# Creating a Complete Custom Keyboard from Scratch

Elias Glauert

25 September 2024

# Contents

# 1 Introduction

This paper is about my journey of creating a custom-made keyboard from scratch. It started after I watched a video by Christian Selig[1], where he created one of these custom boards. Inspired by him, I embarked on this project. Custom keyboards can be costly, often reaching hundreds of euros or dollars, so bear this in mind if you wish to follow the same path.

# 2 Goals and Specifications for the Project

## 2.1 Important Notice for using this as a Guide

The objective of creating a custom keyboard is to have a personalized and enjoyable experience. While this document will detail my choices, they are based on my needs and preferences and may not suit others. Please use this guide as a reference, but feel free to adapt or improve upon my design.

## 2.2 What are my Goals and Specifications?

My main goal was to create a keyboard that made typing easier and more efficient. After learning to touch type on a Mac keyboard, I found that it was unoptimized for my needs. Issues such as excessive movement when pressing backspace, underutilized thumbs, and fingers overlapping in function inspired this project.

### 2.2.1 List of all the Goals

1. The board fits my hands so that no hand movement is necessary while typing.
2. Layout toggles between Windows and MacOS.
3. The keyboard is split into two parts for ergonomic adjustment.
4. The connection is wireless (Bluetooth).
5. No backlighting to save battery life.
6. The switches are silent to avoid disturbing others.
7. The design is simple, sleek, and fits my room's style and color scheme.

[1] `https://www.youtube.com/watch?v=7UXsD7nSfDY`

# 3 Designing the Keyboard Layout

## 3.1 Getting the Layout Right and Personalized

To achieve ergonomic goals, I placed my fingers on a 5mm checkered sheet of paper and drew a 2x2 square around each finger. This served as the base for key placement. Adjustments were made based on finger length and movement, inspired by the Corne layout.

## 3.2 Ergogen – The Software Used to Create the Layout

Ergogen is a free, open-source tool for generating keyboard layout files. By describing the key positions, Ergogen creates a customizable layout. Detailed documentation and tutorials can be found online[2].

# 4 Choosing the Microcontroller

## 4.1 What is a Microcontroller?

A microcontroller is like the brain of the keyboard, converting key presses into signals that the computer understands. For this project, I chose the `nice!nano`[3] microcontroller because of its Bluetooth capabilities, which align with the project goals.

# 5 The PCB Plate and Routing

The PCB plate connects all the keyboard components like the controller, battery, switches, and more. I used KiCad for routing the electrical signals.

# 6 Switches and Keycaps

For switches, I chose the Gazzew Boba U4 Silents[4], which are tactile and designed to be quiet. For keycaps, I opted for YMDK DSA Profile blank caps.

[2] `https://ergogen.xyz`
[3] `https://nicekeyboards.com/nice-nano`
[4] `https://thocstock.com/switches/gazzew-boba-u4-silents`

# 7 Soldering and Assembly

The PCB, components, and case are soldered and assembled according to the custom design. Lead-free solder was used for safety, and proper ventilation is essential during the process.

# 8 Firmware and Keymaps

## 8.1 Creating the Keymap for Windows and Mac

I designed the keymap using ZMK[5] firmware to switch between Windows and MacOS layouts. The firmware was loaded onto the controller, and Bluetooth was used for connection.

# 9 Conclusion

This document outlines my process for creating a custom keyboard. The steps and resources are provided as a guide to help others with similar projects. For detailed code and files, visit my GitHub repository[6].

[5] https://zmk.dev

[6] https://github.com/lultoni/pengo-keyboard