

Cookery: Recipe Generator

Vladislav Mazurov
a11710356@unet.univie.ac.at
University of Vienna
Vienna, Austria

Julian Saria
a01608505@unet.univie.ac.at
University of Vienna
Vienna, Austria

Rosi-Eliz Dzhurkova
a11918541@unet.univie.ac.at
University of Vienna
Vienna, Austria

Lukas Pezzei
a11834075@unet.univie.ac.at
University of Vienna
Vienna, Austria



Figure 1: Cookery Logo.

ABSTRACT

This article talks about making recipe generation app for iOS devices focusing mostly on design processes. It includes research of related work, design prototypes, implementation approaches, evaluation with test-users and reflection on the work done. This article was made for the "Human-Computer Interaction" course at University of Vienna.

KEYWORDS

design, human-computer interaction

1 MOTIVATION

Our motivation was creating an app with user-friendly UI / UX and sustainability in mind. First of all, it should be used by private individuals in their own households and help ensure that all food is consumed and none is wasted. It could also be used by restaurants and larger establishments, but is more specialized in giving private households cooking ideas. Ideally our App should be used several times a day to ensure culinary diversity in the households of private individuals.

1.1 User Analysis

A basic distinction was made between our two groups: "Novice cook" and "Advanced home cooks and professionals".

"Novice Cook": This group does not yet have much experience of cooking and therefore has no recipe ideas. They would then primarily want to look for easier recipes to start with and generally things that are less complex. Furthermore, they would like a very detailed description of the recipes.

"Advanced home cooks and professionals": This group has already gained a lot of experience and has a certain basic cooking skills. Accordingly, this group would prefer more unusual and creative recipes with which they can impress their family and friends even more. who want to find new recipes.

In addition, you should now consider the following factors:

- People with illnesses (allergies / intolerance / diabetes): They want their needs to be taken into account, for example through various options in the app.
- People who want to waste less food: They want to use up the last of the ingredients that can still be found in the fridge at home, but often have no idea what to do with them.
- Vegetarians / vegans: They expect meat-free food and an option for meat-free meals or meals without animal products.
- People with little time: They expect quick cooking, recipes that are simple, require little effort and taste good.
- People with little time: They expect quick cooking, recipes that are simple, require little effort and taste good.

- People who want to go on a diet: They expect low-calorie food, as little fat as possible and generally support for receiving only very healthy recipes that are nevertheless very filling.
- People who want to eat properly to support their training: They expect protein-rich foods above all to help them build muscle.
- People who have problems chewing (after treatment or in general): They expect the opportunity to only look for recipes that are easy to chew.
- People who eat delivered food / frozen products: These users want to avoid cooking for various reasons and make it as quick and easy as possible for themselves.

1.2 Tasks

Our main tasks for achieving our goal of creating this app were:

- Find potential users and use cases
- Define the apps functionality
- Design the user interface / UX of apps based on our analysis
- Implement the backend for the beta test
- Testing with the focus groups
- Release app for beta test with recipes in project context

2 RELATED WORK

2.1 Analysis of the Literature

Current trends suggest that nearly 33% of food is thrown away, more than half of all consumers are willing to buy products with a positive environmental agenda, and nearly 9 in 10 millennials say they are likely to buy the solutions for certain support social problems. This means that our app is in the middle of current trends. If we look at what the future of food and technology will bring, then it definitely has apps on our smartphones. In addition, it can be AI bots with a human touch for user interactions, referrals, social communities, and providing missing ingredients within the app [1]. We could use this knowledge to prototype our app.

First, we need to find our specific audience among general users. The suggested search is that almost 60% of young adults between the ages of 25 and 34 use their smartphone in the kitchen [4]. Another helpful insight into the user base is that "Consumers who are heavily involved with mobile technology and food are most likely young adults and parents" [2]. This gives us the information we need to move forward in choosing the right personas.

