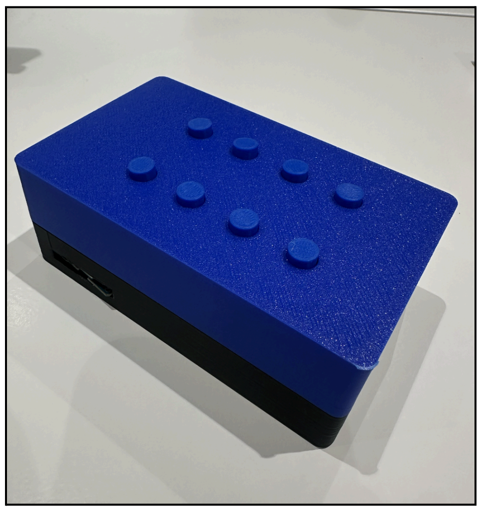
KoboClicker

Programmable Bluetooth-Enabled Counter for Field Data Collection



Tomas Arroyo, Audrey Cheng, Justin Curcio, Roaa Elfishawy, Lorenzo Funk, Guzalkhon Khaydarova, Nate Marinaccio, Kieran McDaniel, Lyat Melese, Maria Pape, Raul Robles, Guadalupe Vazquez, Jade Webb, Norene Williams, Anya Zhang

ENG-SCI 96: Engineering Problem Solving and Design Project

December 15, 2025

Table of Contents

I. Introduction

Engineering Sciences 96: Engineering Problem Solving and Design Project (ES96) is a semester-long course taken by Harvard engineering students across multiple disciplines in their junior year. At the start of each semester, students are presented with a unique, real-world problem posed by an external client. In Fall 2025, ES96 partnered with the Harvard Humanitarian Initiative (HHI), a university-wide research and education center that advances evidence-based approaches to humanitarian response in disasters and crises worldwide. Founded in 2005, HHI integrates research, policy, and training to improve the effectiveness, ethics, and impact of humanitarian action. Through close engagement with HHI, ES96 contributes a technical perspective that aligns with HHI's interdisciplinary approach, culminating in the development of a technical solution to an identified challenge.

At the beginning of the Fall 2025 semester, the class met with key members of the HHI team, including Executive Director Dr. Irini Albanti, Director Dr. Michael VanRooyen, Dr. Phuong Pham, Dr. Patrick Vinck, and other organizational leads, to gain a deeper understanding of HHI's mission and operations. The class then collectively progressed through the phases of investigation, ideation, design, and testing, working toward a final engineering solution defined by a collaboratively developed problem statement. During the investigation phase, students analyzed HHI's core operations by synthesizing insights from client meetings and site visits to identify organizational needs and constraints. This work informed the problem statement that guided the ideation phase, during which the class researched, brainstormed, and evaluated potential solutions. Given the scope of the selected project, the design and testing phase was divided into mechanical, electrical, and software subteams, each responsible for specific technical components. After integrating these subsystems into a unified prototype, the class presented the finalized design to HHI. This report serves as a comprehensive record of the work completed during the semester and outlines recommended next steps for the client.

II. Project Abstract

Currently, the collection of data for rapid needs assessments in the humanitarian crisis response space is commonly done through manual means—such as using a pencil and paper or through a manual clicker—because of the lack of cellular service on the field as well as the possibility of theft if an expensive device is used for recording data. Therefore, the uploading of data must be done off-site to a platform such as Kobotoolbox, which is a software platform developed by HHI that is widely used for humanitarian data collection and management across the world. Therefore, in tackling this problem, we hoped to create a more versatile and robust mechanism for collecting data.

Our project, KoboClicker, is a customizable, Bluetooth-enabled device that assists in field data collection, specifically within the context of rapid needs assessments and flow monitoring cases. This clicker mechanism allows for a wider range of information to be collected, such as demographics data, and provides a secure and automatic method for uploading data. Therefore, the users and organizations using KoboClicker can easily stay informed on the status of both the crisis and its responses.

As for the design, KoboClicker consists of a handheld clicker shell that houses the electrical components integral to the functionality, including a breadboard, rechargeable battery, and microcontroller. The clicker's input interface consists of a power, submit, and clear button, leaving five other buttons to be utilized for various categories of data input. The user can submit individual data entries, which is then written to the microSD as a CSV file, and also sent via

Bluetooth to an Android application. The integrity of the mechanical, electrical, and software components were tested and iterated multiple times to meet the design specifications and the cohesive integration of the full clicker mechanism. Beyond the working prototype, next steps for the project are translating the clicker shell to a more ergonomic design—which can be either purchased or manufactured, fabricating a custom PCB, and integrating the software application with KoboCollect.

III. Overview of the Humanitarian Aid Field

Definition & Scope

Humanitarian aid is the organized provision of assistance to populations facing crises that overwhelm local or national capacities [2]. These crises may include armed conflict, natural disasters, or public health emergencies that necessitate an international response. A key distinction must be made between relief and development. Relief refers to immediate, short-term responses—such as the provision of shelter, food, and medical care—while development seeks long-term systemic change through institution-building, resilience, and capacity strengthening [3]. Humanitarian aid, as a whole, is focused on relief efforts rather than development, emphasizing acute, life-saving responses over long-term social transformation.

Within this relief-oriented framework, humanitarian aid encompasses several core domains. These include advocacy, which elevates the needs and rights of affected populations; emergency relief assistance that provides immediate life-saving support; food security programs that ensure access to adequate nutrition; refugee and displacement services offering protection and resettlement support; and healthcare interventions ranging from emergency medical care to broader public health responses [5].

Humanitarian action must also be distinguished from military assistance. While military actors may at times contribute logistical or security support, their involvement is often shaped by political or strategic priorities. In contrast, humanitarian aid adheres to the principles of neutrality, impartiality, and independence, ensuring that assistance is directed solely based on need [4].

Approaches and Coordination in Humanitarian Aid

Humanitarian aid is delivered through a range of approaches that reflect both the immediacy of crisis response and the longer trajectory of recovery. Emergency relief focuses on immediate, life-saving assistance, including food, water, shelter, and medical care. As conditions stabilize, early recovery efforts aim to restore essential services, rebuild basic infrastructure, and support livelihoods. These approaches are increasingly understood as interconnected rather than sequential, a perspective often described as the humanitarian-development nexus, which emphasizes coordination among crisis response, recovery, and longer-term resilience-building [6].

Effective humanitarian action also depends on coordination among diverse actors operating at international, national, and local levels. International organizations contribute financial resources, technical expertise, and global operational capacity, while local organizations offer contextual knowledge, cultural understanding, and established trust within affected communities. In recent years, the concept of localization has gained prominence, emphasizing the importance of shifting decision-making power and resources toward local actors to improve responsiveness and sustainability [8].

Several formal coordination mechanisms exist to organize these actors and approaches in practice. One prominent example is the United Nations Cluster System, which groups humanitarian organizations by sector, such as health, shelter, education, and logistics. Each cluster operates under a designated lead agency responsible for facilitating coordination, information sharing, and standard setting. For example, the health cluster is led by the World Health Organization (WHO) and works alongside national governments, humanitarian partners, and coordinating bodies such as the Humanitarian Coordinator and the Office for the Coordination of Humanitarian Affairs (OCHA) [9]. Donor coordination platforms and cross-sector partnerships further support alignment of funding, expertise, and operational priorities. Together, these mechanisms provide a structured framework for coordinating humanitarian approaches and delivering assistance effectively across complex crisis environments.

Key Actors in Humanitarian Response

Building on the coordination mechanisms described above, humanitarian response is carried out by a diverse set of actors operating at international, national, and local levels, each contributing distinct capacities and responsibilities. The United Nations (UN) system plays a central coordinating role during large-scale humanitarian emergencies, with the OCHA responsible for overseeing overall response coordination, information sharing, and liaison among governments, United Nations agencies, and non-governmental organizations (NGOs) [10]. Through the Global Cluster System, United Nations agencies are designated as leads or co-leads for sector-specific responses, such as health, food security, shelter, and logistics, to provide technical guidance and facilitate coordination among humanitarian partners [9]. A summary of global cluster leadership arrangements is provided in Appendix B.

Within this framework, agencies including the WHO, the United Nations Children's Fund (UNICEF), the World Food Programme (WFP), the United Nations High Commissioner for

Refugees (UNHCR), the International Organization for Migration (IOM), the Food and Agriculture Organization of the United Nations (FAO), and the United Nations Development Programme (UNDP) assume leadership roles across clusters according to their technical mandates. These agencies work alongside national authorities and humanitarian partners to set standards, coordinate activities, and support local response capacity, with leadership arrangements varying by sector and crisis context [11–15].

