GitHub Username: joeyturczak

Driving Reference

Description

Almost everyone in the US needs to learn how to drive. Driving tests are not easy to pass without studying driving rules and guidelines.

Problem: The potential driver wants to pass their test, and while there are resources, it may be difficult to pay for driving school or find these resources online.
Proposed Solution: An app that conveniently has links to many driver learning resources for their state or a state their planning to move to. This app will help give many people access to resources on their mobile device quickly without have to navigate a government website to find what they need.

Intended User

This app is for teens learning to drive, for parents of teens learning to drive, and for anyone who wants to obtain a commercial or motorcycle license as well.

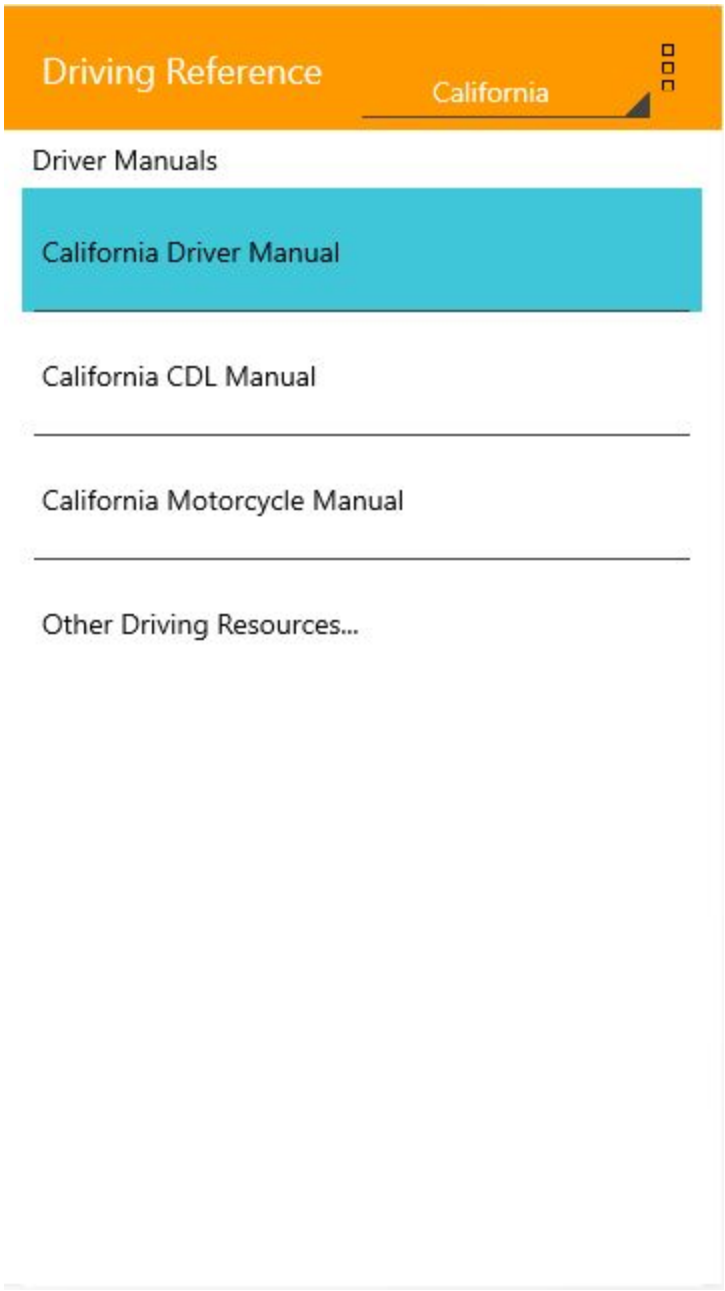
Features

List the main features of your app. For example:

- Uses the user's location to show location appropriate content
- Downloads pdfs to the device for offline use