Regarding the design of the user interface, some research and articles bring up interesting points on the matter. For many people, making something of your own according to the recipe is a fun and creative activity. But most young users struggle with deciding what to cook. Research suggests capitalizing on this stressful experience and making it a fun one [4]. We could potentially guide users to the first recipe that matches the ingredients in the fridge or make it a game (gamify approach?).

One of the main topics of our app is sustainability. If we double this point, we can potentially also reduce the time it takes users to cook a meal from a recipe. Another study found that their participants spent more time cooking than they needed, therefore more energy was spent on kitchen appliances. A simple solution for adding an internal timer to the app is suggested [5]. We could easily

implement this function and show a required time with a timer, which would potentially reduce cooking activities to a minimum, hence environmental pollution.

Another study found that most of the focus groups surveyed did not wash their hands after touching their phone while cooking. Currently, in the midst of the pandemic, it is more important for users to minimize these procedures. This study suggests possible solutions to this problem that we could incorporate into our app. One is to display a reminder message to wash your hands between touching the phone and eating, another: implementing new methods of scrolling through the recipe, such as voice activation. [3]

We also found a case study that was ideal for us as it explores a recipe app similar to ours [6]. It would help us create a meaningful and accessible user interface / UX based on current trends. We could evaluate colors, flows and animations between screens, features and design of this case and take a similar approach in our future app.

2.2 Analysis of competing products

FridgetoTable.com: is a recipe generation website. In contrast to our system, this works without an app or a "smart refrigerator" but is controlled solely by adding ingredients manually. A disadvantage can be seen from this and you always have to check which ingredients you have at home first. Another negative aspect is that no precise nutritional values or portion information are given. As an advantage it must be mentioned that there is a large selection of filters. Not only that you can limit it to vegetarian or vegan, for example, no, the ready-made recipe ideas are also divided into categories such as "pasta" or "rice" or even "child-friendly". You can also add what you always have in your home, so you don't have to enter it every time.

Myfridgefood.com: It is particularly positive that calories, carbohydrates and protein information were evident. Since there is currently the trend towards "Know your food", this is a feature that many users definitely like. The approach is well chosen but unfortunately it doesn't do much because there were no units next to it. Unfortunately, it is just as unfortunate that there is a ready-made list of ingredients from which you have to manually select which ingredients you have. This means that the start page is very long and you have to scroll forever to select everything. It would be better to have a field where you can enter your ingredients and, in the best case, with an automated suggestion. Furthermore, the layout of the page has been chosen unfavorably and this has resulted in only about a third of the website being described with content, the rest is a background image. It is therefore difficult to keep track of things.

Supercook: The Supercook app also allows users to add individual ingredients via voice. An important first step to create accessibility. Another successful feature is the "Cuisine" filter because it allows you to choose which culture is cooking, i.e. whether you want Italian, French or, for example, Indian cuisine. Another important point is that Supercook allows you to save your favorite recipes right away. A negative point of criticism that struck me is that the website is not supported by all browsers, so it is not possible in Safari to change the selection of ingredients as intended.

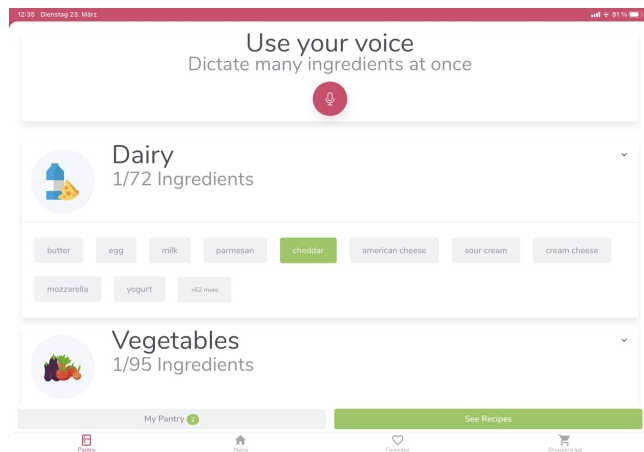


Figure 2: AMR Systems LLC., Supercook (Version 1.17.4) Mobile app. App Store. (<https://apps.apple.com/us/app/supercook-recipe-by-ingredient/id1477747816>).