National governments retain primary responsibility under international law for the protection and assistance of populations affected by crises within their borders. They authorize humanitarian access, coordinate with international responders, and deploy domestic emergency and civil protection services. Donor governments support humanitarian operations by financing responses through multilateral agencies, the Red Cross and Red Crescent Movement, and NGOs, with funding mechanisms and earmarking decisions shaping response priorities and operational capacity [5,15].

Non-governmental organizations serve as essential frontline actors in humanitarian response. International non-governmental organizations (INGOs) contribute operational capacity, technical expertise, and rapid deployment capabilities, while local NGOs and civil society organizations (CSOs) provide contextual knowledge, trusted community relationships, and sustained engagement before, during, and after crises [16, 17]. Humanitarian organizations operate in complementary roles, with no single actor covering all domains of response. Medical organizations such as Médecins Sans Frontières (MSF) and the International Medical Corps (IMC) specialize in emergency and conflict medicine, while organizations including Save the Children and World Vision focus on child protection, education, and food assistance. Other organizations, such as Oxfam, CARE International (CARE), Partners In Health (PIH), and the

International Rescue Committee (IRC), contribute expertise in water and sanitation, gender-focused programming, health systems strengthening, and refugee protection.

Military actors may provide limited support in humanitarian contexts, particularly during large-scale disasters that exceed civilian response capacity. Domestic militaries often assist with logistics, engineering, and search-and-rescue operations, while foreign militaries may contribute airlift or medical capabilities. Their involvement is typically constrained by political sensitivities and the need to uphold humanitarian principles of neutrality, impartiality, and independence [18, 19].

Principles of Humanitarian Aid

Humanitarian action is guided by a core set of principles, including humanity, impartiality, neutrality, and independence [20]. These principles are codified in United Nations General Assembly resolutions and serve as foundational norms for humanitarian organizations operating in complex crisis environments, particularly those involving armed conflict and political instability [21]. Together, they aim to ensure that humanitarian assistance prioritizes human well-being and is delivered solely based on need.

The principle of humanity emphasizes the prevention and alleviation of human suffering, guiding organizations to protect life, health, and dignity through activities such as emergency relief, education, research, and resource provision. Impartiality and neutrality reinforce this objective by requiring that aid be delivered without discrimination and without favoring any side in political, religious, or ideological conflicts. Independence further safeguards humanitarian action by ensuring that relief efforts remain autonomous from political, military, or economic agendas, allowing organizations to maintain credibility and trust among affected populations and stakeholders [22].

Despite their central role in humanitarian practice, these principles are increasingly challenged in contemporary crises. Humanitarian aid is often entangled with political objectives, and aid workers may be perceived as aligned with particular governments or parties to conflict. The growing politicization of humanitarian response has contributed to an erosion of perceived neutrality, increasing risks to humanitarian personnel, and restricting access to populations in need. In some contexts, violence against aid workers has been deliberately used as a tactic to undermine assistance viewed as benefiting opposing groups, highlighting the tension between humanitarian principles and the realities of modern conflict [23].

Challenges to Humanitarian Aid

Despite the existence of widely accepted humanitarian principles and coordination mechanisms, humanitarian organizations face persistent and evolving challenges that limit their ability to respond effectively to crises. These challenges span financial constraints, political interference, security risks, logistical barriers, and structural weaknesses within affected states.

Funding Constraints and Resource Gaps

One of the most significant challenges facing humanitarian aid is the growing gap between humanitarian needs and available funding. Humanitarian organizations rely heavily on donor financing to sustain operations, yet global funding levels have failed to keep pace with rising demand. In 2025, the United Nations estimated that $47.4 billion would be required to support humanitarian response operations worldwide [26]. Declines in available funding have forced organizations to scale back or terminate critical programs, leaving vulnerable populations without adequate assistance. Furthermore, recent shifts in donor landscapes have further exacerbated these constraints. The closure of the United States Agency for International Development (USAID) in July 2025 and its absorption into the U.S. Department of State marked a significant change in global humanitarian financing. Before its shutdown, USAID was the largest single investor in humanitarian data systems and operations, including famine early warning, displacement tracking, and public health surveillance [27]. Its withdrawal has created a substantial funding gap, as the United States has consistently been the largest donor to humanitarian efforts, contributing approximately $40 billion in 2023 [44]. Increasing competition for limited resources has also raised concerns that aid allocation is influenced by media visibility and geopolitical priorities rather than objective assessments of need [28].

Politicization and the Erosion of Neutrality

Humanitarian organizations increasingly struggle to maintain a perception of neutrality in highly politicized conflict environments, which can restrict access to affected populations and undermine operational safety. While humanitarian principles emphasize impartiality and independence, aid efforts often operate in tension with state sovereignty and political agendas. Some humanitarian organizations adopt broader approaches that extend beyond emergency relief to include post-crisis governance, democracy promotion, or peacebuilding. In contrast, others argue that such “transformational humanitarianism” undermines credibility and weakens perceived neutrality [28, p. 25].

Maintaining neutrality becomes even more difficult in contexts where non-state actors play prominent roles in governance, complicating negotiations and access to populations in need [29]. Armed conflict also frequently destroys critical infrastructure, hindering the movement of humanitarian personnel and supplies and further constraining access [30]. In some cases, parties to a conflict deliberately block or delay humanitarian assistance as a tactic to exert political leverage, forcing humanitarian organizations to negotiate for access under dangerous conditions.

For example, in April 2024, OCHA reported that Israeli authorities were blocking approximately 10 percent of planned aid missions into Gaza [31].

Perceptions of political bias are often intensified by the funding structures of humanitarian organizations, which frequently rely on government or politically affiliated donors. In highly politicized conflicts, this perceived alignment can place humanitarian workers at heightened risk. In 2024, the UN reported that 383 aid workers were killed worldwide, nearly half of whom were operating in Gaza and other occupied Palestinian territories [32]. This trend has continued, with provisional data indicating that 265 aid workers had been killed as of August 2025, according to the Aid Worker Security Database [33].

Beyond access restrictions and security risks, politicization can also produce unintended consequences that undermine humanitarian objectives. In some cases, aid has inadvertently strengthened harmful actors or prolonged conflict. During the 1990s famine in North Korea, food aid distributed through government channels was diverted to loyal military and party elites, allowing the regime to endure periods when collapse might otherwise have been likely. Similarly, following the 1994 genocide in Rwanda, refugee camps established by humanitarian organizations were exploited by militias as bases to regroup, rearm, and launch cross-border attacks, contributing to continued regional instability [28]. While humanitarian assistance has overwhelmingly saved lives, cases such as these have complicated public and political perceptions of aid and reinforced skepticism regarding neutrality, further constraining humanitarian operations in highly politicized environments.

Challenges in Food Assistance Delivery

Food assistance is particularly vulnerable to obstruction, manipulation, and logistical constraints in humanitarian crises. In contemporary conflicts, famine is frequently used as a tool by armed actors to control populations or punish opposition movements. While historically famines were often caused by natural disasters or economic collapse, in the twentieth and twenty-first centuries, they have increasingly resulted from deliberate political decisions by governments or armed groups. Governments may also manipulate or restrict access to data on food security, making it difficult for humanitarian organizations to formally declare famine conditions using standardized assessment frameworks such as the Integrated Food Security Phase Classification (IPC), which evaluates mortality rates, acute malnutrition in children, and household food consumption gaps [34]. During the 2021 crisis in Ethiopia's Tigray region, for example, telecommunications shutdowns severely limited aid organizations' ability to collect the data required for famine classification.

Food aid is also frequently subject to theft, diversion, and corruption. A United Nations report from Somalia estimated that up to 50 percent of food aid, valued at approximately $485 million in 2009, was diverted through corrupt networks involving armed groups and aid contractors [43]. These losses significantly reduce the effectiveness of humanitarian response and undermine public trust in aid operations.

Even when access is granted, distributing food safely and efficiently presents significant logistical challenges. Humanitarian organizations primarily deliver food through food baskets or prepared meals, both of which require secure access and functional infrastructure [35,36]. In conflict settings such as Gaza, blockades and movement restrictions have limited ground-based distribution, shifting reliance toward air delivery of food supplies [37]. While aircraft distribution can reach otherwise inaccessible areas, it entails greater logistical risk, requires large, secure drop zones, generates additional waste from reinforced packaging, and is substantially more expensive. Estimates suggest that air delivery can cost up to seven times more than road-based distribution, significantly limiting scale [38].