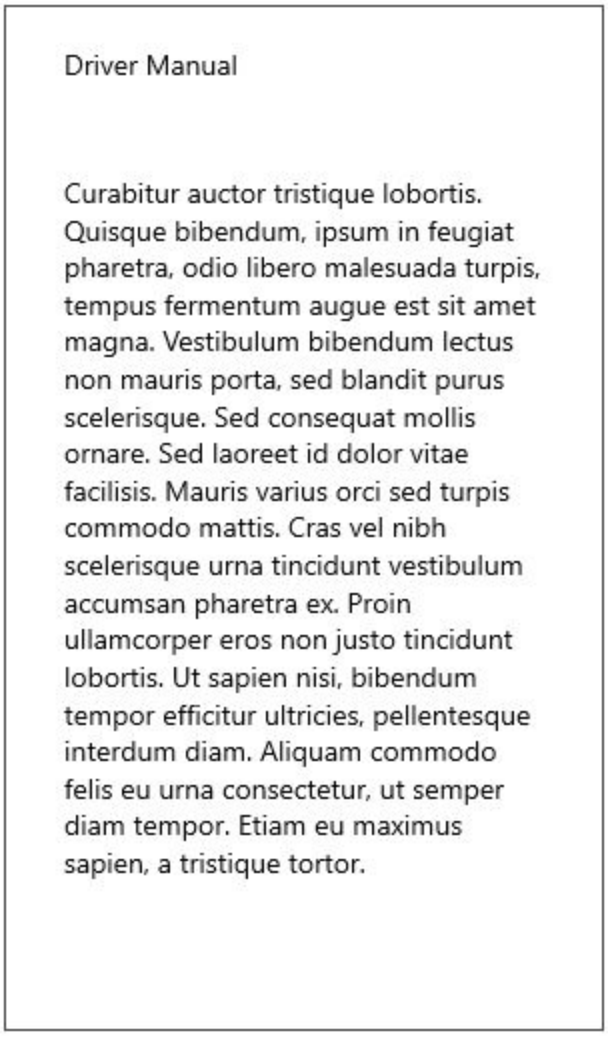
User Interface Mocks

Main Screen



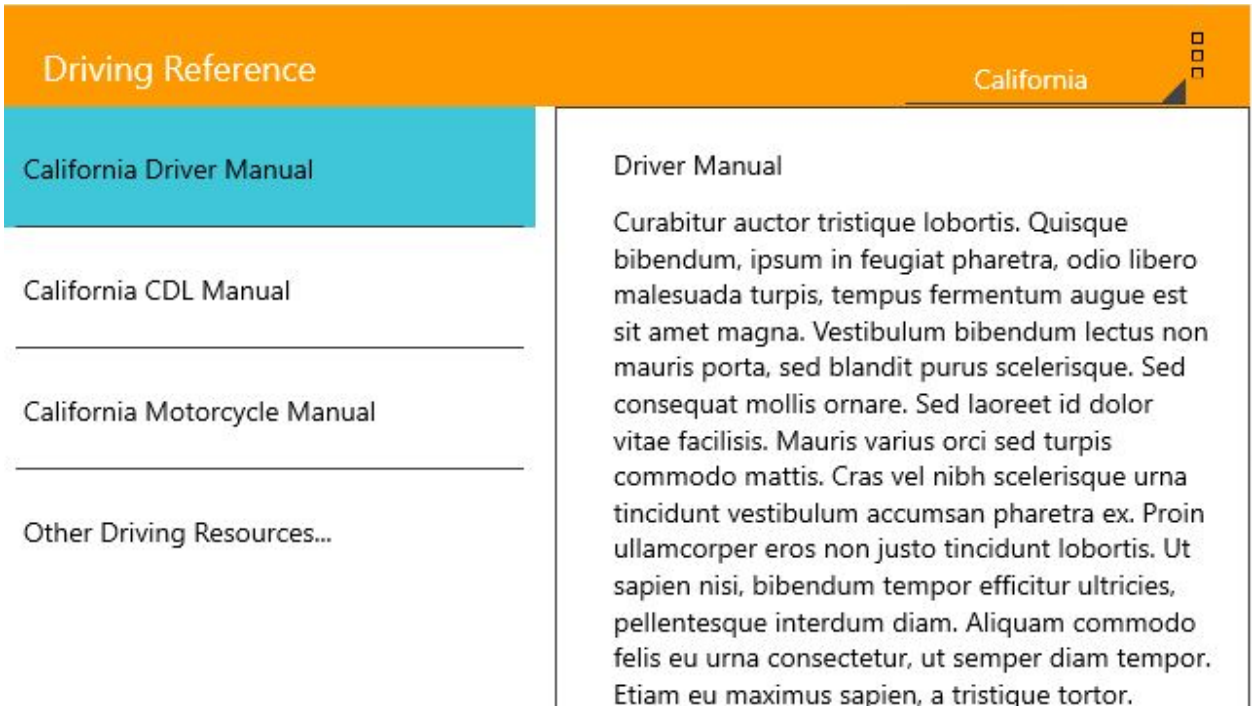
Main Screen. This is a list of the available resources for the selected location.

Detail Screen



Detail Screen. This screen will show the content selected from the main list and provide navigation back to the list.

Tablet



Tablet. On a tablet there will be a master detail flow. The list of resources will be on the left, and the content will be displayed on the right.

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

I will build a content provider to store the information about each resource, such as location, name, and url. This information will be fetched from a Google Cloud Endpoints backend which will use the datastore to hold current information.

Last known location will be stored in SharedPreferences.

Describe any corner cases in the UX.

If the user navigates away from a particular resource, when they go back to it, it will start on the page they left off on.

Describe any libraries you'll be using and share your reasoning for including them.

Objectify will be used to facilitate using Google Cloud Endpoints and the datastore.
https://github.com/objectify/objectify
PhotoView to make an image view zoomable with touch controls.
https://github.com/chrisbanes/PhotoView
AutoFitTextView to resize text in a view with a set width
https://github.com/grantland/android-autofittextview

Next Steps: Required Tasks

Task 1: Project Setup

- Configure libraries
- Configure Google Cloud Endpoints module

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for MainListFragment
- Build UI for DetailFragment
- Build UI for tablet layout

Task 3: Implement Google Play Services

- Store last known location in SharedPreferences
- Create spinner to select location
- Add Admob banner

Task 4: Add Content Provider

- Create ContentProvider class and helper classes
- Setup main list to query database based on location

Task 5: Implement Backend

- Create backend API functions
- Create backend data model
- Populate datastore with information

Task 6: Create Widget

- Create layout
- Create widget service

Task 7: Error Handling and Testing

- Handle connection errors
- Test app for errors and crashes
- Fix bugs