

# Project Style Guide

---

**Product Name:** DishDelish

**Members:** Audrey Ostrom, Alyssa Yee, Annika Gupta, Janvi Rochlani, Alisha Saboowala, Maina Dhar

**Revision Number:** 1

**Revision Date:** 3/08/2024

---

## Variable Naming

For this project, we used camelcase naming for variable names (i.e. "tokenString").

## CSS Selector Naming

For this project, we used hyphenated lower-case naming for classes and ids for styling (i.e. "settings-bar").

## Function Naming

For this project, we used camelcase naming and made sure to have appropriate doc strings in the following format.

```
/**
 * The function `saveIngredients` saves a user's ingredients in MongoDB.
 * @param {String} email - specifies the user (used to query docs)
 * @param {Array} ingredients - Ingredients is an array that contains
 * the list of ingredients that the user wants to save.
 * @return {Object} The `saveIngredients` function is returning a JSON
 * response with a success message if the operation was successful,
 * and an error message if an error occurred.
 */
export async function saveIngredients(email, ingredients) {
  try {
```

## Code Readability Assurance

We made sure to lint our code to ensure readability was maintained within our codebase. We kept most of the default settings: meaning no double quotes, missing semi-colons, trailing spaces, blank padding for us.

We specifically formatted using prettier (``npx prettier -write <file name>``) and eslint (``npm run lint``). We followed the Google standard, excluding some key exceptions we'll talk about here.

*A screenshot of our .eslintrc.json file in our repository.*

A screenshot of a code editor showing the contents of a file named .eslintrc.json. The file is a JSON object with the following structure: { "extends": ["next/core-web-vitals", "google"], "rules": { "react/no-unescaped-entities": 0, "no-tabs": 0, // don't mess with prettier "new-cap": 0, // in auth.js, doesn't recognize built-in Next functions "strict": 2, "indent": ["error", "tab"] } }. The code is syntax-highlighted, with strings in orange, comments in green, and keywords in blue. Line numbers 1 through 11 are visible on the left side of the editor.

```
1  {
2    "extends": ["next/core-web-vitals", "google"],
3    "rules": {
4      "react/no-unescaped-entities": 0,
5      "no-tabs": 0, // don't mess with prettier
6      "new-cap": 0, // in auth.js, doesn't recognize built-in Next functions
7      "strict": 2,
8      "indent": ["error", "tab"]
9    }
10 }
11
```

We silenced warnings for “react/no-unescaped-entities”, because it was trying to flag us for `useEffect()` hooks when it was not applicable. Similarly “new-cap” was silenced because it was flagging built-in NextJS functions that were being imported. “no-tabs” was silenced so that prettier and eslint can be run in tandem.

Thankfully, our repo has no errors and is relatively clean!