Food procurement practices introduce additional constraints. Much of the food distributed by U.S.-based organizations is purchased domestically and transported internationally on U.S. vessels, a process that can take up to six months and does little to support local economies. Regulatory requirements and intermediary costs further reduce efficiency, with estimates indicating that in 2013, approximately 59 cents of every dollar spent on U.S. food aid went toward middlemen rather than food delivery itself [41].

Climate Change and Increasing Humanitarian Demand

Climate change is an increasingly significant challenge for global humanitarian aid efforts. Rising temperatures and shifting climate patterns are intensifying the frequency and severity of extreme weather events, increasing the baseline likelihood of humanitarian crises worldwide. In 2021, the World Bank estimated that up to 216 million people could be internally displaced by slow-onset climate impacts by 2050, including approximately 86 million people in Sub-Saharan Africa alone [40]. Beyond displacement, climate change disproportionately affects low-income countries by weakening domestic capacity to prevent, prepare for, and respond to disasters. These compounding vulnerabilities increase reliance on international humanitarian assistance while straining already limited resources. At the same time, uncertainty surrounding the pace and effectiveness of global climate mitigation efforts makes future humanitarian demand difficult to predict, complicating long-term planning and resource allocation for humanitarian organizations.

Coordination Across Aid Organizations

Coordinating humanitarian response across multiple organizations remains a persistent challenge, particularly in large-scale and complex emergencies. At the country level, OCHA leads international humanitarian coordination by conducting needs assessments, mobilizing resources, and managing the humanitarian cluster system. Through this framework, OCHA encourages collaboration among governments, NGOs, and UN agencies to reduce duplication of effort and improve the efficiency of relief operations [26].

Despite these coordination mechanisms, significant obstacles remain. In conflict settings such as Gaza, damaged telecommunications infrastructure has severely limited effective communication among humanitarian actors. OCHA has reported that communication blackouts, jammed satellite phones, and outdated very high frequency (VHF) equipment have delayed operational planning and increased security risks for aid workers [39]. Beyond technical barriers, coordination is further complicated by the diversity of institutional mandates, funding streams, and operational models across humanitarian organizations, leading to misaligned priorities and fragmented responses. As a result, aligning multiple independent actors under a unified strategy remains resource-intensive and complex, even within established coordination frameworks.

Differences in organizational scale and specialization also present coordination challenges. Some humanitarian organizations operate across dozens of countries and sectors, allowing them to mobilize rapidly in response to large crises but increasing complexity in managing geographically dispersed operations. Others concentrate resources in fewer locations to provide deeper, specialized care, which can be particularly important in complex health or protection contexts. Balancing these approaches requires careful coordination to ensure that the breadth of coverage does not come at the expense of effectiveness or coherence. While there is broad agreement across the humanitarian sector on the importance of linking emergency relief with recovery efforts, differences in organizational ethos and capacity continue to complicate coordinated action.

Lack of Emergency Infrastructure and Institutional Preparedness

Another significant challenge to humanitarian aid is the absence of robust emergency systems in peacetime that can be rapidly mobilized during crises. Disaster sociologist Dr. Kathleen Tierney has argued that disasters are primarily social phenomena shaped by vulnerability, inequality, governance, and institutional trust. According to this perspective, hazards such as earthquakes, tsunamis, and hurricanes only become disasters when they intersect with fragile social systems. Addressing these weaknesses requires correcting structural inequities and strengthening public trust in institutions responsible for emergency response.

Dr. Tierney highlights Hurricane Katrina as an example, noting that many deaths resulted not from the storm itself but from the lack of evacuation plans for residents without access to private transportation. Media narratives and local government responses that framed affected communities as “looters” rather than victims further delayed the delivery of effective aid. Dr. Tierney also points to the Fukushima disaster, in which the Japanese government's reluctance to share accurate risk information with the public slowed emergency response and humanitarian assistance [42]. In both cases, failures in governance, communication, and social equity significantly shaped disaster outcomes.

Humanitarian organizations must operate within these preexisting structural conditions, which can limit the effectiveness of even well-coordinated response efforts. Common challenges across humanitarian operations include organizational capacity constraints, staff safety risks, funding volatility, and accountability concerns. Aid workers frequently face security threats in conflict settings, while funding levels fluctuate in response to donor priorities and public attention. Instances of misconduct within humanitarian organizations have also prompted calls for stronger accountability mechanisms [45]. Additionally, large international non-governmental organizations face increasing pressure to partner meaningfully with local actors, balancing global operational capacity with community-level empowerment and sustainability.

IV. Overview of Harvard Humanitarian Initiative (HHI)

The Harvard Humanitarian Initiative (HHI) was formally established in 2005 as a university-wide academic and research center dedicated to advancing humanitarian studies and practice. Its roots trace back to 1999, when Harvard created a program on humanitarian crises and human rights at the François-Xavier Bagnoud Center for Health and Human Rights. As global demand for technical expertise and training grew beyond the capacity of this program, Dr. Michael VanRooyen and Dr. Jennifer Leaning founded HHI in 2005 as co-directors [25]. Since 2010, Dr. VanRooyen has served as director, and in 2011, HHI launched the Humanitarian Academy at Harvard, the first educational institution of its kind. Today, HHI operates as an interfaculty initiative under the Office of the Provost, in close affiliation with the Harvard T.H. Chan School of Public Health and Brigham and Women's Hospital.

HHI's mission is clear: "to create new knowledge and advance evidence-based leadership in disasters and humanitarian crises" [25]. As with other humanitarian aid organizations, HHI focuses on immediate response rather than long-term development, emphasizing scientific evidence and principled action while protecting the rights, dignity, and autonomy of people affected by crises. Over nearly two decades, HHI has worked in more than 100 countries, trained over 200,000 humanitarians, and partnered with leading organizations such as the United Nations and the World Health Organization [25]. This mission continues to drive HHI's research, policy engagement, and education of the next generation of humanitarian leaders.

HHI is built upon a strong foundation of values that emphasize innovation, people-centered approaches, and measurable impact [25]. Its guiding principles include protecting human dignity, promoting equity and inclusion, and fostering partnerships at both global and local levels. These values inform how the Initiative designs its programs and engages with communities worldwide. In 2023 alone, HHI ran 18 programs, reached over 56,000 individuals across 190 countries, and collaborated with more than 150 partners [25]. As global humanitarian needs continue to grow, including widespread displacement, climate-related disasters, and conflict, HHI's integration within Harvard University and its partnerships with international NGOs, governments, and UN agencies position it as a hub for advancing humanitarian research, leadership development, and innovation.

HHI Programs and Organizational Structure

Organizationally, HHI is a collection of interconnected yet independently operating programs, each with a distinct focus, leadership structure, and set of external partners. While certain thematic areas overlap across programs, each operates with a high degree of autonomy within either the educational or research branches of the Initiative. Although HHI is unified by a broader mission and shared values, understanding its individual programs is essential to appreciating how the Initiative translates research, training, and policy engagement into practice. A summary of HHI's major programs, leadership, and focus areas is provided in Appendix C.

The Emergency Health Systems (EHS) team, managed by Dr. Catalina Gonzalez Marquez, focuses on strengthening emergency care systems during peacetime to improve humanitarian response capacity during crises. Since the start of the Russia–Ukraine war in 2022, the EHS team has supported emergency and trauma care training for frontline workers in Ukraine. A central component of this effort has been training local teachers to lead EHS instruction, a model that has enabled the program to become fully Ukraine-led. By embedding training capacity within local institutions, this approach promotes sustainability and resilience even as external funding fluctuates. The team is also exploring the use of virtual reality as a remote training tool, recognizing the need for continued skills development among deployed humanitarian workers [24].

The Noncommunicable Diseases and Conflict (NDC) team, led by Dr. Sylvia Kehlenbrink, addresses the management of chronic conditions such as diabetes in conflict and humanitarian settings. A major focus of the team's work involves ensuring access to insulin for populations affected by crises, a task complicated by insulin's strict temperature requirements. Maintaining cold-chain transport presents significant logistical and technological challenges, as highlighted during the COVID-19 pandemic. By prioritizing noncommunicable diseases, the NDC team seeks to address a gap in humanitarian response that has historically emphasized infectious diseases while overlooking chronic health needs.

The Children in Crisis team, led in part by Carolyn Baer, works to improve pediatric health and well-being in humanitarian contexts, with a particular emphasis on mental health. The team operates primarily through school-based programs in regions such as Ukraine, Jordan, and Ethiopia's Tigray region. A key focus of this work is connectivity, including access to reliable electricity, which is critical for delivering mental health education and support services in crisis-affected communities.

