

# Restaurant Quick Start Design Document

By Will Hofkamp

[willhofkamp.com/rqs](http://willhofkamp.com/rqs)

## Table of Contents

1. Introduction
2. Basic Design Aspects
3. Landing Section Design
4. Restaurant Section Design
5. Menu Section Design
6. Staff Section Design
7. What could be improved on

## Introduction

In order to make Square products and services more accessible for new Square users and restaurant owners, the Restaurant Quick Start will give them a head start by quickly setting up and generating the necessities for running a restaurant. The ease of use of Square's services and graceful transition would likely increase brand loyalty and public company perception by empowering anyone to start a business with a Square account. The faster and easier the user is intertwined with Square's services, the less likely they will go with competing products and services.

## Basic Design Aspects

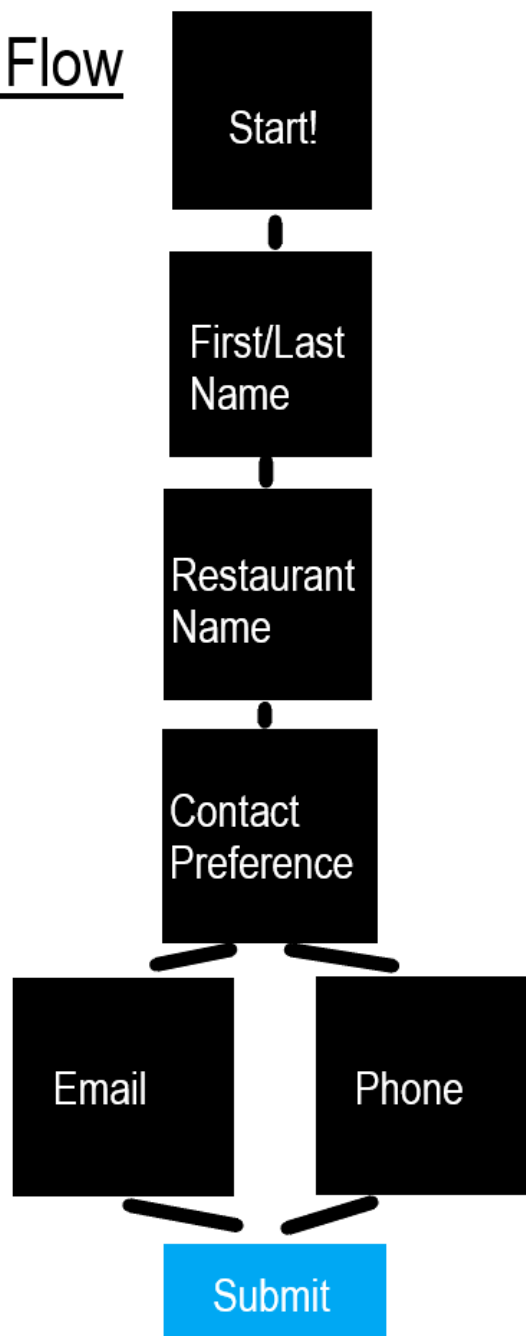
The Restaurant Quick Start can be easily scalable to eventually fit all types of businesses with some minor tweaks. The performance will be the toughest part to optimize because there could be multiple API calls that are doing a lot in the back end. With each call, there is a chance for failure. The availability should not be worried much about though, as users can just continue on and fill in the information later. The form will be locally cached after each step and cached in a cdn after each section submission to help prevent multiple API calls that are especially time consuming such as uploading photos or generating menu items, so that the user can come back to where they were at any time. This data can then be tracked and analyzed, and can be used to determine how to update the flow based on what users are and aren't filling out.

## Landing Section Design

The initial questions establish a relationship with the customer through conversational questions that set up an account. By gathering a small sample of simple base information first, the user commits to going through the Restaurant Quick Start and will be more likely to push through to fill out the entire form. The conversational tone in the questions make the process more personable, and the restaurant name input is reflected in the following questions about the restaurant to further personalization and increase user experience.

API Calls: Public call to load the page, private call to submit the form and create an account and/or restaurant

### Landing Flow



## Restaurant Section Design

This section is by far the most important and most complicated, because it is all about the restaurant. The majority of the flow is straightforward and linear. The required fields must be filled in and can't be skipped over. What can be skipped over are the three steps with potential API calls, the restaurant logo upload, color scheme, and seating chart. The logo upload is simply uploading the user's logo to Square. The color scheme section is the most unfinished and can be skipped over by simply saying "Looks Good" and selecting no colors from the color pickers, which sets all white as the default theme. The seating chart has the most options out of all choices in the form. The user can either upload pictures of their seating chart from their device, make the seating chart then and there, or skip over it entirely. The user then has the option to link their website or have a Square Online site generated for them, or to just ignore it and continue on in the flow.

