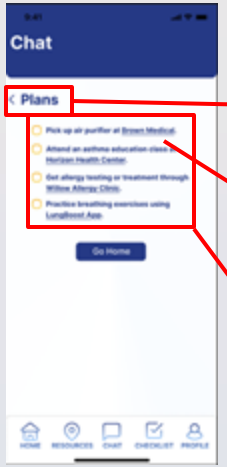
Justification: The back button allows users to return to the previous page if they have made a mistake about their search. This gives users a quick fix for any errors.
Progress on Usability Goals: Increases **efficiency**, as users can quickly recover from a wrong search query via the “Back” button.

Revision #6: Checklist Page

There is minimal visual hierarchy on the "Checklist" page. (H1: Visibility of System Status, Complex Task)
"Checklist" is not the most intuitive terminology for this feature. (H2: Match between System & World, Complex Task)
Language like "Pick up Medi-Cal approved HEPA air purifier at Brown Medical" is intimidating. (H12: Value Alignment & Inclusion, All Tasks)

Before

After



Changed from "Checklist" to "Chat">"Plans"

Locations are underlined.

Vocabulary is more accessible & actionable: perform "Action X" at "Location Y"

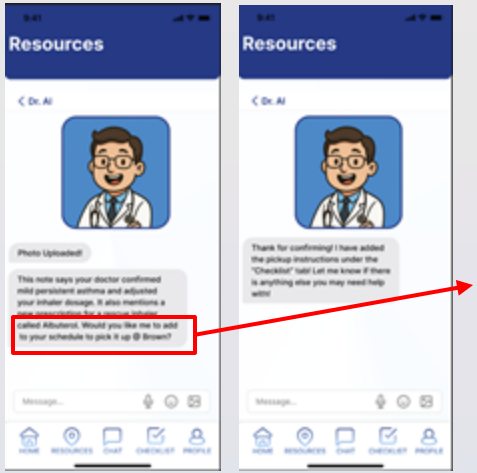
Justification: We have added visual hierarchy by underlining all locations in the checklist - this is an affordance for users to click and view the corresponding resource. We have also made the wording of the checklist items more accessible by avoiding specialized vocabulary, and reworded them in a way that provides users with simple instructions. We have also changed "Checklist" to "Plans" and made them local within the chat.
Progress on Usability Goals: Increases **learnability**, as users can now more easily understand where to click, what each item means, and how to navigate through resources without prior knowledge of the app's structure.

Revision #7: AI automatically edits checklist

Dr. AI automatically edits my schedule/checklist during our conversations. (H5: Error Prevention, Complex Task)

Before

After



AI asks for confirmation before making changes

Justification: In the original flow, AI may make mistakes when it edits the user's schedule. By letting AI ask for confirmation before making changes, users can decide whether they want AI to change their schedule, thereby reducing errors.

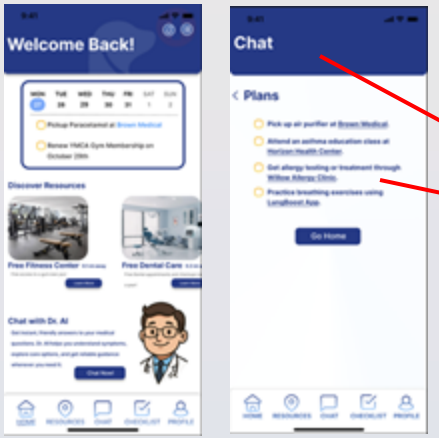
Progress on Usability Goals: Increases **usefulness**. If AI always make mistakes that cannot be prevented, then users would not find our chat feature useful due to increased error rate.

Revision #8: Changes in color

Some of the text and links are in blue colored font. As mentioned in class, having blue font is not the best design choice as it is hard to read. (H11: Accessible Design, All Tasks)

Before

After



Theme is changed to dark blue. Text is made darker to increase contrast with background.

Justification: We have increased contrast between the text and the background to make it easier for users to read the text. We have changed the color scheme of our app from light blue to dark blue to ensure visual consistency.