The Resilient Communities team, headed by Dr. Enzo Bollettino, partners with local organizations to strengthen disaster preparedness and response to natural hazards. The team emphasizes developing and maintaining early warning systems, particularly in disaster-prone countries such as the Philippines. Through nationwide surveys and community engagement, the program collects data on crisis experiences to inform locally grounded strategies for resilience and risk reduction.

The Humanitarian Geoanalytics team, managed by Dr. Erica Nelson, applies spatial analytics to study patterns of migration, conflict, public health, and land use. By analyzing geospatial data, the team aims to identify the root causes of humanitarian crises and anticipate future areas of risk. Ethical considerations are central to this work, particularly as mobile technologies and digital data sources become increasingly important tools in humanitarian response. The team places strong emphasis on responsible data collection, privacy, and informed use of spatial information.

The Humanitarian Academy at Harvard focuses on training professionals and public health students for work in the humanitarian sector. The Academy offers three core courses: the Humanitarian Response Intensive Course (HRIC), the Monitoring and Evaluation for Humanitarian Programs Workshop, and the Urban Humanitarian Emergencies Course. HRIC is the Academy’s most immersive offering, providing a three-day field simulation that replicates the pressures and complexities of real-world humanitarian crises.

Finally, the Transitional Justice team, co-led by Dr. Phuong Pham and Dr. Patrick Vinck, examines the intersection between humanitarian response and longer-term development and recovery. This team played a central role in piloting KoboToolbox, a free and widely used platform for humanitarian data collection and management. KoboToolbox enables offline data collection, real-time analysis, and secure data storage, making it especially well-suited for low-resource and crisis settings. The program continues to explore high-throughput data collection methods to support program evaluation and evidence-based decision-making.

V. Investigation & Synthesis

Client Engagement and Initial Observations

As part of the investigation phase of this project, our team conducted a client site visit and attended information sessions to better understand HHI's activities, organizational structure, and priorities. During our visit to HHI's offices at the Harvard T. H. Chan School of Public Health, we heard directly from multiple program leads about their ongoing work and the challenges they face in the humanitarian field. These discussions provided crucial context for understanding how HHI operates across research, education, and applied humanitarian engagement. A key observation from these conversations was that HHI's research and education efforts are closely intertwined. Many program leads are involved not only in conducting research but also in designing and delivering training programs for humanitarian practitioners. One example highlighted during the visit was the Humanitarian Response Intensive Course (HRIC), a simulation-based training intended to build both technical skills and emotional resilience in crisis settings. While HRIC was consistently described as a highly impactful learning experience, HHI staff noted that its reach is limited by its annual offering and relatively small cohort size.

Alongside education, data collection and analysis emerged as a recurring theme across conversations with HHI staff. Program leads emphasized that humanitarian data are often fragmented, collected using inconsistent standards, and difficult to share across teams and partner organizations. Privacy and ethical considerations further complicate data collection, particularly in conflict-affected and resource-limited settings. Although these challenges manifest differently across programs, they were consistently raised as barriers to improving humanitarian decision-making and impact. These initial observations informed the direction of our subsequent synthesis efforts and motivated a deeper examination of how data are collected, analyzed, and used across HHI.

Organizational Structure and Constraints

Building on our initial observations, subsequent conversations with HHI program directors and researchers provided more profound insight into how the organization is structured and how this structure shapes both collaboration and constraints. While HHI operates under a unified mission and executive leadership, its individual programs function with a high degree of autonomy. Each program independently develops relationships with external partners, including governmental agencies, non-governmental organizations, and academic collaborators. Although multiple programs may engage with the same partner organizations, these relationships are typically tailored to each team's specific research or humanitarian focus. This decentralized partnership model enables programs to remain flexible and responsive to specialized humanitarian needs, while also aligning expertise closely with project goals. At the same time, HHI staff noted that collaboration across teams is common when initiatives have overlapping objectives or can benefit multiple programs. Despite this internal collaboration, the individualized nature of partnerships remains a defining feature of HHI's organizational structure.

Funding dynamics further shape HHI's operations. Much of the financial support for projects is secured at the program level through external grants and partnerships, while all funding ultimately flows through HHI's central executive structure. This model provides oversight and consistency but also introduces constraints, as program priorities are often influenced by donor interests and funding availability. HHI leadership emphasized that, given the volatility of the humanitarian funding landscape, the Initiative must remain opportunistic in both project selection and partnership development. This necessity strongly influences HHI's team-centered structure and affects how projects are initiated, sustained, or scaled.

Together, these organizational characteristics help explain why challenges such as coordination and data integration persist across HHI. While decentralized teams and tailored partnerships support innovation and specialization, they can also lead to fragmentation in practices, including how data is collected, managed, and shared across programs. Understanding these structural constraints was essential for contextualizing the cross-cutting data challenges that emerged during our synthesis process.

Visual Mapping Synthesis

To synthesize the information gathered from client visits, presentations, and background research, our team employed a series of visual mapping techniques during the investigation phase. These exercises helped consolidate qualitative observations, reveal gaps in our understanding, and clarify HHI's priorities and values. Visual mapping also allowed us to move beyond individual program descriptions and examine patterns across the organization.

One key mapping exercise shown in Figure 1 involved plotting HHI's programs along two axes: the degree of emphasis placed on each program by HHI and the program's proximity to application, defined as deployment in active humanitarian crises. This exercise revealed that a significant portion of HHI's work is concentrated on data processing, analysis, and education, rather than on technologies or interventions physically deployed in crisis settings. Programs such as geospatial analysis, research-driven evaluation, and training initiatives were positioned closer to upstream stages of humanitarian response, emphasizing HHI's role as an educator and research institution rather than a direct aid implementer.

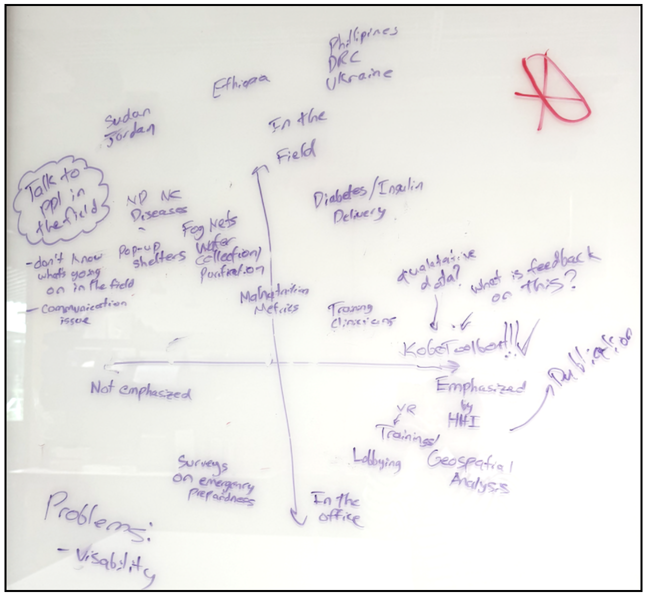


Figure 1. Mapping of HHI programs by degree of organizational emphasis and proximity to application in active humanitarian crises. Programs cluster toward upstream research, data analysis, and educational activities, highlighting HHI's primary role as an evidence-generating and capacity-building organization rather than a direct aid implementer.

This realization was critical in shaping our understanding of HHI's organizational identity. HHI consistently framed its mission around generating evidence, training humanitarian leaders, and informing future interventions rather than directly delivering aid. Data, in particular, emerged as a common thread across programs, serving as both an input to research and a foundation for educational content. Understanding how HHI views its own role helped ensure that our problem framing aligned with the organization's priorities rather than imposing external assumptions about what humanitarian impact should look like.

Additional synthesis exercises involved categorizing the challenges shared by HHI staff across programs, with supporting visual mappings provided in Appendix D. While both education- and data-related challenges surfaced repeatedly, data collection, analysis, and standardization issues appeared across a broader range of teams. These challenges included difficulties analyzing qualitative data, inconsistencies in metrics and definitions, and obstacles to integrating data from multiple sources. In contrast, education-related challenges were more program-specific, often centered on limitations in scalability and realism within training environments. This distinction reinforced the decision to focus the problem statement on data as a cross-cutting issue affecting HHI at multiple organizational levels.

Data Centered Challenges

Across nearly all HHI programs, challenges related to data collection, management, and analysis emerged as persistent and cross-cutting constraints. Unlike education-related limitations, which tended to be specific to particular programs or training formats, data challenges affected research, field engagement, and decision-making at multiple organizational levels. These issues not only slowed project execution but also limited HHI's ability to synthesize insights across programs and translate research into actionable guidance for humanitarian practitioners.