### Restaurant Location

API Calls: Third Party API call for autofill address

### Restaurant Logo Upload

API Calls: Private call to upload the photo

System Calls: User will use OS's option to select the photos they'd like to upload

### Restaurant Color Scheme

API Calls: private call to send the partially completed form data, returns list of generated theme colors

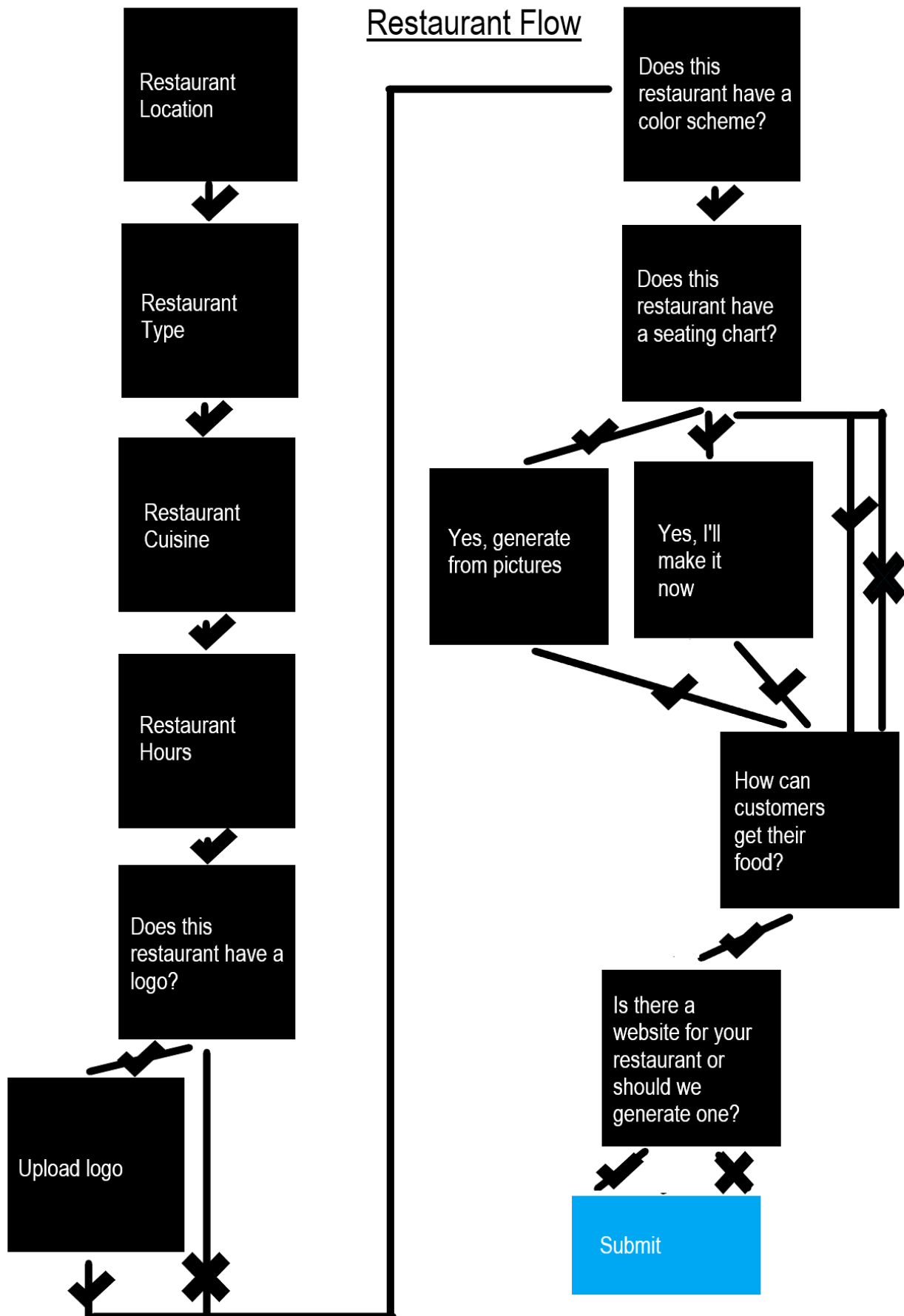
### Restaurant Seating Chart

API Calls: Private call to upload the photo and returns the generated seating charts, private call to bring up the seating chart maker

