

## How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [ Select All → Copy → Paste into new document ]
  2. Name your document file: “**Capstone\_Stage1**”
  3. Replace the text in green
- 

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** vladimirjeune

## Prep 4 Week

### Description

Prep 4 Week, is an Android app that helps you to keep track of the prep work for your next week's meals. You can make time in your hectic week for healthy home cooked meals instead of takeout. The trick is doing all the prep work for your weekly meals, on one day a week and saving the prepared ingredients for use when you are ready. Just take a couple hours, one day a week, to prepare and properly store your ingredients. Then, when you are ready to cook, your ingredients are ready to go.

Things that can be prepped include vegetables; meat that needs to be cut, or seasoned; or even sauces and fillings.

## Intended User

Preppers, and people who want their families to eat a healthy dinner, but feel that they don't have the time to cook during the weekday craziness.

## Features

- Holds preparation information for users
- Gives an easy way to tell if you fully prepped for the week's dinners
- Helps users plan the weeks dinners
- Can share list of items that still need to be bought or prepped to others
- Widget can show Grocery or Prep checklist.

## User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, [www.ninjamock.com](http://www.ninjamock.com), Paper by 53, Photoshop or Balsamiq.

**Screen 1**

P4W

Prep Day 9/17/19

Tuesday

Chicken Marsala Tu

Poutine Tu

Corn Bread Th

View Modify Delete

Prep Day 8/20/19

Sunday

Ginger Roast Turkey M

Veggie Lasagna M

Jeweled Rice W

View Modify Delete

(+)

#1 MainActivity: This is where the user can peruse previous plans. FAB is to make New Plans. Overflow Menu holds Search & About info. This Activity is a list of plans from the past up until now. When the app is first started it will be empty except for the FAB. Each plan can be Viewed, Modified, and Deleted, as the User wants. A previous plan can be renewed by hitting the Modify Button at the bottom of the card, then changing the Prep Day to today, or a time in the future. This will clear all checkmarks, so they can be checked again as the user prepares their week of dinners.

**Screen 2**

## Create Plan

Prep Day

Tuesday, Feb 27<sup>th</sup>

(T) W (Th) Fr (Sa) Su M

Recipe Name

Ingredient to prep Amt

Preparation

Day

Recipe Name

Sub-Recipe to Prep

Ingredient  Prep Amt

Preparation

Day

#2 Create/Modify: User will pick the day and time to do meal prep. As they fill in the recipes and the days the recipes will be made, the days will be circled. The User will fill in the Recipe name, the Ingredients/Sub-Recipes(fillings, salad, sauces) that can be easily prepped ahead of time. The Calendar button will allow the user to pick Date and Time. The + Button next to Recipe, Ingredients/Sub-Recipes will allow another row of that type to be added until the user is done. The section of Recipes will be scrolling under the Prep Day area; which will be stationary.

**Screen 3**

View Plan

Prep Day: Tues, Feb 27<sup>th</sup>  
5:00 PM

(T) (W) (F) (S)

Wed: Chicken Cacciatore

Skinned, Boneless, Cut 2 lbs  
Chicken Breast

Pounded  $\frac{1}{2}$ " thick

Medium Yellow Onion  $\frac{1}{2}$   
Thinly Sliced

Buono Salad

Tomatoes 3

Cored, Chopped, Salted

Fri: Burrito Bowl

Garlic Cloves 3  
Peeled & Diced

#3 View Plan: This is where the user can view the plan they have just created. They can also view any plan they have selected. It states the Prep Date and Time, as well as show what days dinners have been planned for cooking.

Under the Prep Day Information and Cook Day Indicators, there will be a Scrolling View of Cards containing the information that was entered when this plan was created. The Overflow menu can go to Modify, and Share.

**Screen 4**

## Availability

- 3 1/2 Medium Yellow Chicken  
Onion Cacciatora
  - 1 Onion Chili
  - 8 oz Cremini  
Mushroom Chicken  
Cacciatora

#4 Availability Checklist: Checklist of ingredients that will be Scrollable. The Ingredients can be reordered how the User wants. But it defaults to the unchecked items being first. Each item consists of a checkbox, indicators that the element can be moved, amount of ingredient needed, ingredient name, and the name of the recipe it is for. Overflow menu goes to Shareable Grocery List of unchecked items. Toggle in AppBar can change the Checklist to be a Prep Checklist. An appropriate Transition of elements will occur if that is triggered.