My Fridge Food: The My Fridge Food app also wrote down all the ingredients individually, but solved the problem of a long list by making four columns visible at once. However, the app advertises with a subscription, which is unfortunately a shame because it is otherwise very successful. Furthermore, the app shows which ingredients are still missing before you click on a recipe. Of course, there are also less successful features and since there are only a few filters to choose from, the app still lacks a certain fine-tuning.

3 DESIGN

The application has incorporated a functionally minimalistic design, which focuses on the ease of use. At the same time, it is able to present its user with a vast functionality. It is important to note that the latter was established as the main focus during the conducted user interviews. Besides functionality, two of the most appealing aspects of the app are its customization and its ability to perform complex recipes queries in accordance to different criteria set by the users. For the purpose of establishing the UI requirements prior to the application development, a set of predefined primary and secondary components, such as colors and fonts, are offered. A description for the correct paddings and margins is also introduced prior to the application's implementation. The previously established specifications are strictly followed on all screens.

The main theme of the application incorporates a set of green colors, highlighting the logo of Cookery. Furthermore, numerous physiological researches suggest that the color "green" has beneficial effects towards calming the mind and relaxing the eyes. The application uses a set tab bar with a pinned search button, whose purpose is to provide the user with a quick search capability regardless of the currently active tab. Another aspect of the UI worth noting is the showcased card design, which facilitates the user understanding of the current context and makes it intuitive for the users to navigate through the different flows.