A major obstacle identified was the lack of standardization in data collection methods and metrics. HHI teams rely on a wide range of data sources, including household surveys, geospatial datasets, clinical records, and qualitative interviews. However, differences in definitions, measurement tools, and reporting formats often lead to duplicative efforts and inconsistencies that complicate comparison and aggregation across projects. For example, variations in how indicators such as malnutrition, displacement, or health system capacity are defined and measured make it challenging to align findings across regions or over time.

Field data collection presents additional challenges. While digital tools such as tablets and smartphones can improve efficiency, they are often costly, vulnerable to theft, and challenging to maintain in insecure or resource-limited settings. Limited internet connectivity further restricts real-time data upload and sharing, delaying analysis and response. Security and privacy concerns also play a significant role, as sensitive information collected in humanitarian contexts must be protected against misuse, interception, or breaches that could place communities at risk.

Analyzing qualitative data was another recurring concern. Many HHI projects depend on interviews, open-ended surveys, and narrative reports to capture community perspectives and contextual nuance. However, these data are time-intensive to process and difficult to standardize, particularly across languages and cultural contexts. While artificial intelligence and automated analysis tools offer potential efficiencies, HHI teams noted that many existing tools are not well adapted to local languages or context-specific humanitarian data, limiting their reliability and usefulness.

Taken together, these challenges underscore the need for more cohesive, secure, and adaptable data practices across HHI. Improving how data are collected, standardized, and analyzed would not only enhance individual program effectiveness but also strengthen HHI's ability to generate evidence that informs education, policy, and humanitarian practice more broadly. These findings directly informed the framing of our problem statement, which centers on data as a foundational constraint shaping HHI's capacity to scale impact across its diverse initiatives.

Education as a Supporting Challenge

Education and training are central components of HHI's mission, serving as key mechanisms for translating research into practice and preparing humanitarian professionals for work in crisis settings. Throughout the investigation phase, HHI staff emphasized the importance of experiential learning, particularly through in-person simulations such as the Humanitarian Response Intensive Course (HRIC). These programs are widely regarded as effective for building both technical skills and emotional resilience among participants.

However, the structure of these trainings presents inherent limitations. HRIC and similar immersive simulations are resource-intensive, requiring substantial faculty time, facilities, and logistical coordination. As a result, they are offered infrequently and can only reach a relatively small number of participants each year. This creates a persistent tension between depth and scale: highly realistic simulations provide meaningful learning experiences but are difficult to expand. At the same time, lower-cost virtual or asynchronous training formats can reach broader audiences but lack the realism needed to fully capture the uncertainty and complexity of humanitarian crises.

HHI has explored emerging technologies, such as virtual reality (VR), as a potential way to bridge this gap by increasing immersion while improving scalability. While these tools show promise, they also introduce new challenges related to cost, accessibility, technical support, and user training. Moreover, discussions with HHI staff suggested that educational delivery challenges, while important, are often downstream of broader organizational constraints, particularly those related to data availability, standardization, and analysis. Research outputs and field data inform the content of educational programs, meaning that limitations in data practices can directly shape the quality and relevance of training materials.

As a result, education emerged during synthesis as a supporting challenge rather than the primary focus of the problem statement. While improving the scalability and realism of training remains an essential goal for HHI, addressing underlying data-related constraints offers a more foundational opportunity to strengthen both research outputs and educational programming across the organization.

Challenges in the Synthesis Process

The synthesis phase presented several challenges that required careful reflection and iteration. One of the most significant was resisting the tendency to move prematurely toward solution development. Early in the process, the team often gravitated toward specific technologies or tools and worked backward to define the problem they might address. This approach limited a full exploration of HHI's organizational needs and risked narrowing the problem space too early. Through iterative mapping exercises and structured reflection, the team was able to step back and engage more holistically with the information gathered during the investigation phase.

Another consideration during synthesis involved aligning the framing of HHI's challenges across a diverse set of observations and perspectives. Team members initially highlighted different dimensions of HHI's work, including education, training delivery, field engagement, and data management. These perspectives led to multiple iterations of the problem framing, each emphasizing a different aspect of HHI's operations. Refinement across these iterations clarified the need to balance specificity and breadth. It reinforced the importance of identifying a challenge that was relevant across programs rather than limited to a single initiative.

Feedback from course staff, particularly Professor Mooney, played an essential role in strengthening the synthesis process. Instructor guidance emphasized clearly separating problem identification from solution ideation and grounding the analysis in patterns emerging from the investigation rather than assumptions about technical feasibility. Incorporating this feedback strengthened our synthesis process and supported convergence toward a better defined, more defensible problem statement grounded in cross-cutting organizational challenges rather than isolated issues.

Macro-Level Patterns and Problem Framing

Examining HHI's work at a macro level revealed patterns that were not immediately apparent when considering individual programs in isolation. While HHI's initiatives span research, education, field engagement, and policy advising, common constraints emerged across these domains, particularly around how data are collected, shared, and translated into actionable insights. Taking this "zoomed-out" perspective allowed us to move beyond program-specific challenges and identify systemic limitations shaping HHI's capacity to scale impact across the humanitarian sector.

This synthesis also reinforced alignment with HHI's mission to advance evidence-based leadership in disasters and humanitarian crises. Across conversations with staff and program leads, data consistently appeared as the connective tissue linking research outputs, educational content, and field decision-making. Limitations in data accessibility, standardization, and real-time availability constrained not only research quality but also the effectiveness of training programs and the ability of humanitarian actors to respond dynamically in crisis settings. While education remains a core component of HHI's work, the synthesis process indicates that many educational challenges are downstream of broader data-related constraints.

Together, these observations informed the framing of our problem statement, which focuses on data as a foundational, cross-cutting challenge affecting multiple levels of HHI's operations and partnerships.

Problem Statement

At its core, the problem centers on the absence of reliable, accessible, and real-time field data that can flow seamlessly into HHI's research, training, and operational partnerships. **HHI requires improved tools for providing widely accessible real-time field data to humanitarian aid workers and organizations.** Without a consistent mechanism for capturing structured information in crisis environments, educational content becomes difficult to update, field simulations lose relevance, and humanitarian actors are left without timely insights to inform decisions. Addressing this gap requires solutions that strengthen data collection and integration at the source, ensuring that HHI's educational programming and the broader humanitarian community can draw on accurate and immediate field information.

VI. Ideation

At the conclusion of Phase 1, the investigation and synthesis process, our team developed a problem statement that guided the ideation phase of the project. Phase 2 focused on generating and evaluating potential solutions that could address the data-related challenges identified in collaboration with HHI. Given the breadth of the problem space, the ideation process was intentionally expansive and iterative, as shown in Figure 2, allowing the team to explore a wide range of project directions before narrowing the focus.

Initially, we considered a broad scope of approximately 10 to 15 potential project ideas. Through successive rounds of refinement and feedback from both the ES96 teaching staff and HHI, these ideas were evaluated and narrowed to a single solution for further development. This section describes the ideation process, the considerations that guided idea generation and evaluation, and the rationale for selecting the final concept.

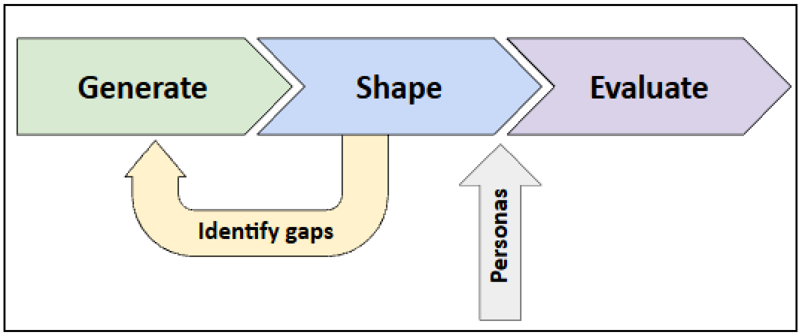


Figure 2. Iterative ideation and evaluation process used to generate, organize, and refine project concepts. Individual brainstorming, collective clustering, gap identification, persona consideration, and structured evaluation informed the selection of the final project direction.

