GitHub Username: DeltaJulio

Pantry Bank

Description

Pantry Bank helps you manage your kitchen supplies, making it quick and easy to manage your grocery lists. By keeping track of what food is in the house, you'll know what staples you'll need to pick up. Our recipe book lets you plan out your weeks menu, and lets you know if you're short on supplies.

Intended User

Although a wide audience can find these features useful, families will benefit the most.

Features

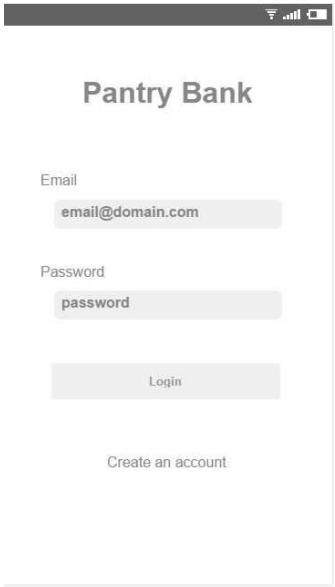
List the main features of your app. For example:

- Save food items to pantry list
- Create recipe lists

- Checklist for grocery runs
- Meal planner
- Checks meal planner vs. pantry list (ex: meals need 3 tomatoes, only 1 in pantry)

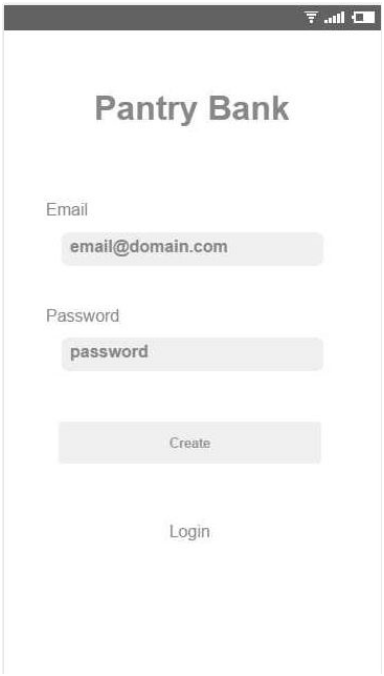
User Interface Mocks

Login



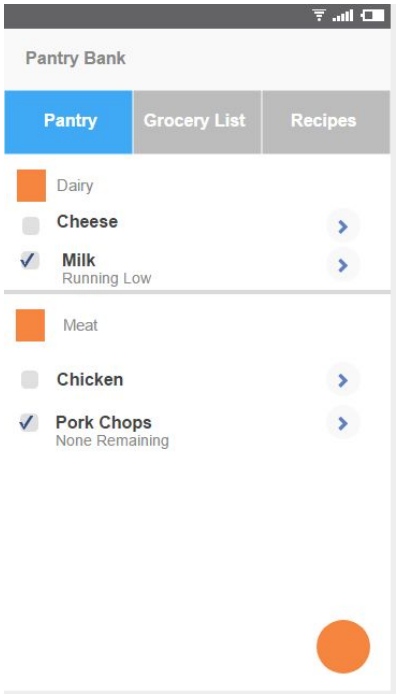
Simple login screen when first running the app. Users can switch to account creation screen.

Account Creation



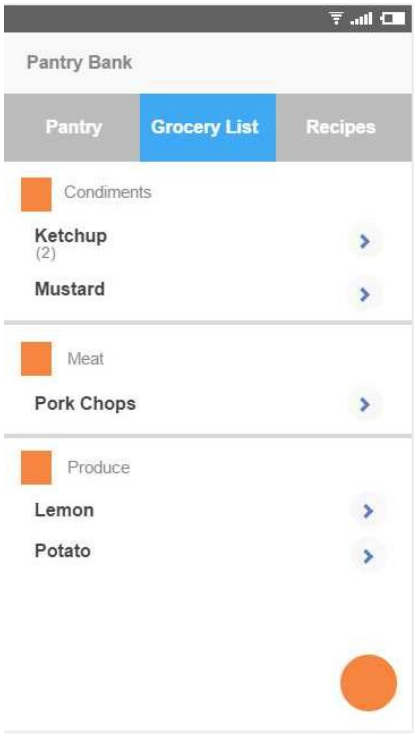
Very similar to the login screen, here users can create a new account.

Pantry



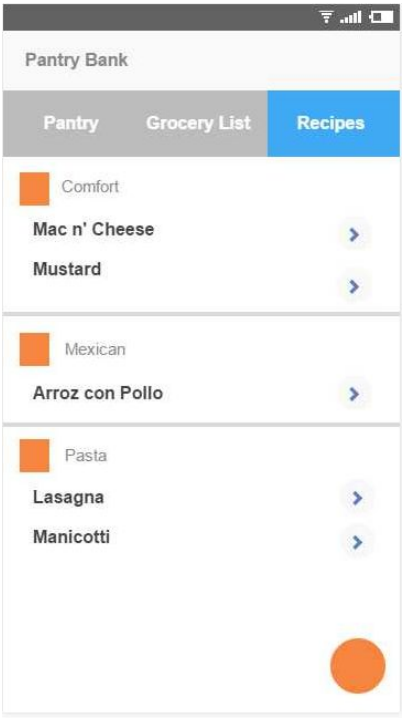
The main focus of the app. This screen shows your inventory of categorized food items. Users can add new items via FAB, and pin certain items to mark them as favorites.

Grocery List



Similar to the pantry screen, except this shows what items are missing or necessary.

Recipes



Also similar in style and function to the two previous screens. Here users can add their own recipes, which allows them to quickly add groups of items to the grocery list.

Key Considerations

How will your app handle data persistence?

I will need to create my own content provider to provide data to widgets.
If I want to keep user data sync across their device, I will need to user a service like FireBase to store that data.

Describe any corner cases in the UX.

Some users might prefer different options for interacting with lists. Some users may find the primary action of list items too easy to press by accident (tap to mark as done) and may prefer to keep that behavior in the expanded menu

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

I will be using Firebase for its realtime database and authentication.

Describe how you will implement Google Play Services.

I will be using AdMob to display banner ads and In-app Billing to sell an ad-free experience.

Next Steps: Required Tasks

Task 1: Project Setup

- Configure libraries
    - Configure Firebase library (in IDE)
    - Configure Firebase app settings (in web interface)
- Login/Registration
    - Create UI screens
    - Implement login/registration via Firebase
- Add main screens (pantry/list/recipes)
    - Create UI for each screen
    - Implement main features (add/remove food items from lists)
    - Implement secondary features (increment/decrement/mark as pinned/etc)

Task 2: Implement UI for Each Activity and Fragment

- UI for login/registration
    - LoginFragment
    - RegistrationFragment
- UI for MainActivity
    - PantryFragment
    - ListFragment
    - RecipeFragment

Task 3: Data Syncing

- SyncAdapter for regular updating
- IntentService for adding items/on demand refresh

Task 4: Handle Error Cases

- Duplicate item entries
- Simultaneous deletion (two devices on same account)

Task 5: Create Build Variant (paid/free)

- Implement AdMob
    - Add banner ad to xml
- Implement In-app Billing