

# **Filter, Search, and Customize: A Guide to Image Display in Your App**

**By - Bishal Dalai**

- **Introduction**
- **Filter Functionality**
- **Search Functionality**
- **Customizable Layout**
- **Conclusion**

## Introduction

In this presentation, we will discuss the implementation of a filter and search functionality in our app that allows users to easily find and customize images based on their preferences. We will also explore the challenges we faced during the development process and the solutions we implemented to overcome them.

Our goal is to provide a seamless user experience by allowing users to filter images by categories, search for specific keywords, and customize the layout of the images.



## Filter Functionality

To implement the filter functionality, we used React Native and Redux as our tech stack. We created a filter component that displays a list of categories, such as animals, nature, and architecture. When a user selects a category, the UI updates to display only those images that belong to that category.

We faced some challenges while implementing the filter functionality, such as ensuring that the UI updates smoothly without any lag or delay. To solve this problem, we optimized our code and used asynchronous programming techniques to ensure that the UI updates in real-time.



## Search Functionality

To implement the search functionality, we used a custom search algorithm that utilizes fuzzy search and substring matching. This algorithm allows users to search for images based on keywords or phrases that appear in the image title or description.

We faced some challenges while implementing the search functionality, such as ensuring that the search results are accurate and relevant. To solve this problem, we fine-tuned our search algorithm and tested it extensively to ensure that it returns the most relevant results.



## Customizable Layout

To implement the customizable layout feature, we used React Native's flexbox layout system. We created a layout component that allows users to choose between a grid or carousel format, adjust the size and spacing of the images, and customize other layout options.

We faced some challenges while implementing the customizable layout feature, such as ensuring that the UI remains responsive and flexible regardless of the user's device or screen size. To solve this problem, we used responsive design techniques and tested our code on various devices and screen sizes.



## Conclusion

In conclusion, we have successfully implemented a filter and search functionality in our app that allows users to easily find and customize images based on their preferences. We utilized React Native and Redux as our tech stack and overcame various challenges during the development process.

Our app provides a seamless user experience by allowing users to filter images by categories, search for specific keywords, and customize the layout of the images. We are confident that our app will provide value to our users and enhance their overall experience.