Initial Project Brainstorming

Our brainstorming process was centered around the core of our problem statement. As a team, we recognized that the Harvard Humanitarian Initiative (HHI) requires improved tools to provide widely accessible, up–to–date, real-time field data to humanitarian aid workers and organizations. Simultaneously, we recognized that our problem statement identifies multiple gaps in HHI's infrastructure and current operations. Given this complexity, we were committed to pursuing a solution that not only addressed our client's needs but was also realistic within the scope and constraints of our project timeline and resources. To ensure a well-rounded, strategic approach, we adopted an iterative, collaborative brainstorming process, as shown in Figure 2. We began by generating ideas individually and then collaboratively mapping them into various groupings, allowing us to visualize patterns, themes, and possible areas of overlap amongst our ideas, shown in Figure 3. After forming our initial groupings, we analyzed them to explore alternative ways to categorize or frame our ideas. After refining our groupings, we stepped back to identify any gaps we had overlooked and generate additional ideas to address them. This reflective step was crucial in expanding our ideation process and ensuring comprehensive coverage of the problem we identified.

To further ground our ideas in real-world impact, we incorporated user-centered design principles by developing representative personas. For example, we created “Mark,” a frontline humanitarian aid worker responsible for tasks like counting children in a displaced population. These personas allowed us to critically assess the value, usability, and relevance of our proposed solutions from the perspective of different stakeholders. Finally, we evaluated each idea through various lenses, including feasibility, alignment with our problem statement, and potential to meet the evolving needs of HHI.

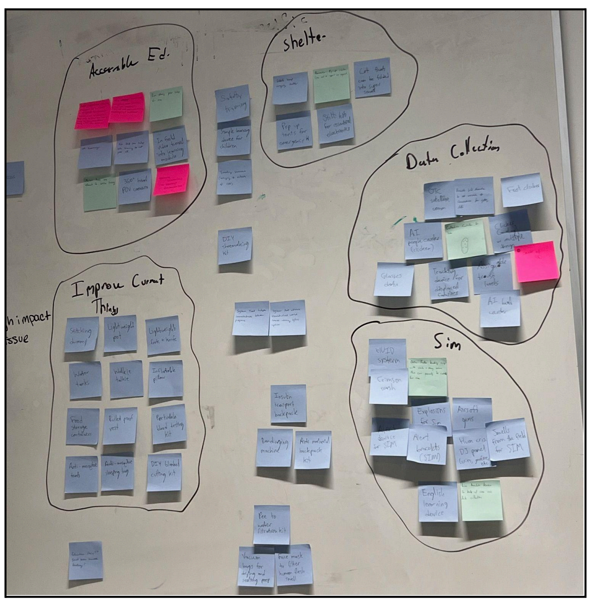


Figure 3. Collaborative clustering of individually generated project ideas into thematic categories to support pattern recognition, gap identification, and iterative refinement during the ideation process.

A particularly useful lens during the ideation process was considering the interests of HHI subgroups and partner organizations. In meetings with HHI staff, several recurring issues affecting broad segments of the organization were consistently emphasized. HHI also provided guidance based on their understanding of our team's resources and technical capabilities, which helped steer ideation toward specific programs and project types. Taking these factors into account, the team evaluated the strengths and limitations of the proposed concepts to ensure alignment with both HHI priorities and project feasibility. To capture a representative range of HHI's interests, ideas were organized into three categories: research data–oriented solutions, educational tools, and field services devices.

Initial Project Ideas

Research Data–Oriented Solutions

Within the category of research data–oriented solutions, the team developed three initial project concepts: GIS Integration Technology, a Coordination and Communication Platform, and KoboClicker.

The GIS Integration Technology concept explored the development of software tools to simplify access to, use of, and interpretation of satellite imagery in humanitarian contexts. The goal was to make geospatial information more accessible not only to researchers and humanitarian aid workers, but also to local communities affected by crises. Improved GIS tools could help standardize service localization, support navigation after displacement, and enable real-time decision-making in the field. As part of this concept, the team also considered complementary hardware solutions, such as portable high-capacity chargers or energy-generating devices, to support on-site use of GIS technologies in low-resource environments. This idea was ultimately refined into a proposal for a lightweight mobile application with offline imagery packs, allowing users to view, annotate, and interact with map data without continuous internet access.

A second concept was the Coordination and Communication Platform, envisioned as a shared digital space for faculty, researchers, and partner organizations to exchange data, research findings, and insights. This platform would also link research outputs to relevant educational programming, enabling stronger integration between data generation and training efforts. Such a tool could support internal coordination across HHI programs and facilitate collaboration with external partners operating in the field.

The third concept, KoboClicker, focused on improving quantitative data collection in humanitarian settings. This idea proposed a handheld device with programmable buttons mapped to customizable data collection parameters. Designed to interface directly with KoboToolbox, the clicker could be Bluetooth-enabled or wired, allowing field workers to record survey responses without relying on high-value devices such as smartphones or tablets. By reducing dependence on easily stolen or fragile equipment, this approach aimed to improve data security, usability, and reliability during field data collection.

Educational Tools

The team generated several concepts focused on enhancing educational tools, particularly those designed to support simulation-based learning. One set of ideas aimed to increase the realism and instructional value of the Humanitarian Response Intensive Course (HRIC) simulation. The immersive soundscape concept proposed integrating a sound-control system that replicates the auditory conditions of real humanitarian crises, such as crowd noise, emergency communications, and environmental stressors. This addition would enhance realism and support training in stress management and situational awareness.

Another simulation-focused concept involved creating a simulated financial system that mirrors real-world money transfer mechanisms and credit card transactions. This system would allow participants to practice making financial trade-offs under pressure, accounting for budget constraints, donor requirements, and evolving field conditions. Such an approach would help trainees better understand the operational and ethical complexities of humanitarian decision-making. A third simulation enhancement, simulation wearables, proposed low-technology wearable bands capable of changing color or vibrating to communicate discreet instructions or status updates to participants without disrupting the immersive environment. These wearables would allow facilitators to guide participants while maintaining continuity within the simulation experience.

These simulation-focused ideas were later consolidated into a single concept presented to HHI as Simulation Sensory and Coordination Controls, combining environmental realism with discreet participant communication.

In addition to simulation enhancements, the team explored tools to support broader educational programming. One concept involved improving learning kits, packaged similarly to first-aid kits, that could accompany virtual or in-person training sessions. These kits would include simplified, low-cost versions of physical tools commonly used in crisis settings, enabling hands-on learning outside of immersive simulations. This idea was expanded to include improved training mannequins designed to better represent diverse anatomies and physical conditions encountered in humanitarian contexts.

Finally, the team considered augmented and virtual reality (AR/VR) training tools to improve visual immersion and simulate a broader range of crisis scenarios, such as medical interventions or disaster response environments. A related idea explored developing a game-based simulation to expand access to virtual training and engage learners through interactive, scenario-driven experiences. These concepts emerged from discussions with HHI about the need to scale educational offerings while maintaining meaningful experiential learning.

Field Response Devices

The final category of initial project ideas focused on field response devices, emphasizing tools that could be deployed directly in humanitarian settings. Three concepts were explored within this category: Pop-Up Shelters, AI integration into KoboToolbox, and E-Cash Wearables.

The Pop-Up Shelter concept aimed to improve the speed and simplicity of emergency shelter deployment. Ideas included structures that could be assembled using origami-inspired folding mechanisms or rapidly deployed through inflation-based systems. These approaches were intended to reduce setup time, minimize the need for specialized tools or training, and improve adaptability in rapidly evolving crisis environments.

Another concept involved integrating artificial intelligence into KoboToolbox to support both survey design and data analysis. This integration could assist users by generating context-specific survey questions, automating portions of qualitative and quantitative analysis, and producing real-time summaries and insights from collected data. By reducing duplication of effort and accelerating the transition from data collection to actionable information, this approach could significantly shorten the data-to-action pipeline in humanitarian response.

The third concept, e-cash wearables, explored alternative methods for distributing and managing electronic cash assistance. This idea proposed a wearable device, such as a bracelet or belt-mounted tag, capable of handling secure e-cash transfers and payments. Compared to smartphones, the wearable would be less conspicuous, reducing theft risk, and more accessible in communities with limited mobile phone ownership or unreliable cellular connectivity, particularly in rural or low-resource settings.

Solution Selection Considerations

In total, the team generated 13 initial project ideas, grouped into three categories: research data–oriented projects, educational tools, and field response devices. These concepts were presented to the ES96 teaching staff on September 29 for feedback. Based on instructor input and further team reflection, we decided to narrow and, where appropriate, combine several ideas. To support this process, we developed a rating-based evaluation framework to systematically assess and prioritize the proposed concepts.

Solution Rating Metrics

To compare our potential project ideas quantitatively, we used four criteria. We chose the criteria based on their ability to highlight the essential requirements for both the class and our client: humanitarian aid impact, cost, development time, and the likelihood of HHI adopting the solution. We rated each project on a scale of 1 to 5 for each of the four criteria, where a score of 5 signifies a highly desirable outcome, and 1 represents a less desirable one.