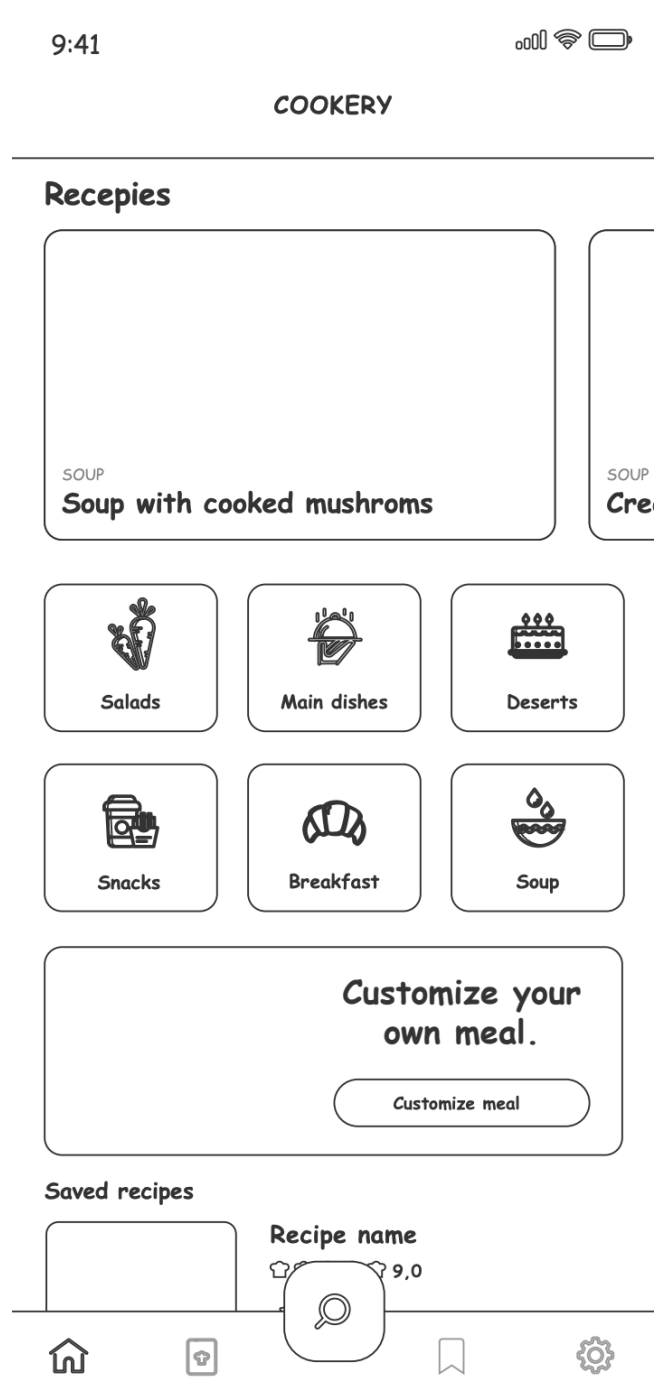


Figure 3: One of the Prototypes for the App.

4 IMPLEMENTATION

The application was implemented using the Swift language on XCode 12 IDE supporting iOS versions 14.0 and above. The used architecture is MVC complemented by the Facade, Adapter and Delegation patterns. The networking layer is comprised by the

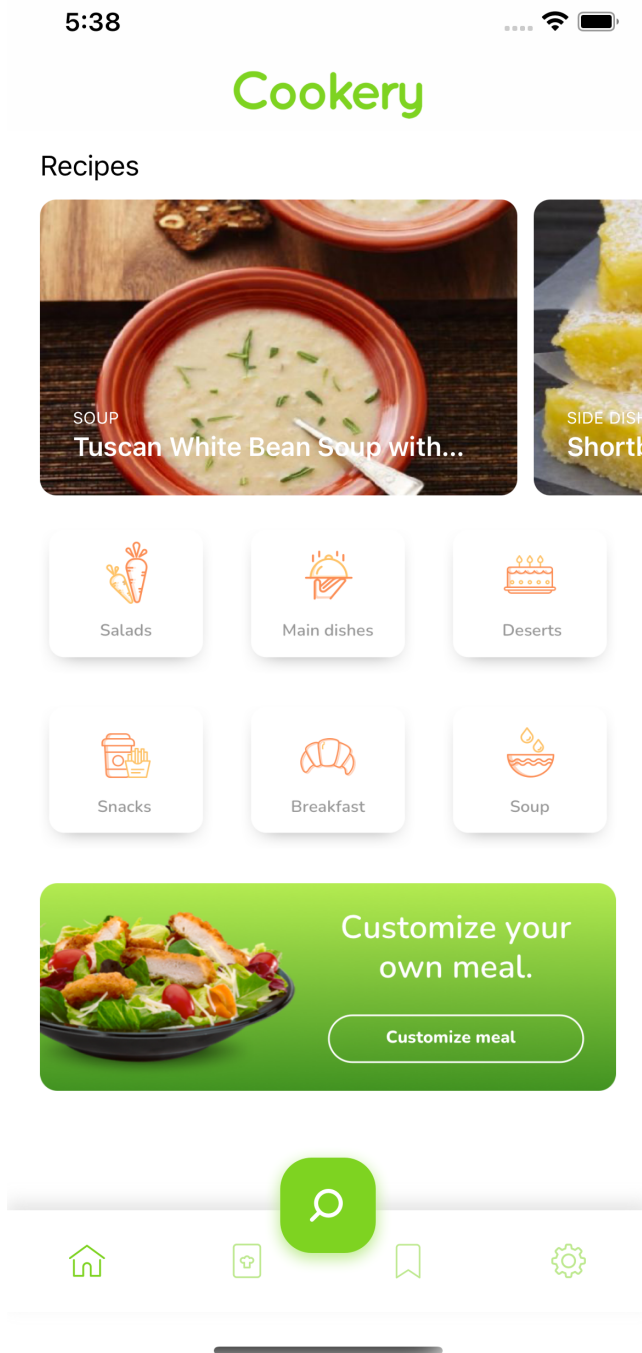


Figure 4: Home Screen of the App.

NSURLSession native module, supplied by the Apple's SDK. The Application uses The Spoonacular API, consuming the Following services:

- Recipes Search;
- Random Recipes;

- Recipe Details;
- Recipe Instructions;
- Ingredient Auto completion.