Progress on Usability Goals: Increases **efficiency**. Users will be able to perform actions quicker and read texts more efficiently if it is more legible for them.

Violations that are product critiques

- **Users cannot start a new chat to find an additional resource after completing their first conversation (H3).**
  - Our task flow only involves users finding one resource. Finding additional resources is an extra feature not part of the original task flow, and will be included in the final prototype.
- **Lack of consideration for people living in locations with sparse healthcare resources (H12).**
  - People living in locations with sparse healthcare resources will still see the nearest resources.
- **No chat history (H12).**
  - This is an advanced feature that will be implemented in the final prototype.
- **Checklist page cannot be accessed before chatting with AI (H3).**
  - The checklist page is not part of our task flows, so it is not accessible in our med-fi prototype. It will be implemented in our high-fi prototype.
- **Lack of multi-language support (H12).**
  - It is quite impractical to translate all our app contents to different languages for the medium fi prototype. We will provide multi-language support in the final prototype.
- **White male doctor avatar may reproduce existing stereotypes (H12).**
  - We will let users customize their avatar in the final prototype.

Current Progress on Prototype

03

Recap: Our Tasks

Discover nearby resources

Proactively find specific resource

Chosen for today's checkpoint.

Make plans for healthy habits

Core Frameworks

React Native (ver 0.73.6)

Cross-platform mobile framework enabling iOS and Android development from a single codebase

Expo (ver 50.0.0)

Development platform providing build tools, over-the-air updates, and native module access

TypeScript (ver 5.1.3)

Type-safe language layer ensuring code reliability and developer productivity

Navigation and Backend

Expo Router (ver 3.4.0)

File-based **routing system** implementing authentication flows and tab-based navigation

Supabase

Backend-as-a-Service platform handling user authentication, session management, and database operations

AsyncStorage

Local persistence layer for authentication tokens and user preferences

Maps & Location Services

Needed for implementing **Task 1: Discover Nearby Resources.**

Libraries

React Native Maps (ver 1.10.0)

Native map component library providing Google Maps integration.

Expo Location (ver 16.5.5)

Location services wrapper (currently using hardcoded Stanford location for prototype).

Google Maps APIs

Places API

Autocomplete search functionality for location discovery.

Directions API

Route calculation and turn-by-turn navigation.

Geocoding API

Address-to-coordinate conversion

AI Integration

Needed for implementing **Task 2: Proactively Find Specific Resource** and **Task 3: Make Plans for Healthy Habits.**

OpenAI API (ver 4.20.0)

GPT-powered **chat assistant** providing healthcare guidance and Q&A functionality

UI/UX Libraries

Expo Vector Icons

Ionicons **icon library** for consistent visual design

React Native Reanimated

High-performance **animation engine** for smooth UI transitions

R.N. Gesture Handler

Native **gesture recognition** for touch interactions

R.N. Safe Area Context

Safe area handling for modern device layouts

Usage of AI: Implementation

Code Generation and Architecture

- **Where:** Initial project setup, component structure, navigation setup.
- **Why:** AI was used to rapidly scaffold the React Native/Expo project structure, set up routing, and create boilerplate code for screens and components. This allowed for faster iteration and focus on feature implementation rather than setup.

Feature Implementation

- **Where:** Google Maps integration, directions API integration, autocomplete search functionality.
- **Why:** AI assisted in implementing complex API integrations (Google Maps Directions API, Places API) by generating the correct API call patterns as APIs have complex documentation and require specific formatting.

Usage of AI: Visual Design

Component Styling

- **Where:** All screen components, buttons, cards, modals
- **Why:** AI helped create consistent design patterns, color schemes, and spacing using the Theme constants. This ensured visual consistency across the app without requiring a dedicated designer.

Layout and Spacing

- **Where:** Safe area handling, responsive layouts, map container sizing
- **Why:** AI assisted in calculating proper spacing for different screen sizes and safe areas, ensuring the app looks good on various device sizes.

