
FoodLab



Recap

Food
Waste



Easy way to search up
recipes that use the
ingredients you have

First Sprint Recap

- Basic login/logout
- Majority of screen activities UI and basic functionality
- Basic search functionality (search by recipe)
- Populate recipe pages

Second Sprint

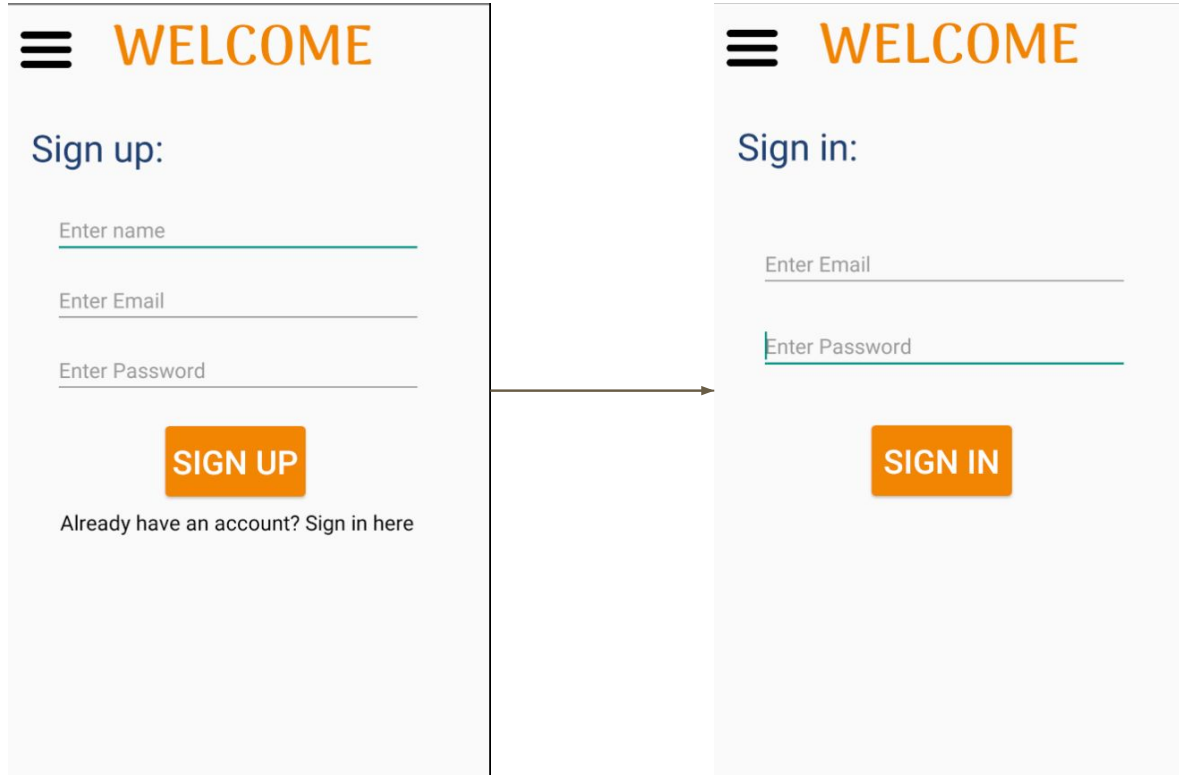
- Migrated user datas and preferences into Firebase
- Changed sign in/sign up method
- Implemented more features in Settings
- Implemented tips and ratings section
- Implemented sorting options
- Fixed some UI to match that of design document

Migrated user datas into Firebase

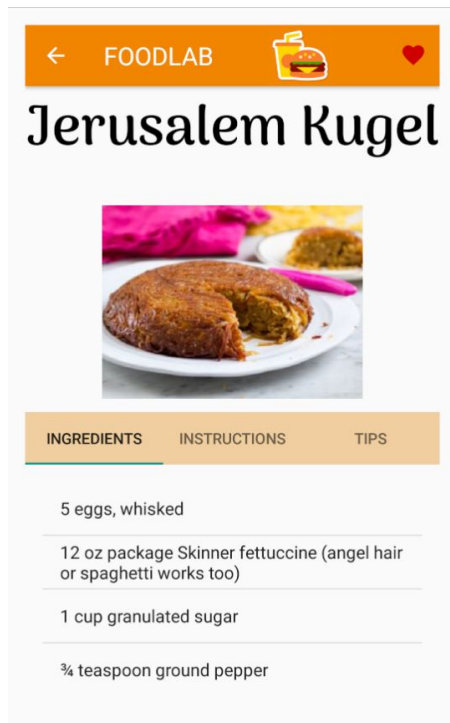


- Easier access to data for other app functions
 - Automatic name input for tips
 - Favorites