The storage mechanism is based on the native UserDefaults component for persistent data containment. The project is separated into logical modules representing the different screens. The layout for all modules is presented by a shared interface builder – the storyboard. The inter module navigation is accomplished with the help of the Storyboard Identification system. The tab bar hierarchy is implemented using view controller containment with a custom handling for the tab switches and the search button invocations. Custom UIView components are introduced for the Recipe details macros indicators and for the intolerances chips, using native UIKit components. The majority of logical modules incorporate a mix of UITableView and UICollectionView fragments for populating list and data items, configured using the Delegation Pattern. The data composition for the majority of the functional class is established using a mix of static data, singleton invocations and basic dependency injections, assisted by the introduction of a Protocol oriented programming. This is supposed to simplify any addition of unit tests aiming for the future development of the application.

4.1 Challenges

First, it should be noted that an excessive number of requests were generated, when the ingredients autocomplete functionality was initially attempted. **Solution:** Introduction of debouncing – a mechanism which takes into account the entry input speed and once its algorithm determines that the user has finished their input, it triggers its event. The event creates the service call invocation for the current query.

Second, custom Search button on the TabBar, which is not supported by the UIKit Framework, should appear. **Solution:** Introduction of a custom TabBar component with a custom view container for hosting a view controller containment mechanism

Third, data for particular logical modules had to be extracted from multiple BE services. **Solution:** Support for a queue based data aggregation was introduced to the networking layer, allowing for queuing a finite number of requests to be performed synchronously and for establishing a notification mechanism, which triggers a completion handler once all the data is aggregated.

5 EVALUATION

We asked 8 different test-users where everybody asked two persons and then we made a list of the improvement suggestions and highlighted which of them we wanted to implement, implement later or do not want to implement at all, because we did not like them or the API is not able to implement them.

User-tasks for our test-users contained simple processes and steps. First task, for example, was about general recipe search: "You opened this App and find yourself on the Home Screen. You want to find ceasar salad recipe. What will be your process? What do you think of it? Would you change/add something to this process?". The next one about preference modifications: "You want to exclude recipes containing diary products. Navigate to the preferences screen and do the necessary modifications". Some longer tasks about retrieving saved recipe looked like this: "You really

loved the recipe of a Cesar salad that you found earlier. Navigate to it again and add it to your Saved recipes. Once completed, you realize you have forgotten the quantity of chicken that you need to prepare it. For a quicker access, find the recipe from the dashboard screen". Mainly this tasks asked out of user to do something simple like opening some recipe or saving one, without iterating between a lot of screens.

After each task we measured how much time did it take to finish it and also asked 3-4 similar questions like:

- What is your impression of this screen?
- How would you rate this process?
- Would you use this screen?
- Would change anything on this screen?

As the result of this evaluation we highlighted for ourselves a few ways how we could improve our app:

- fixing the customize meal screen where the elements were displaced a bit
- fixing the max kcal amount to being bigger than 100
- adding save button on the Intolerance Screen
- adding functionality of removing saved recipes with a swipe

6 REFLECTION

Team projects especially in Corona-times are really hard and it's not so easy to organize everything online. It is often hard to get the result we expected to do, because everyone coded their part for themselves and it is difficult to adjust your own code to the code someone else wrote.

6.1 Separation of Tasks

For **Milestone 1**:

- Lukas Pezzei: Analysis of competing products.
- Julian Saria: User and Context Analysis.
- Vladislav Mazurov: Analysis of literature and Projectmanagement part.
- Rosi-Eliz Dzhurkova: Creating Personas.
- Alltogether: Analysing use-cases.

For **Milestone 2** each of the team-member made one app prototype which brings 4 prototypes for this milestone in total.