Implemented Features

1. Authentication System

- ★ **Sign Up Screen:** User registration with email/password
- ★ **Sign In Screen:** User login with email/password
- ★ **Privacy Policy Screen:** Legal document display
- ★ **Terms of Service Screen:** Legal document display
- ★ **Session Management:** Persistent login sessions via Supabase
- ★ **Protected Routes:** Automatic redirect to sign-in if not authenticated

Implemented Features

2. Home Dashboard

- ★ **Welcome Screen:** Personalized greeting with user's first name
- ★ **Calendar View:** Weekly calendar with date selection
- ★ **Checklist Preview:** Shows upcoming checklist items for selected date
- ★ **Quick Actions:** Navigation to other sections
- ★ **Resource Cards:** Preview of nearby healthcare resources

Implemented Features

3. Resources Screen

- ★ **Interactive Google Maps:** Full map integration with markers
- ★ **Nearby Resources Display:** Shows 4 hardcoded healthcare facilities
- ★ **Search Functionality:** Google Places Autocomplete API integration
- ★ **Directions System:** Route visualization and step-by-step instructions
- ★ **Map Features:** Scroll, pinch, and rotate enabled.

Implemented Features

3. Resources Screen (Details)

Search Functionality

- Real-time search suggestions as user types
- Dropdown with nearby locations
- Click-outside-to- close functionality

Directions System

- Turn-by-turn directions display
- Route visualization on map (polyline)
- Step-by-step instructions
- Distance and duration calculations
- "Start Route" functionality
- "Cancel Directions" and "I'm here!" buttons

Map Features

- Pinch-to-zoom enabled
- Scroll, pitch, and rotate enabled
- Custom Stanford location marker
- Destination markers
- Route polylines with darker blue when route started

Implemented Features

4. AI Chat Assistant

- ★ **Chat Interface:** Real-time messaging UI
- ★ **OpenAI Integration:** GPT-powered responses
- ★ **Message History:** Scrollable chat history
- ★ **Loading States:** Visual feedback during AI processing
- ★ **Healthcare Context:** Chat is contextually aware of healthcare needs

Implemented Features

5. Checklist Management

- ★ **Task List:** Display of healthcare-related tasks
- ★ **Date Filtering:** Filter tasks by selected date
- ★ **Completion Status:** Mark tasks as complete/incomplete
- ★ **Task Details:** View task information
- ★ **Modal Confirmation:** Confirmation dialog for task completion

Implemented Features

6. User Profile & Navigation System

User Profile

- ★ **Profile Screen:** User information display
- ★ **Account Management:** Basic profile settings

Navigation System

- ★ **Custom Tab Bar:** Bottom navigation with 5 tabs
- ★ **Tab Icons:** Visual indicators for each section
- ★ **Active State:** Highlighting for current tab
- ★ **Smooth Transitions:** Animated navigation between screens

Unimplemented Features & Plans to Finish

Total Timeline: 2 Weeks.

Week 1: Core Functionality Completion (Tasks ranked by priority)

Feature	1. Real Location Services	2. Real Healthcare Resources Database	3. Checklist Persistence	4. User Profile Editing
Current Status	Hardcoded for prototype	Currently using 4 hardcoded resources	Local state only	Display only
Plan	Implement GPS location tracking with proper permission handling	Integrate with healthcare provider APIs or build Supabase database	Supabase database integration for user-specific checklists	Add edit functionality and profile picture upload
Timeline	2-3 days (Douglas)	3-4 days (Sid)	2 days (Douglas)	1-2 days (Sid)
Depend-encies	Location permissions, device testing	API research, database schema design	Database schema, Supabase setup	None

Unimplemented Features & Plans to Finish

Total Timeline: 2 Weeks.