**Screen 5**

Prep

2 cans White Chili  
Beans

Drained & Rinsed

302 Roma Tomatoes Burrito  
Cored & Diced Bowl

1lb Boneless Skinned chicken  
Chicken Thighs Caviture

Trimmed, Seasoned, Refrigerated

||||| / / / / / / / /

#5 Prep Checklist: Items in this checklist will be similar to Availability Checklist, but include how the ingredient is to be prepared. Overflow Menu goes to Share a list of items that have yet to be prepared. This way if some things have not been done; the work can be delegated before dinner time.

**Screen 6**

## Calendar

Prep Day

Feb 27<sup>th</sup>, Tues, 6 PM

Chickpea Salad

Wed Feb 28<sup>th</sup>, 7 PM

Chicken Cacciatore

Burrito Bowls

Friday Mar 2nd 7 PM

Lasagna

#6 Calendar: Calendar will show Prep Day and all Meal days in a single view. If the Prep Day is GREEN, that means that, all prep has been done, and ingredients have been properly stored for later use, for all Recipes. YELLOW, means that there is some prep that is still needed and any recipes on this page that are Yellow still need some work done. Lastly, Prep Day will be RED if any recipe is still waiting for ingredients to be bought. Any following Red Recipe will be the one that requires attention.

Clicking on a GREEN View will go to the View Checklist page. YELLOW View will go to the Prep Checklist, and RED will go to the Availability Checklist.

**Screen 7**

## Grocery List

- 1 onion Chili
- 8oz Creamy Chicken  
Mushroom
- 12oz Broccoli Roasted  
Veggies
- Crumble Topping Chicken  
Pot Pie
- Flour
- Parsley

#7 Grocery List Widget: Will be populated with the non-checked-off items from the Current Meal Plan; if there is one. If there is not one, it will say, “**No Current Plan**”. If all ingredients needed for preparation are present then it will say, “**All Prep Ingredients Ready**”.

Item in a populated list will consist of a Checkbox, Amount of item needed, Item Name, and Recipe Name. Grocery List can be shared.