For **Milestone 3**:

- Lukas Pezzei: Customize Meals, Meal Type Screens.
- Julian Saria: Customize Meals, Meal Type Screens, Screenshots und Beschreibung des Prototyp hinsichtlich Funktionen, Inhalte und Interaktionen.
- Vladislav Mazurov: Preferences, Saved Recipes, Recipes Search Screens.
- Rosi-Eliz Dzhurkova: Splash Screen, Dashboard, Add/Search Ingredients, My intolerances Screen, Recipes Overview, Recipe Detailed Overview, Technical Backbone of the App.

For **Milestone 4** each of the team-member took 2 test users and interviewed them. Additionally:

- Lukas Pezzei: Evaluation, Reflection, Conclusions and future work parts of the final report, fixing App bugs, adding improvements based on the feedback.

- Julian Saria: Evaluation, Reflection, Conclusions and future work parts of the final report, screenshots and description of the prototype functions report.
- Vladislav Mazurov: Motivation, Related Work, Evaluation Parts of the final report, Abschlussbericht Latex document, fixing App bugs, adding improvements based on the feedback.
- Rosi-Eliz Dzhurkova: Design, Implementation Parts of the final report, fixing App bugs, adding improvements based on the feedback.

7 CONCLUSIONS AND FUTURE WORK

At the end of our work on this project we could summarize following strengths and weaknesses.

Strengths:

- Simple, intuitive design
- filtering does now work properly on every screen
- regarding to our interviews easy to use

Weaknesses:

- sometimes App does load very long
- filters are not compatible with each other, filter from customize meal doesn't take into account the ingredients. The ingredients filter works on his own, but doesn't work with the customize meal filter. The intolerances filter also does not take other filters into account.
- there are no further app settings other than intolerances
- there is no pantry

As to our next steps in regard to development this app, we could implement certain additional features for different parts of the app.

Recipe Details Screen:

- Description text with no html-Tags.
- Information about grams and portions
- all ingredients with grams in one list instead of swiping

Customize Meal Screen:

- Calories should change together with Nutrients, since they depend on each other. (Not enough knowledge about the exact dependencies)
- Implementing additional filter for equipment
- Bar for saturated and unsaturated fats

Intolerances Screen:

- Instead of having the intolerances Screen on the settings screen, we might add app settings to that as well.

Saved Recipes Screen:

- sort by most used recipes shown first
- rating of recipes
- bug fixes that might still occur

REFERENCES

- [1] Accenture Strategy [n.d.]. *The Future Of Food: New Realities For The Industry*. Retrieved June 07, 2021 from https://www.accenture.com/us-en/_acnmedia/pdf-70/accenture-future-of-food-new-realities-for-the-industry.pdf
- [2] Charles Edward Heath Allison E. Doub, Aron Levin and Kristie LeVangie. 2015. Mobile app-etite: Consumer attitudes towards and use of mobile technology in the context of eating behaviour. *Journal of Direct, Data and Digital Marketing Practice* (July 2015). <https://link.springer.com/content/pdf/10.1057/ddmp.2015.44.pdf>

- [3] Yi Chen Amy M. Lando, Michael C. Bazaco. 2017. Consumers' Use of Personal Electronic Devices in the Kitchen. *J Food Prot* (April 2017). <https://doi.org/10.4315/0362-028X.JFP-17-172>
- [4] Jenny Cooper. 2015. *Cooking trends among millennials: Welcome to the digital kitchen*. Retrieved June 07, 2021 from <https://www.thinkwithgoogle.com/future-of-marketing/digital-transformation/cooking-trends-among-millennials/>
- [5] Andrew May Luis Carlos Rubino de Oliveira, Val Mitchell. 2013. Designing a Smart Phone App for Sustainable Cooking. *UbiComp '13 Adjunct* (Sept. 2013). <https://doi.org/10.1145/2494091.2497315>
- [6] Vlad Taran Marina Yalanska. [n.d.]. *Case Study: Perfect Recipes App. UX Design for Cooking and Shopping*. Retrieved June 07, 2021 from <https://blog.tubikstudio.com/case-study-recipes-app-ux-design/>