Week 2: Enhanced Features & Polish (Tasks ranked by priority)

Feature	5. Resource Filtering & Sorting	6. Favorites / Bookmarks	7. Notifications	8. Accessibility & Polish
Current Status	Basic display only	Not implemented.	Not implemented	Basic accessibility
Plan	Filter by type, distance, rating; sort by relevance	Save favorite healthcare resources to user profile	Push notifications for appointments and reminders	Screen reader support, error handling improvements, UI refinement
Timeline	2 days (Douglas)	1 day (Sid)	2 days (Sid)	2 days (Douglas)
Depend-encies	None	None	Expo notifications setup	None

Future Enhancements

Total Timeline: 2 Weeks.

Post-Week 2: Implemented if we have time

Appointment Scheduling

Integration with provider scheduling systems (requires API access)

Multi-language support

Spanish language support (requires translation)

Offline Mode

Resource caching and offline directions (requires caching strategy)

Wizard-Of-Oz Techniques

1. Hardcoded User Location (Stanford)

- ★ **Technique:** User location is always set to Stanford University coordinates (37.4275, -122.1695) regardless of actual device location
- ★ **Justification:**
  - Prototype testing requires consistent starting point
  - Avoids location permission complexity during development
  - Allows testing directions functionality without GPS access
  - Simplifies demo scenarios for stakeholders
- ★ **When to Replace:** Once location permissions are properly tested and GPS accuracy is

Wizard-Of-Oz Techniques

2. Dummy Healthcare Resources

★ **Technique:** Only 4 hardcoded healthcare facilities are displayed (Peninsula Healthcare, Ravenswood, Samaritan House, Pacific Free Clinic)

★ **Justification:**

- No healthcare provider database API available during prototype phase
- Allows testing of map markers and directions without external dependencies
- Provides realistic test data for UI/UX testing

★ **When to Replace:** When real healthcare provider API or database is integrated

Wizard-Of-Oz Techniques

3. Dummy Checklist Items

- ★ **Technique:** Checklist items are hardcoded in constants file, not persisted to database
- ★ **Justification:**
  - Rapid prototyping without database schema design
  - Allows testing of UI interactions immediately
  - Simplifies demo scenarios
- ★ **When to Replace:** When Supabase database schema is designed and checklist persistence is implemented

Wizard-Of-Oz Techniques

4. AI Chat Without Context Memory

- ★ **Technique:** Chat messages are not persisted, each conversation starts fresh
- ★ **Justification:**
  - Reduces API costs during development
  - Simplifies implementation for prototype
- ★ **When to Replace:** When chat history persistence is needed for production

Hard-Coded Aspects

1. Stanford University as User Location

- ★ **Location:** `app/(tabs)/resources.tsx` - `STANFORD_COORDS` constant
- ★ **Justification:**
  - Consistent testing environment
  - Avoids location permission issues during development
  - All directions calculations use Stanford as origin
- ★ **Replacement Plan:** Replace with actual GPS location once permissions are stable

Hard-Coded Aspects

2. Four Healthcare Resources

- ★ **Location:** `constants/DummyData.ts` - `dummyResources` array
- ★ **Justification:**
  - No real-time healthcare provider API available
  - Provides realistic test data
  - Allows testing of map markers and directions
- ★ **Replacement Plan:** Integrate with healthcare provider database or API

Hard-Coded Aspects

3. Checklist Items

- ★ **Location:** `constants/DummyData.ts` - `dummyChecklistItems` array
- ★ **Justification:**
  - Rapid prototyping without database setup
  - Allows immediate UI testing
- ★ **Replacement Plan:** Move to Supabase database with user-specific items

Hard-Coded Aspects

4. Theme Colors and Spacing

★ **Location:** `constants/Theme.ts`
★ **Justification:**
- Design consistency
- Centralized styling

★ **Note:** This is intentional design, not a limitation

Hard-Coded Aspects

5. Google Maps API Key

★ **Location:** Environment variable `EXPO_PUBLIC_GOOGLE_MAPS_API_KEY`

★ **Justification:**

- Required for Google Maps functionality
- Should be kept secure (not committed to git)

★ **Note:** This is standard practice, not a limitation

Issues & Questions: Resolved Issues

EMFILE Error (Too Many Open Files)

- **Status:** RESOLVED
- **Solution:** Installed watchman via Homebrew to handle file watching
- **Impact:** Development server now runs without errors

Location Permission Handling

- **Status:** RESOLVED (via hardcoding)
- **Solution:** Hardcoded Stanford location for prototype
- **Impact:** Consistent testing, no permission issues

Issues & Questions: Open Questions

Offline Functionality

What level of offline functionality is needed?

