Gbiv Project Proposal

Release 1.0

Dux D-zine

Nov 09, 2022

CONTENTS:

CHAPTER

ONE

CONCEPT OF OPERATIONS (BASIC REQUIREMENTS)

1.1 Current System

There are generally two types of websites that exist today with functionality similar to our application. On one hand, there are sites such as Canva and Adobe Color Extractor which take user uploaded photos and generates a palette that matches the colors in the picture. Alternatively, there are sites like Coolors and MyColorSpace that allow the user to choose a hex color and view color palettes that have that hex color in them.

1.2 Justification For New System

Design and color theory are everywhere from web development to fashion to interior design and architecture. No existing site has the functionality of our application which provides a powerful tool for applying color theory to the real world. We often have physical items such as paint, clothes, or furniture where we cannot simply check the hex code of the dominant color. Our application combines this analysis of the color in terms of hex along with recommendations for colors to combine it with based on color theory.

1.3 Operational Features of Proposed System

The operational features of this proposed system will center around what happens after a user has uploaded an image. First a dominant color will be extracted from the photo and displayed to the user. In addition to this, the site will generate several colors that are related to the dominant color by way of color theory. The final output will be a collection of palettes that have the dominant color in it or are in some way adjacent to the color.

1.4 User Classes

Our application will have a single user class which will be designers of all skill levels who are looking to apply color theory in their design choices. This could be users looking for specific color relationships to something they already have (e.g. a color that complements their green wall) or users who are looking for palettes to inspire overall color schemes. The beauty of our application is that it appeals to professional designers by providing a crucial service in a convenient way, as well as the designer in the everyday person by introducing color theory and providing a fun way to experiment with colors.

1.5 Modes of Operation

There will be a single mode of operation for all regular users of our application. This mode of operation is entered when they visit our site and end up on the main page where photos can be uploaded. On top of this, our team as developers will have a mode of operation where we can upload new palettes and tweak the code for new releases.

1.6 Operational Scenarios

1.6.1 Scenario 1

Uploading an Image

The user who wishes to extract colors from something tangible in the world, such as a piece of clothing, a building, nature, etc. would want to utilize the Image Upload feature. Upon uploading their selected image, our program will compute a single hex color that occurs most in the image, and captures the general essence of the image. Using this color, our program will then compile a list of color palettes, and another list of related palettes that might not necessarily use the initial color, that the user can view.

1.6.2 Scenario 2

Selecting Related Colors and/or Palettes

Any user who wishes to browse through our page of palettes can view a main list of palettes as an archive, or they may enter or select a specific color, and view palettes that work with that color, as well as related palettes.