Changed Sign in/Sign up method



Implemented Favorite button



Implemented more features in settings

 **SETTINGS**

Change Password:

Change Email:


SAVE


LOG OUT


Dietary Restrictions:

Diary Free ▾

Culinary Preferences:


Italian

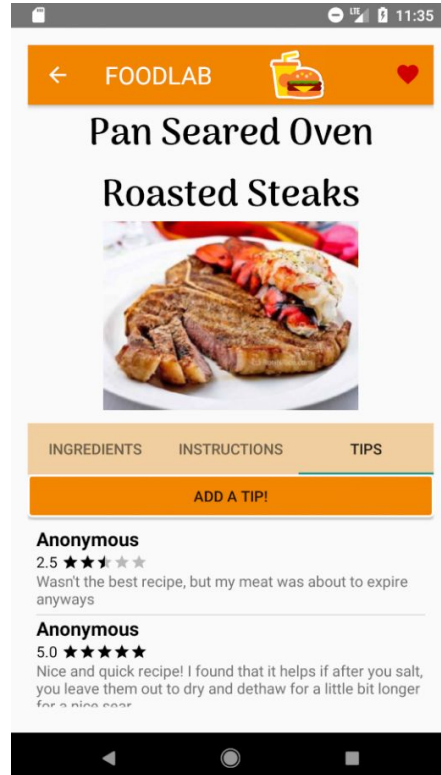
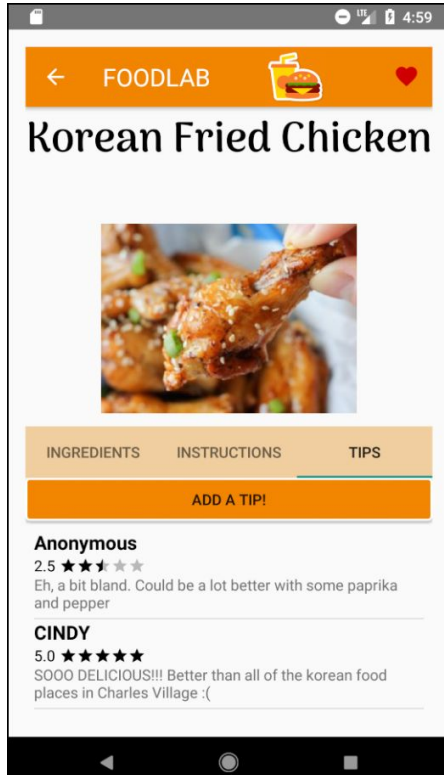

Chinese


Mexican

- Added confirm password field
- Added dietary and culinary restrictions to use for filtering recipes from results

Demo of Sign In/Out and Settings

Tips and Ratings



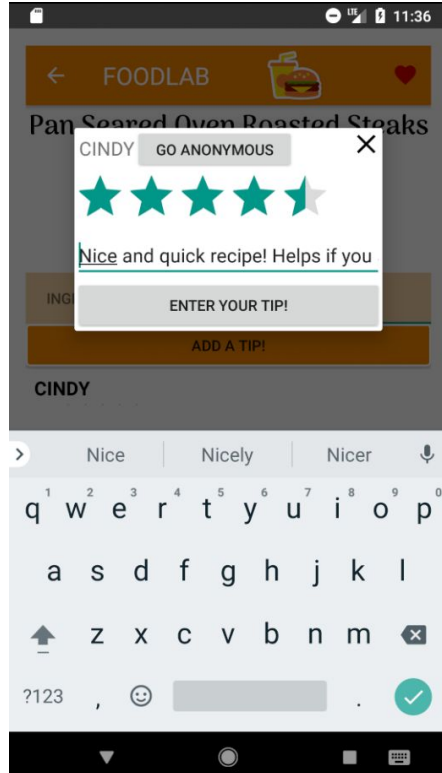
- Added Tips Fragment to allow for user input on recipe quality and possible improvements

Tips and Ratings (cont.)