Considerations:

- Cache healthcare resources
- Store directions offline
- Offline checklist access

Actions Needed:
Define offline requirements and implement caching strategy

Multi-language Support Priority

Which languages are highest priority?

Considerations:

- Spanish is likely highest priority for Medi-Cal beneficiaries
- Other languages based on user demographics

Actions Needed:
Survey target users for language preferences

Notification System

What types of notifications are most valuable?

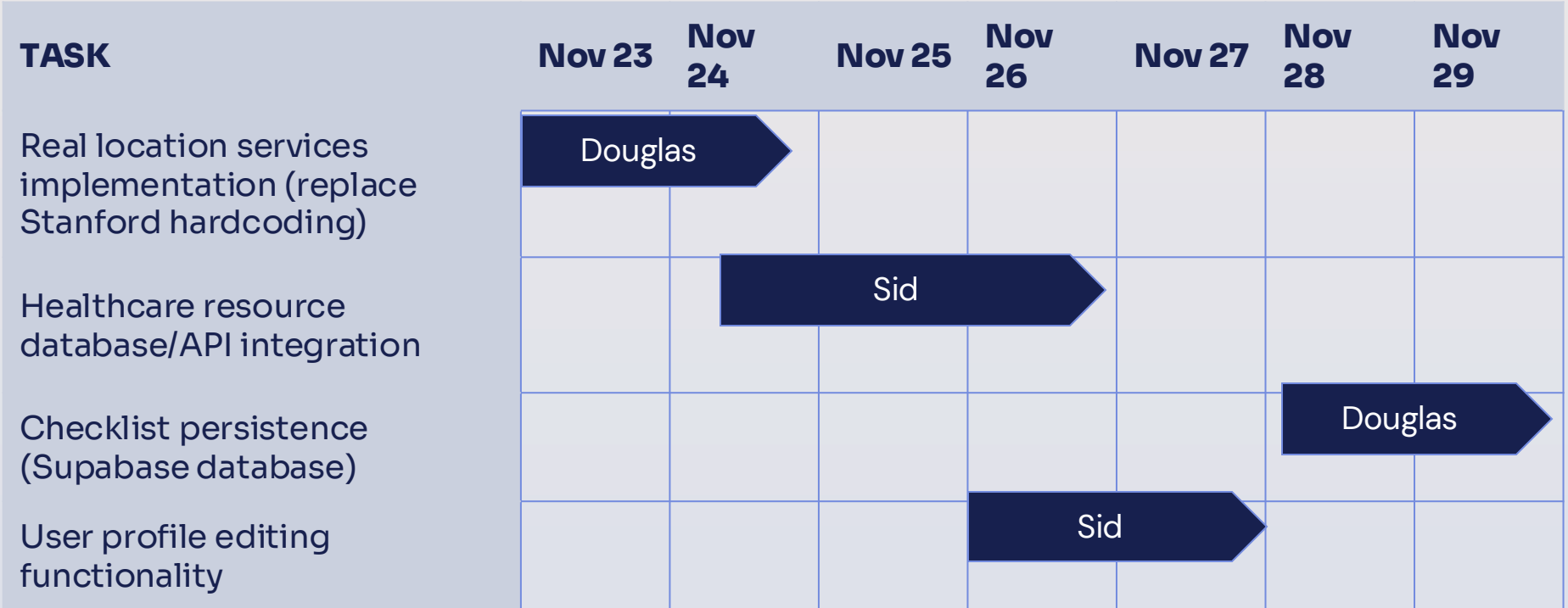
Considerations:

