|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ratings** | 5=excellent | 4=good | 3=acceptable | 2=poor | 1=bad |

|  |  |  |  |
| --- | --- | --- | --- |
| **Heuristics** | **Rating** | **Description** | **Notes** |
| 1. **Visibility of system status** | 5 | The system should always keep users informed about what is going on, through appropriate feedback within reasonable time. | Feedback through system was helpful with checkmarks, being marked as signed up, and growing out of buttons |
| 1. **Match between system and the real world** | 5 | The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order. | System makes sense in terms of way it’s laid out and phrasing is straightforward |
| 1. **User control and freedom** | 4 | Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo. | Scroll functionality not developed so prototype is flat, but that is a chance to come. On recipe page no way to x out or press back that communicates users they can close |
| 1. **Consistency and standards** | 4 | Users should not have to wonder whether different words, situations, or actions mean the same thing. | Again a back button or consistent ways feedback is given (x, check, greying out) |
| 1. **Error prevention** | 5 | Even better than good error messages are a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action. | n/a |
| 1. **Recognition rather than recall** | 5 | Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate. | App keeps you in the know and communicates what you’ve done back to you |
| 1. **Flexibility and efficiency of use** | 5 | Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions. | Layout and Vocab streamlines steps, but a button to go back as you go would make the flow clearer when otherwise you’d be stuck at the end of some screens |
| 1. **Aesthetic and minimalist design** | 5 | Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. | Communication is clear and consistent |
| 1. **Help users recognize, diagnose, and recover from errors** | n/a | Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution. | No errors encountered |
| 1. **Help and documentation** | 5 | Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large. | Like that everything is explained and clearly laid out as you go so you’re not questioning your actions |
| 1. **Promotes communal interaction** | 5 | Specific to this application, users should be led to actions which promote communal interaction and collective support | Groups and messenger when implemented seem promising, and the meal share concept itself is the quintessential community promotion event |
| 1. **Provides consummation to each act** | 5 | Users should feel satisfaction with each macro and micro interaction. From pressing a button, navigating through menus, or completing a specific task, the application should communicate a sense of finality and feedback for each step. | Getting Signed up or being told you’re hosting lets you know what’s going on right away, as well as downloading the recipes.  Clicking on buttons is simple and does what it expects. The lack of filler makes the app clear and to the point |
| **Overall Average** | 4.8 | | |

**Other Notes**

* Enjoy feedback of check marks and darkening buttons.
* Things are where you expect and the flow is good
* Back buttons or ways to escape would be much more usable to app novices
* The aspects and pages provided are clear about their action and descriptions help to streamline actions and inform users as they go
* Did a good job with usability and accessibility, but could use more feedback for people not familiar with navigating apps

|  |  |
| --- | --- |
| **Recommendation** | **Level of Impact** |
| 1. Standardize way in which app gives feedback | medium |
| 1. Create back buttons or X in order for people to know how to escape or undo their actions | medium |
| 1. Create Home page to show you what you have done, recipes you have, and events you’re going to/ hosting, so all your actions are communicated back to you | high |

**Impacts**

High = may prevent the user from completing a task or accessing information

Medium = might cause the user difficulty but the task could be completed

Low = minor problems that would not significantly affect task completion