**Screen 8**

Search by Recipe Q

Prep Day: 11/17/18

Chicken Cacciatore

Bacon Cornbread

Jewelled Rice

Prep Day: 10/27/18

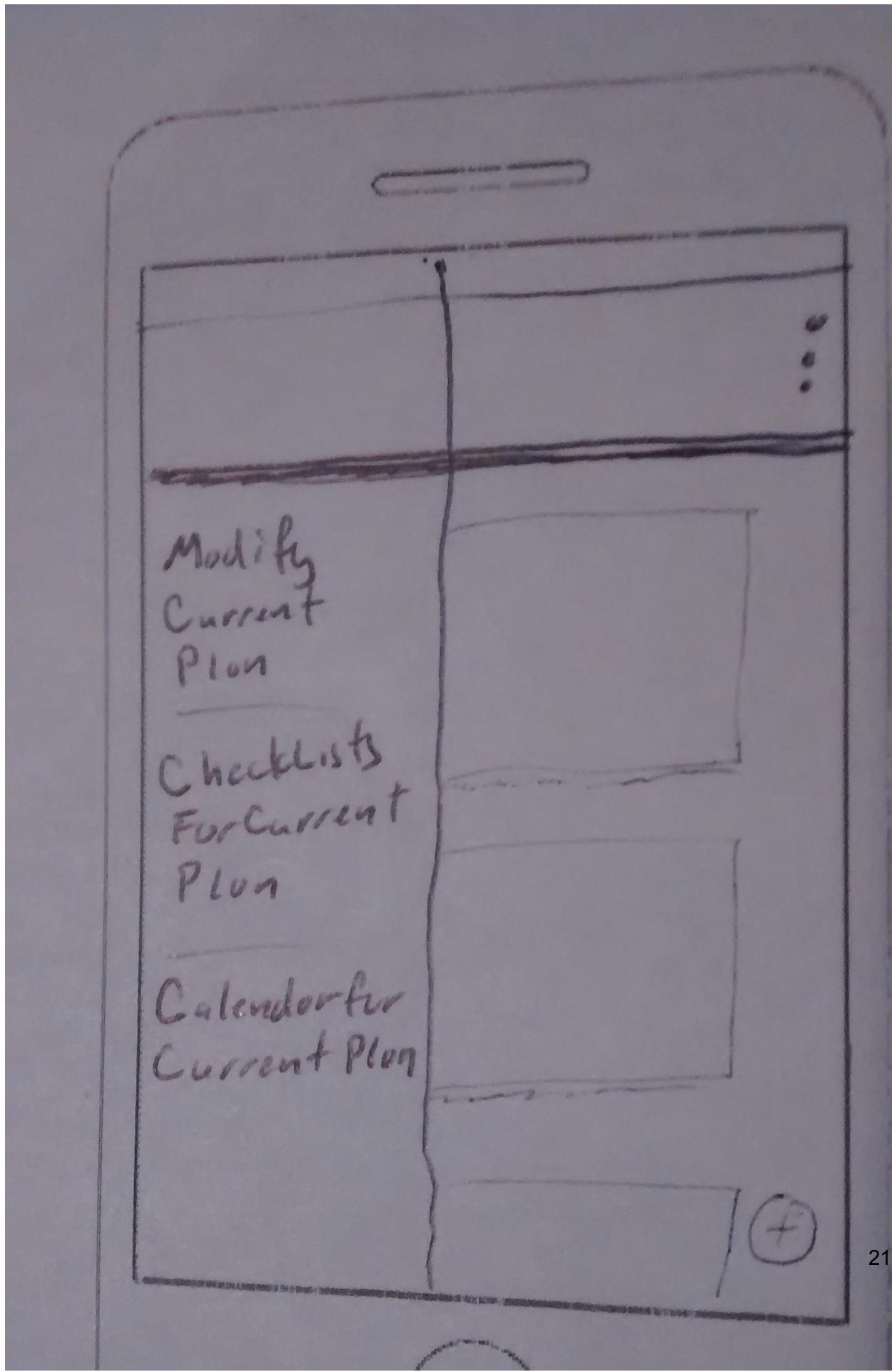
Corn Salad w/Bacon

Red Grape Muffins

Fennel Salad

#8 Search Page: Will consist of a Search Box, over a scrolling area of Cards. The cards will ID the matching Meal Prep Plans by Prep Day Date, and associated Recipes. Clicking on a matching plan takes you to that plan's View.

**Screen 9**



#9 Navigation Drawer: Will allow navigation to Activities that have to do with the current Active Plan. The Active Plan is the last one that was created. If it is case that there are no current Active plans the Activities will be empty. You can Modify the Current Plan, View Checklist (Prep/Availability) for the Current Plan, and the Calendar for the Current Plan.

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

## Key Considerations

### How will your app handle data persistence?

*Describe how your app will handle data. (For example, will you build a Content Provider or use Firebase Realtime Database?)*

Prep 4 Work, will use a Content Provider and a Database to persists data about the User's intended Prep Day and cook days, as well as, prep ingredients to recipes that are entered. Past Prep Plans will be available for modification and could be made to be this week's plan. ASyncTasks and Loaders will be used as necessary.

### Describe any edge or corner cases in the UX.

*For example, how does the user return to a Now Playing screen in a media player if they hit the back button?*

There are a few Activity specific pathways in the App. So in order to make sure that the user can Get anywhere they want, there is a Navigation Drawer that allows the user to go to main different parts of the app. The app will have to deal with the fact that there can be no data if this is the first time the app has been used. Activities will have to have appropriate cues for the user about the situation.

Old Meal Plans still have access to the Checklists, but they will not be interactive since they have past. The paths in the Nav Drawer always refer to the Current Active Plan, but all the parts of the app are reachable for older Meal Plans through normal means.