- Appointment reminders
- Checklist task reminders
- New healthcare resources nearby

Actions Needed:
User research to prioritize notification types

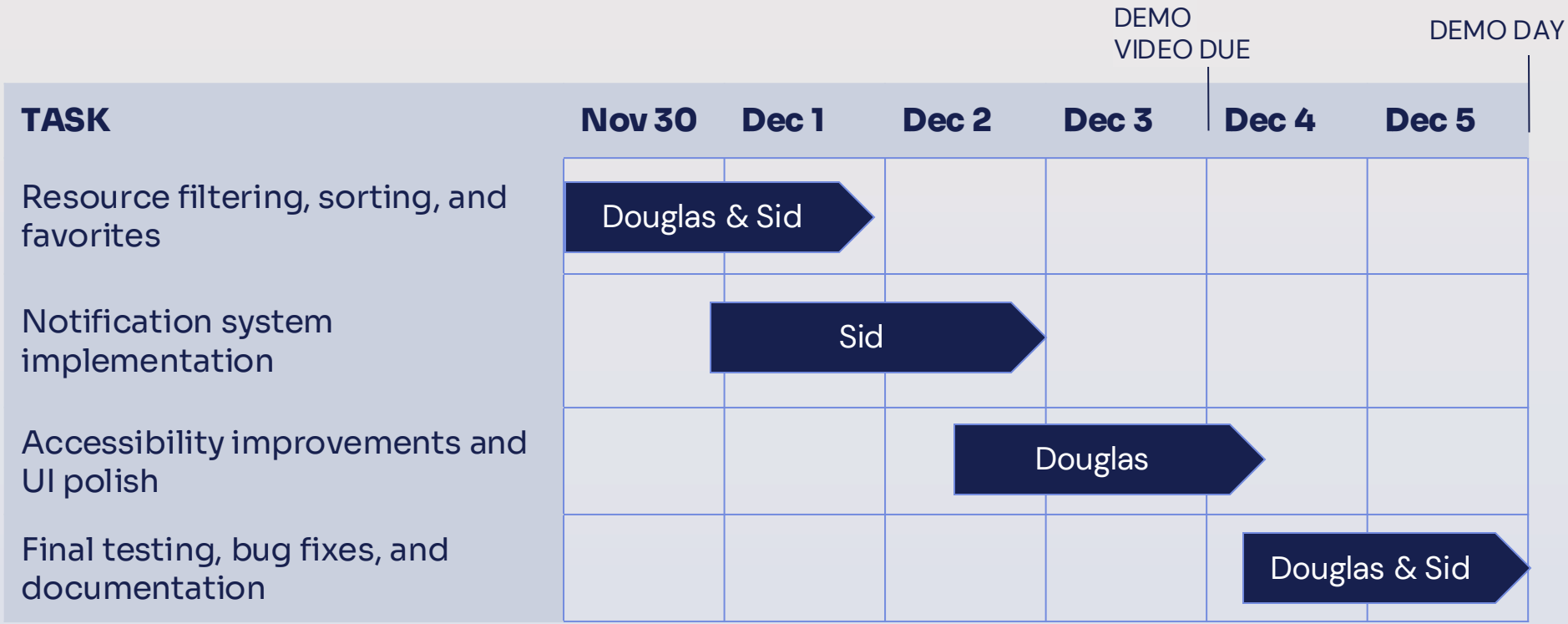
Our Plan for Finishing

Week 1 Focus: Core Functionality



Our Plan for Finishing

Week 2 Focus: Enhance Features & Polish



Team Responsibilities

Backend Dev (Sid)

- Supabase database schema design and implementation
- Checklist and user data persistence
- Healthcare provider API integration

Frontend Dev (Douglas)

- Real location services implementation
- Resource filtering/sorting UI
- Profile editing interface
- Accessibility improvements

UX Designer (Nash)

- Design our project poster
- Compile our project report
- Support frontend design

Live Demo

04

Enjoy our Demo!