- Tips are stored in a Firebase Database
- Displayed in a listview using a CustomListViewAdapter to display TipItems

```
public class TipsListViewAdapter extends ArrayAdapter<TipItem> {  
    Context context;  
    List<TipItem> tipsList;  
  
    public TipsListViewAdapter(Context context, int resourceId,  
                               List<TipItem> items) {  
        super(context, resourceId, items);  
        this.context = context;  
        this.tipsList = items;  
    }  
  
    public View getView(int position, View convertView, ViewGroup parent) {  
  
        LayoutInflater inflater = (LayoutInflater) context  
            .getSystemService(Activity.LAYOUT_INFLATER_SERVICE);  
        View listView = inflater.inflate(R.layout.tip_item, null, true);  
  
        TipItem tipItem = tipsList.get(position);  
  
        TextView tip = (TextView) listView.findViewById(R.id.textView10);  
        TextView name = (TextView) listView.findViewById(R.id.textView11);  
        RatingBar rating = (RatingBar) listView.findViewById(R.id.ratingBar2);  
        TextView ratingNumber = (TextView) listView.findViewById(R.id.ratingNumber);  
  
        tip.setText(tipItem.getTip());  
        name.setText(tipItem.getUser());  
        ratingNumber.setText(String.format(Locale.US, "%.1f", tipItem.getRating()));  
        float number = tipItem.getRating();  
        rating.setRating(number);  
        Drawable drawable = rating.getProgressDrawable();  
        drawable.setColorFilter(Color.parseColor("#000000"), PorterDuff.Mode.SRC_ATOP);  
  
        return listView;  
    }  
}
```

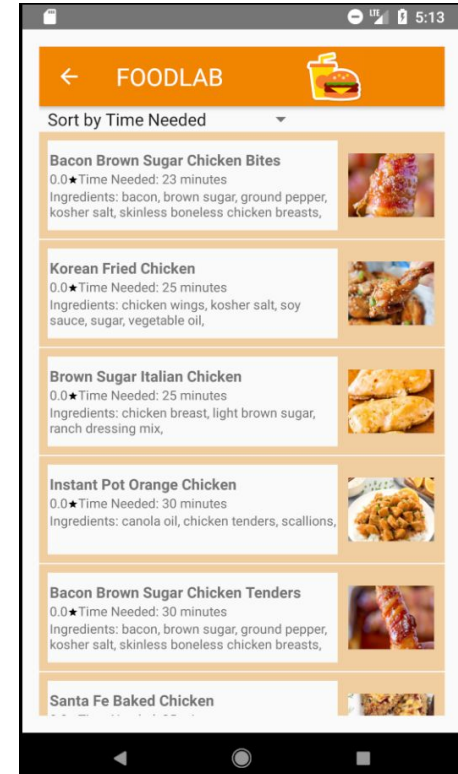
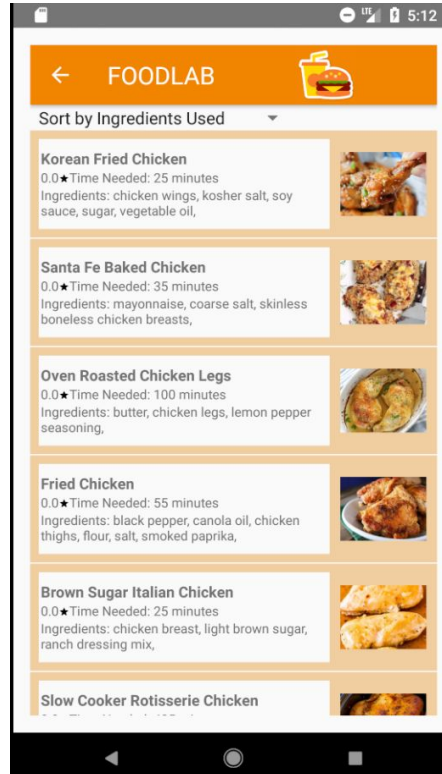
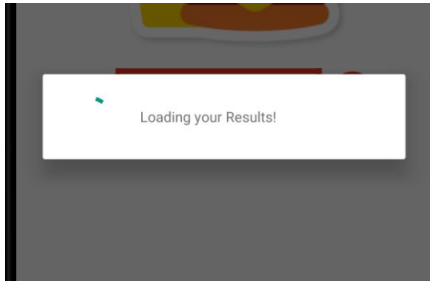
Tips and Ratings (cont.)



- Able to add ratings and/or tips through a dialog fragment
- Able to submit anonymously
- Hold your tip to edit

Sorting Recipes and Freshened-Up List

- Listview shows more information
- Able to sort by:
 - Ingredients Used
 - Time Needed
 - Rating
- Pushed processing time to be front loaded



Full Demo

Give us some ingredients you have!