These four criteria were selected because they balance both the engineering and humanitarian dimensions of the project. Humanitarian aid impact ensures that our design meaningfully advances HHI's mission by maximizing the number of people helped. Cost reflects the financial constraints that humanitarian organizations often face, encouraging scalable, sustainable solutions. Development time considers the technical feasibility and logistics of completing the project within the class timeframe while maintaining engineering quality. Finally, the likelihood that HHI adopts the solution ensures that our efforts lead to an implementable, real-world outcome: one that aligns with HHI's operational goals and has a genuine chance of deployment in the field.

Together, the criteria as detailed in Table 1 guided our team toward a project that is impactful, feasible, and aligned with both the humanitarian mission of HHI and the learning objectives of a student-led engineering course.

Table 1. Scoring criteria used to evaluate and compare proposed project concepts across humanitarian impact, cost, development time, and likelihood of adoption by HHI. Scores range from 1 (lowest) to 5 (highest) for each criterion.

Criteria	Score = 1	Score = 2	Score = 3	Score = 4	Score = 5
Humanitarian Aid Impact *(Based on OECD, CHC, and Sphere) [1-3]*	Little to no measurable improvement in well-being, safety, or dignity.	Limited indirect benefits, such as modest improvements in training, coordination, or administrative efficiency.	Moderate, targeted improvements in well-being or resilience for specific users or contexts.	Significant improvements in access to essential services, safety, efficiency, or decision-making for humanitarian actors.	Direct & measurable reductions in mortality, morbidity, or protection risks, with substantial benefits for aid workers & affected populations.
Cost	\$50,000+	\$10,000 - \$50,000	\$5,000 - \$10,000	\$1,000 - \$5,000	< \$1,000
Development Time	24+ months	12 - 24 months	6 - 12 months	3 - 6 months	< 3 months
Likelihood of HHI Adopting	< 30%	30 - 50%	50 - 70%	70 - 90%	90%+

Project Idea Scores

We had thirteen original ideas, and we rated each idea in the four pre-determined criteria and came up with a total score for each as shown in Table 2. The next section explains our methodologies for calculating certain scores.

Table 2. The Pugh matrix captures the scores of each project idea, evaluated under the various categories detailed above.

Idea	Technical Feasibility	Human Impact	Cost	Develop Time	Adoption Probability	Total Score
GIS Integration Technology	2	3	2	3	2	**12**
KoboClicker	5	4	4	5	5	**23**

Coordination Platform	3	2	2	4	4	**15**
Immersive Soundscape	3	2	5	5	4	**19**
Simulated Financial System	2	2	2	3	2	**11**
Simulation Wearables	4	2	3	5	4	**19**
Learning Kits	5	2	4	4	3	**19**
Educational Dummies	5	2	2	3	3	**17**
AR/VR Trainings	1	2	2	2	5	**12**
Video Game Simulations	4	2	2	1	5	**14**
Pop-Up Shelters	3	3	3	1	3	**13**
AI KoboToolbox Integration	3	3	3	4	3	**16**
E-cash Wearable	3	3	2	2	4	**14**

Score Reasoning

In the subsection above, as mentioned each project proposed was quantitatively graded on the same scale from 1 to 5, on multiple categories that are relevant to the discussion of selecting the most pertinent project. On these grounds, insight into the grade associated with each project across the categories is provided in the Tables 3-5 below.

Table 3. Evaluating the level of improvement of each project in the respective field.

Project	Reasoning

GIS Integration Technology	Gives people a better sense of direction and information about the services available; however, it is only available to those with smartphones, 1 in 4.
KoboClicker	Helps collect population data more accurately and efficiently.
Coordination Platform	Software helps share data between researchers, but does not specifically have a large impact on the humanitarian field.
Immersive Soundscape	Makes crisis simulation more realistic but not a drastic improvement for the field.
Simulated Financial System	Test financial decision-making in crisis simulation, does not have many tangible effects in saving lives.
Simulation Wearables	Coordinate crisis simulation training, and slightly improve performance in simulation.
Learning Kits	Provide hands-on training to virtual trainees, moderate tangible improvements in the field
Educational Dummies	Mimic giving medical aid in crisis with precision data
AR/VR Trainings	Recreate humanitarian crises virtually, little field tangible effects on quality of life
Video Game Simulations	Practice decision making in dynamic crisis situations, moderate training improvement.
Pop-Up Shelters	Targeted improvement in temporary shelter to displaced people, moderately improves QoL
AI KoboToolbox Integration	Synthesize data into actionable learnings, moderate targeted action improvement.
E-cash Wearable	Securitizes distribution of E-cash, slightly improves the QoL.

Table 4. Evaluating the technical feasibility of each project.

Project	Reasoning
GIS Integration Technology	Requires access to large datasets and mapping APIs, which demand advanced GIS knowledge and computational resources.

	Feasible only with specialized expertise and additional infrastructure.
KoboClicker	Highly feasible using simple microcontrollers (e.g., ESP32 or Arduino) connected to KoboToolbox via its API. The hardware, software, and integration are all achievable within available SEAS lab resources.
Coordination Platform	Technically feasible but software-heavy. Would require significant time for backend development, UI design, and secure database hosting, making it less suitable within the project's timeframe.
Immersive Soundscape	Feasible through the use of embedded components like small speakers, microcontrollers, and sound libraries. Complexity lies in coordinating multiple sound outputs and synchronizing them with simulation events.
Simulated Financial System	Low feasibility due to the need for secure user management, financial simulation algorithms, and backend database systems. Requires more complex software and cybersecurity considerations.
Simulation Wearables	Moderately feasible; uses existing sensor and actuator technologies. Challenges include compact hardware design, signal synchronization, and ensuring consistent wireless communication.
Learning Kits	Integration of low-cost sensors and simple circuits to provide real-time feedback; highly feasible within current lab capabilities but requires precision calibration, durability testing, and refinement for consistent performance.
Educational Dummies	Feasible in concept but requires advanced fabrication, pressure sensors, and mechanical systems for realistic physiological feedback, making prototyping more complex and time-consuming.
AR/VR Trainings	Low feasibility; requires advanced hardware, 3D modeling, and VR environments not accessible to the team. Development would be resource- and time-intensive.
Video Game Simulations	Requires specialized programming skills and game engine knowledge (e.g., Unity or Unreal). Technically demanding and difficult to complete within the semester.

Pop-Up Shelters	Requires structural testing and reliable materials; pneumatic or foldable mechanisms add complexity and may exceed available fabrication tools. Weight, insulation, and climate adaptability are additional engineering challenges.
AI KoboToolbox Integration	Would involve machine learning models and backend integration for data analysis. High technical complexity and data requirements make it infeasible within available resources.
E-Cash Wearable	Hardware components (NFC, microcontroller) are manageable, but the main challenge is implementing secure encryption and reliable data transfer, which requires advanced cybersecurity and backend knowledge.

Table 5. Evaluating the cost of each project.

Project	Reasoning
GIS Integration Technology	Minimum Viable Product would still need paid data access, software licensing, or cloud credits, likely above $10K.
KoboClicker	MVP with a few microcontroller-based devices and open-source tools could be built under $5K.
Coordination Platform	Even with open-source components, integration and secure hosting would exceed $10K.
Immersive Soundscape	MVP using low-cost speakers and free sound libraries could stay under $1K.
Simulated Financial System	Interactive dashboards and secure user handling push MVP above $10K.
Simulation Wearables	A small batch of sensor prototypes with basic enclosures likely costs $5-10K.
Learning Kits	Limited MVP with Arduino sensors and low-cost fabrication fits within $1-5K.
Educational Dummies	Custom molding and embedded electronics keep MVP above $10K.
AR/VR Trainings	Developing immersive scenes and hardware setup for MVP exceeds $10K.
Video Game Simulations	Creating 3D assets and functional gameplay would put MVP above $10K.

Pop-Up Shelters	Basic prototypes using low-cost materials could stay near $5-10K.
AI KoboToolbox Integration	Simple automation or data-cleaning MVP with open-source AI fits in $5-10K.
E-Cash Wearable	Secure NFC chips and backend setup make MVP cost exceed $10K.

Solution Narrowing Based on Client Feedback

Using the evaluation metric system described above, the team narrowed the initial set of project ideas. Within the research data–oriented projects category, GIS Integration Technology and the Coordination and Communication Platform were eliminated due to lower evaluation scores. In the educational tools category, simulated financial systems, AR/VR training tools, and video game–based simulations were removed. The immersive soundscape and simulation signaling wearable concepts were merged into a single proposal, Simulation Sensory and Coordination Controls, and educational modeling dummies were incorporated into the Learning Kits concept. Finally, within the field response devices category, the AI integration into KoboToolbox concept was removed.

