**1. Decide the MVP Order**

**a. BLE Receiver on Device (KeyCatcher CircuitPython BLE)**

* You need a working “BLE mailbox” on the hardware first.
  + CircuitPython or Arduino: KeyCatcher listens for text from BLE central.
* Once this works, *any* sender (phone, PC, etc.) can connect.

**b. Phone App/Workflow**

* **Start with a “Flow 1” screen:**
  + Large textbox (“Paste or type text here”)
  + “Send” button
* **Build a basic BLE sender app in MAUI (since you know it well).**
  + Use Plugin.BLE, connect to KeyCatcher, write text to characteristic.

**c. UI/UX Planning**

* Map out the “pairing walk-through” screens (step-by-step, with icons and clear prompts).
* Flow 2: Add “Share to KeyCatcher” Android/iOS action.
* Settings page for “auto-send” or “edit before send.”

**2. Paring Process (Pairing UX)**

* **Onboarding “wizard”:**
  1. Plug in KeyCatcher to computer (“You’ll see a blinking light”)
  2. Enable Bluetooth on your phone
  3. App scans, finds “KeyCatcher-####”
  4. Tap to pair (or enter PIN if needed)
  5. Show “Ready to send text!” page
* **Show real-time status:**
  1. “Connecting…”
  2. “Paired!”
  3. “Error—try again”
* **Show “typed text will appear on your computer” tip!**

**3. Workflow Pages**

**Flow 1: *Basic Text Send***

* Paste/type text → Tap “Send” → KeyCatcher types it into computer.
* **Future:** Add “Capture from Camera/OCR” button.

**Flow 2: *Share Pipeline***

* Register “Share to KeyCatcher” as a share target (Android/iOS, or browser extension)
* User can “share” from any app (browser, email, ChatGPT app, etc.) directly to KeyCatcher.
* App receives text, either:
  + **Sends immediately** (auto-send)
  + **Shows edit/review page**, then sends on confirmation

**Settings Page**

* Toggle “Auto-send” (skip review)
* Default action (send vs. edit)
* Manage paired devices

**4. UI/UX Next Steps**

* **Sketch wireframes** for each page/step (can be on paper or with Figma/Whiteboard/Miro).
* Build out **basic navigation** and “happy path” flow in MAUI Shell.

**5. Tech Order Recommendation**

1. **Start BLE receiver on KeyCatcher hardware** (CircuitPython or Arduino)
   * Confirm you can connect and send text from BLE central.
2. **Build the “Flow 1” MAUI app page** (textbox + send)
   * Focus on connection flow and sending simple text.
3. **Add pairing walk-through screens**
4. **Prototype “Share” integration**
5. **Iterate on settings and feedback UX**

**TL;DR:**

* **Start with the BLE mailbox on hardware**—the rest of the app is “just UX” on top of a working connection.
* **MAUI app first page:** simple, clean “type or paste, then send” UI.
* **Pairing and onboarding flow** is *crucial* for non-techies—make it easy.
* **Share pipeline and OCR** are killer features, but come after basics are proven.
* KeyCatcher: KeyCatcher: Universal Phone-to-PC Accessory

## Use Cases

For the gamer to do typing rather than using their controller

* For the person that wants to take picture and type it into their computer
* For the office worker that wants to use ai help but is locked out
* For the techie that wants to use LLM results at work but doesn’t have access

# Icons and Share To

**That Dr. Seuss–style “net catching letters into a keyboard” idea is absolute gold!**  
It’s playful, visually unique, and totally fits the “KeyCatcher” brand—instantly communicating “fun, clever, and useful.”  
Think: whimsical net, big swoopy letters flying out of a phone, being “caught” into a happy keyboard, maybe even some playful motion lines.

### ****Icon Concept: Dr. Seuss–Style KeyCatcher****

* **Elements:**
  + Big, bouncy letters (maybe a few colorful “A, B, C”) flying from a cartoon phone
  + A curvy net, exaggerated and playful, reaching out to “catch” them
  + The “net” aims down toward a chunky, happy keyboard or a big key with a face
  + Seussian swirls, maybe a little star or sparkle for whimsy
  + Colors: pastel blues, reds, greens, and white for fun & contrast
* **Possible Layout:**
  + **App Icon:**
    - Simplified: a cartoon net (maybe with a motion swoosh), a couple of floating letters, and a key or keyboard beneath.
  + **Splash Screen or Marketing Art:**
    - Full scene: Phone on left, letters flying, net catching, keyboard on right.
  + **Send Button:**
    - A small net icon catching a letter, or a key with “send” lines.

### ****Ready for “Share To” Tonight****

* I’ll be prepped with **MAUI “Share To” implementation** options (Android-first, iOS with extension path), and a fast flow for integrating shared text into your app for BLE sending.
* We’ll also sketch how the “caught letters” can animate from share-sheet to the send screen for maximum delight.

## ****What You’ll Get Later:****

* **A couple of Dr. Seuss–inspired icon sketches** (SVG/PNG, ready for Figma or quick feedback)
* **Step-by-step for “Share To KeyCatcher” in MAUI** with example code