CHAPTER

TWO

BASIC DESIGN OVERVIEW

2.1 Technologies Used

Backend

- MongoDB
- Python
  - ColorThief

Frontend

- HTML
- CSS
- Javascript
- Bootstrap

Entire System

- Flask
- Firebase
- Jira
- GitHub

Documentation

- reStructuredText/Sphinx
- Microsoft Visio
- Google Drive

2.2 Basic Software Architecture

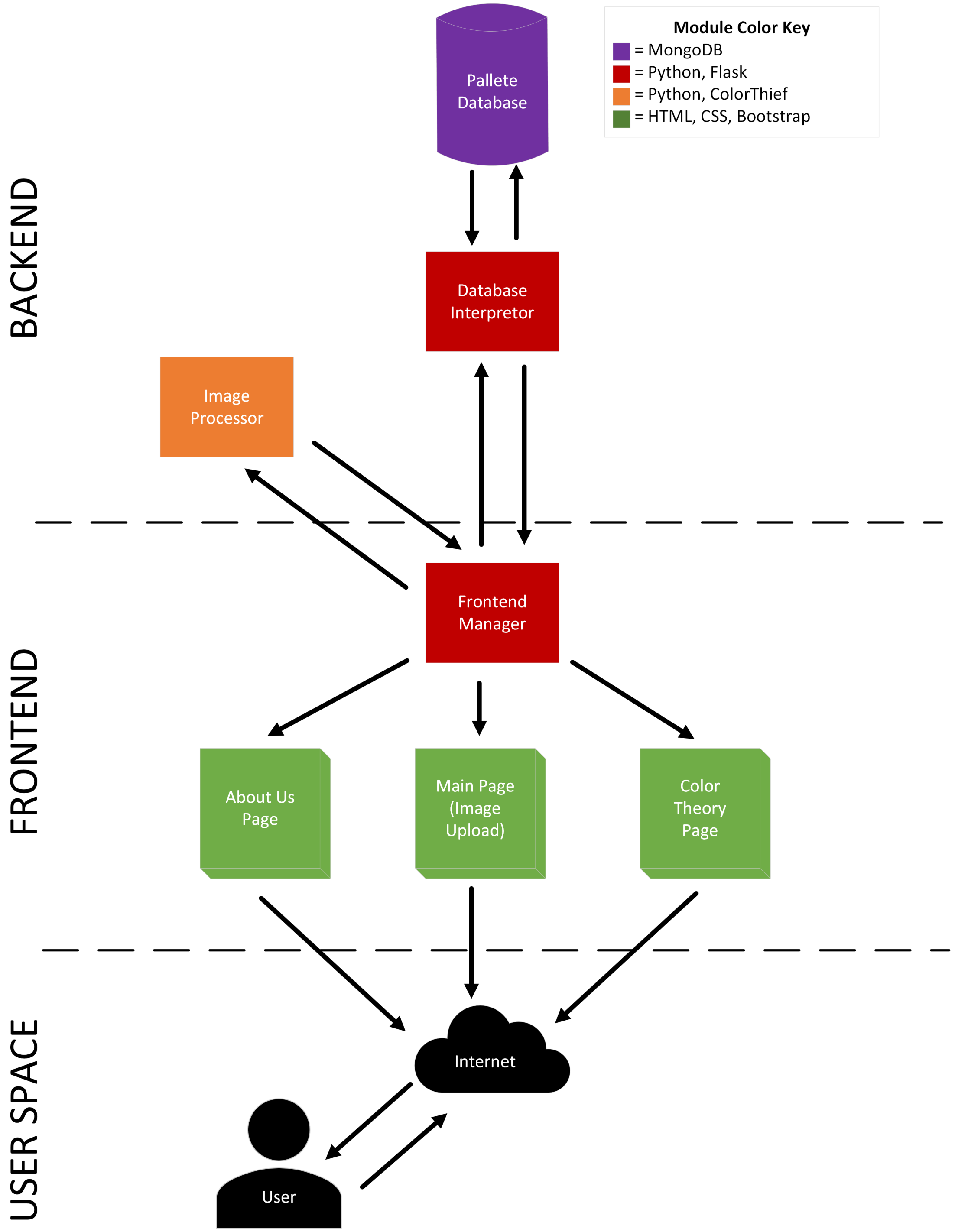

Fig. 2.1: Software Architecture (Version 1)

CHAPTER

THREE

PROJECT PLAN

3.1 Division of Labor

Team Member	Primary Responsibilities	Secondary Responsibilities
Sam Heilenbach	Database Management	
Cheyanne Kester	Frontend Development	Flask Frontend Interface
Ian McConachie	Documentation	Bootstrap Styling
Scott Wallace	System Design	Jira Management
Austin Warren	Image processing	

3.2 Rough Timeline of Work

Week 1 (Nov 7-13)

(Frontend and Backend): Create project with placeholder/dummy functions and features; integrate frontend and backend before major development; get web hosting set up.

Week 2 (Nov 14-20)

Frontend: Get placeholder sections on the web page configured; start making functions to handle backend requests; if web hosting is not done yet, get it done.

Backend: 1: Crete modules for simple extraction of dominant colors as well as finding complementary colors given a color. 2: Establish the structure of databases/collections in MongoDB and begin establishing methods for querying palettes based on other colors.

Week 3 (Nov 21-27)

Frontend: Design the web page sections cohesively; perhaps begin testing integration.

Backend: 1: Implement palette querying, potentially reconfigure database/collection hierarchy to optimize query times and facilitate the most common searches between palettes. 2: Start producing the filtering methods necessary to provide the color palette selection functionality. 3: Flesh out color extraction modules and fully connect them to the database (s•t• colors are extracted from pictures and uploaded to the correct collection).

Week 4 (Nov 28-Dec 2)

Frontend: Polish web page sections; ensure that all functions are working as expected.

Backend: Finalize filtering methods necessary to allow for users to look for palettes related manually (i.e., without uploading any images).