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

*For example, Picasso or Glide to handle the loading and caching of images.*

I want to start to get used to using Databinding. So I will need a Databinding Library, like [Butterknife](#).

If it is available for the latest version of Android, I would like to use [Stetho](#). I enjoyed using it for a previous app. It helped me quite a bit during that project.

At this time I do not see a need for too many pictures. But if I change my mind, I will use [Picasso](#). I have some experience with using for other projects and I feel that I like the way it works.

Describe how you will implement Google Play Services or other external services.

*Describe which Services you will use and how.*

I will use **Firebase Cloud Messaging** to send **Notifications** to the Users about their scheduled Prep Day, and Dinner times.

## Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

### Task 1: Project Setup

*Write out the steps you will take to setup and/or configure this project. See previous implementation guides for an example.*

*You may want to list the subtasks. For example:*

- Configure libraries
- Something else

*If it helps, imagine you are describing these tasks to a friend who wants to follow along and build this app with you.*

First I want to add what I want to use to my App build.gradle. I will be using **Google Play Services**, **Butterknife**, and **Stetho** and **Picasso** if applicable.

I will also set up a **GitHub repo** so that I can use the version control and Git's lightweight branching to help me make changes to the app without fear of not being able to revert to a pristine state.

Work out how the **Database** will store the data for easy usage. Define Database and use Stetho to see how well it handles queries.

Set up Gradle to be able to build and test easily. This will shorten amount of time between tests.

## Task 2: Implement UI for Each Activity and Fragment

*List the subtasks. For example:*

- *Build UI for MainActivity*
  - *Build UI for something else*
- 
- First, build the UI for the MainActivity. I will have to deal with being empty and having data. The empty state may have a message in the background telling the user to use the FAB to create their first Prep Meal Plan, so that they can have enough time for home cooked meals during the week. The data containing state will be a scrolling list of data about the meal plans that have already been created. The Prep Day date should be prominent, and the Recipes names should also state what day those recipes are to be prepared on. Modification of these Prep Plans should be easy and obvious to the user (View, Modify, Delete). FAB will lead to Create Plan, and Overflow Menu will lead to About, and Search.
  - The design should be material, with proper elevations throughout.
  - Build UI for Create Plan: Important parts of the page include Calendar button, that you will have bring up appropriate Date and Time Choosers, Prep Day Info should be persistent, and Days of the Week. Recipe information that user provides will scroll when there is too much information to hold on the page. Ingredients and Sub-Recipes can all be added to the Recipe until the user is done. Recipes should also be able to be added until the user is finished inputting information.
  - Build UI for View Plan: This data is mostly static, so important part is that the Overflow Menu that sends us to modify works, on the Current Meal Plan, and previous ones. If the Weekday bullets are a Custom View, then they should be used here as well.
  - Build UI for Calendar: Follow directions that came with Calendar image above. It is important that the user can tell at a glance if they are ready for Prep Day. This will keep disappointment about not being prepared to a minimum.
  - Build UI for Navigation Drawer.
  - Build UI for Search Activity.

- Build UI for Grocery List Widget
- Build UI for Checklists: The Checklist Activity will have two Fragments that it uses one at a time. Which one shows up depends on the toggle. The AppBar Title will also depend on the toggle.

### Task 3: Your Next Task

*Describe the next task. For example, “Implement Google Play Services,” or “Handle Error Cases,” or “Create Build Variant.”*

*Describe the next task. List the subtasks. For example:*

- Create layout
  - Something else
- 
- Create Espresso Test. Actually, create more detailed Espresso tests
  - Tighten up the Material Design and make sure you are following guidelines
  - Work on some transitions for the app
  - Work on Notifications for when Prep Day is getting close
  - Work on the Widget

### Task 4: Your Next Task

*Describe the next task. List the subtasks. For example:*

- Create layout
  - Something else
- 
- Make sure that project is conforming to Android guidelines.
  - Move all strings to strings.xml

### Task 5: Your Next Task

*Describe the next task. List the subtasks. For example:*

- Create layout
- Something else

Add as many tasks as you need to complete your app.

---

### Submission Instructions

- After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
  - Make sure the PDF is named "**Capstone\_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone\_Stage1.pdf**"