System Calls: User will use OS's option to select the photos they'd like to upload

### Submit form

API Calls: Private call to system to save data and generate website if option is selected



## Menu Section Design

This flow sets up a skeleton menu for the restaurant. It starts by asking for any third-party apps so that as many values can be prefilled as possible before the user manually edits the menu. The other option is uploading photos of physical menus from the user's device, and with machine learning, we can create a photo analysis method that will analyze each photo that is uploaded and determine each menu item, then sort the items to the correct menu and display group. The user can then manually add any simple menu items. Ideally, the form would be more like what exists when adding a menu item today. This is all optional and can quickly be skipped over.

### Third-Party Question/Integrations

API Calls: Third party call to login to services, private call to link the third party account to the user account

### Upload user menu photos and generate items

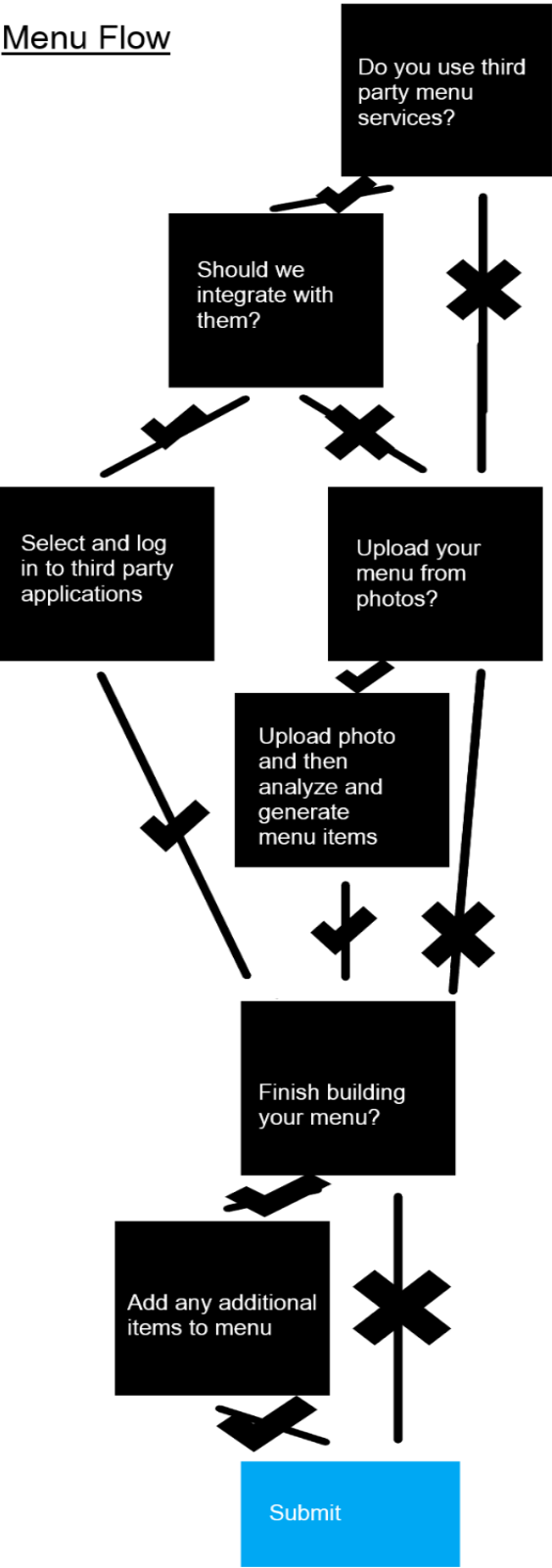
API Calls: Private call to upload the photos, returns list of menu items added

System Calls: User will use OS's option to select the photos of their menu they'd like to upload

### Finish building out the menu/section submit

API Calls: Private call to save the final menu

Menu Flow



## Staff Section Design

This is the simplest section of the Quick Start flow, as it can be instantly skipped over since there's only one section. Here the user can add simple employees to their Square account. The only data currently collected is name and job, and everyone will be set to "Pending" status because their invitation is pending. This table can be expanded so that the user can add more information similar to normally adding the employee.

API Call: Private call to save the new employees

## Summary

Thanks for checking out my project! It was very fun to do and I hope I can implement it at Square someday. I learned a lot about all the cool things that Square has and how many useful services that have been implemented already. Square has the opportunity to shake things up and I'd love to help out with it.