Following this refinement, five concepts were presented to the client on October 1: KoboClicker, Simulation Sensory and Coordination Controls, Learning Kits, Pop-Up Shelters, and E-Cash Wearables, and we received valuable feedback. Dr. Pham expressed strong support for KoboClicker, noting its practical applicability in field settings. She also suggested that integration with location data could increase its utility, while cautioning that GPS functionality may introduce security risks. KoboToolbox's existing use of location stamps was noted as a potential reference point for balancing functionality and safety.

Both Simulation Sensory and Coordination Controls and Learning Kits were well received as feasible and potentially impactful educational tools. While limited concerns were raised, it was noted that versions of learning kits may already be in use at Brigham and Women's Hospital, suggesting the importance of differentiation or adaptation.

The Pop-Up Shelter concept prompted extensive discussion regarding field implementation. The client emphasized the need to respect the dignity of displaced families, ensure cultural adaptability, and carefully consider material weight and deployment logistics. It was noted that shelter needs vary significantly by context and conflict, and that a one-size-fits-all shelter solution would be unlikely to achieve the intended impact.

Finally, E-Cash Wearables received a positive response, particularly for their potential to address challenges related to phone dependency and verification. However, the client highlighted the importance of developing a robust and secure verification mechanism. It was suggested that this concept might be best pursued in partnership with organizations already implementing e-cash systems, such as the UNHCR or the World Food Programme (WFP). While the idea aligned well with humanitarian needs, it was noted that HHI itself would likely not be the product's primary end user.

Further Solution Narrowing

After narrowing down the solutions, the five remaining were KoboClicker, Pop-Up Shelters, Learning kits, an E-cash wearable, and simulation sensory and coordination controls. Each of these solutions was then considered in further detail and described below.

KoboClicker

The first solution we explored in further detail is KoboClicker. KoboToolbox, developed by HHI, is an open-source data collection, management, and visualization platform widely used in humanitarian crises. However, KoboToolbox typically requires the use of expensive and high-value devices, most often tablets, for data collection. This dependence increases the risk of theft and can limit data collection capabilities due to short battery life. It may also introduce data inaccuracies, as collectors must repeatedly shift their attention between the device and their surroundings.

To address these challenges, we proposed KoboClicker, a device designed to integrate directly with KoboToolbox and simplify on-the-ground data collection. KoboClicker features an ergonomic form factor with easily programmable buttons that users can map to specific data parameters of interest. In the field, users can quickly record observations, such as age or gender, by pressing the corresponding buttons without looking down at a screen. Data collected through the clicker would sync directly with KoboToolbox. Additionally, KoboClicker may improve data security, as the device itself would not store sensitive information but instead only record button inputs for transmission to the platform.

Pop-Up Shelters

Temporary shelters remain a significant challenge in humanitarian crisis response. Existing modular shelter systems are often slow to deploy, poorly adapted to extreme weather conditions, and costly to produce and maintain. To address these limitations, we proposed an improved pop-up shelter design that draws from existing origami-inspired and inflatable shelter technologies.

The proposed shelter incorporates integrated sensors to support climate monitoring and control, including measurements of humidity, carbon dioxide (CO_2) levels, and other environmental variables that may affect occupant health. Improved insulation and optional air-conditioning components would enhance temperature regulation, helping maintain comfortable living conditions across diverse climates. The shelter's interior layout would be reconfigurable, with movable internal walls that allow for increased privacy and customization based on user needs. In addition, the shelter roof could incorporate solar panels to enable on-site energy generation, helping ensure reliable electricity access in low-resource settings. Further modular features or add-on components could be included to allow the shelter to be adapted to varying humanitarian contexts and operational requirements.

Learning Kits

In-person simulations provide extremely valuable hands-on educational experiences to humanitarian aid workers. However, their reach is limited as the simulations are location-specific, only happen once a year, and can only accommodate a small number of participants. Conversely, online courses are easily accessible but are limited in the effectiveness and realism of their content. To bridge this gap, we proposed hands-on learning kits designed to accompany virtual (or in-person) training. These kits would accompany specific training areas, including medical practices, hygiene, and water filtration. They would be packaged similarly to existing first aid kits used in crisis zones; they would be compact and include simplified, cheaper versions of commonly used physical tools. Some specific examples of tools that may be developed for the learning kits are improved training dummies and inflatable training arms. Current dummies are unrealistic and typically only resemble adult males. Our improved version would be adjustable, allowing users to experience treating various body types, and more realistic by simulating physiological systems such as airways, sound, smell, and pulse. They might also include force sensors to inform the user if they are using too much or too little force. The inflatable arm would be a helpful tool for aid workers to practice diagnosing medical conditions such as malnutrition. Pressure sensors in the arm would provide immediate feedback via an app, helping users better understand proper measurement techniques.

E-Cash Wearable

A prominent concept that emerged during the ideation process was the e-cash wearable. During our initial meeting with the Director of HHI, we noted a growing trend in humanitarian aid delivery: leveraging displaced individuals' personal smartphones to distribute assistance in the form of electronic cash (e-cash), using the devices for identification and verification. Compared to the physical transport and distribution of material aid, e-cash transfers are often more cost-effective and flexible within complex humanitarian environments.

However, further research revealed significant limitations to smartphone-dependent aid delivery. A 2016 report from the UNHCR indicated that refugees are 50% less likely than the general population to own an internet-enabled phone, that 29% of refugee households have no phone at all, and that approximately 20% of refugees living in rural areas lack network coverage [61]. Although access to mobile technology has improved since then, persistent barriers remain, including affordability, digital literacy, charging access, and limited infrastructure.

To address these constraints, we proposed leveraging existing cash transfer stations while replacing smartphone-based identification with a low-cost wearable device equipped with a secure near-field communication (NFC) chip. This wearable would allow recipients to authenticate their identity and transfer e-cash to physical currency without relying on a personal smartphone. To reduce visibility and theft risk, the wearable could be designed in a discreet form factor, such as a belt-mounted tag or a necklace. Additional security measures were also considered, including biometric verification through a fingerprint reader or a mechanical locking mechanism that disables NFC functionality until authentication is complete. Together, these features would reduce dependence on mobile phones and network connectivity, streamline identity verification, and mitigate risks of device theft and related violence.

Simulation Sensory and Coordination Controls

One of the projects we presented to our client focused on enhancing the Humanitarian Response Intensive Course (HRIC), run by the Harvard Humanitarian Initiative (HHI), by developing a sensory and coordination tool. The HRIC is a highly immersive, large-scale field simulation that allows students to apply classroom knowledge in a realistic, high-pressure environment of a humanitarian crisis zone. Designed and run by faculty, staff, and expert practitioners, the simulation requires extensive coordination across a wide array of instructors and volunteer actors who role-play in complex, evolving scenarios. Recognizing the scale, complexity, and immersive nature of this simulation, our team proposed an innovative wearable wristband for actors, designed to address two critical aspects of creating the simulation experience: real-time coordination and auditory immersion.

The wristband would serve as a discreet yet powerful tool, streamlining communication and enabling dynamic, responsive adjustments during the simulation. Using a centralized control system, such as a dedicated app or remote control panel connected to actors' wristbands, simulation coordinators could send real-time or pre-programmed instructions to actors. The instructions could be delivered via color-coded visual alerts displayed on the band, signaling different types of events or urgency levels, or simple messages/cues that guide actor responses or transitions. This system enables reactive, scenario-based storytelling, where actor movements and interactions dynamically evolve in response to the decisions and actions of student participants, creating a more adaptive and engaging learning environment.

To deepen the realism of the simulation, the wristbands would also include built-in speakers capable of playing localized sound effects that simulate crisis-zone environments, such as distant gunfire, crowd noise, sirens, etc. The sounds could also be triggered remotely via the centralized control system and synchronized across multiple wristbands to create an immersive audio environment without the need for an integrated in-setting sound system. The sounds could also be scaled in intensity to be sensitive to participants' comfort levels and to avoid overstimulation or trauma triggers. By decentralizing sound output and enabling mobile, actor-specific audio cues, this feature amplifies the simulation's immersive quality while offering greater flexibility for scene transitions and environmental changes.

Further Scoring

We scored the five remaining project ideas using the Pugh matrix in Table 6, ranked from 1 to 5, where 1 indicates the least alignment with the criteria, and 5 indicates the most alignment with the criteria. Given client feedback, our criteria were mission impact, user value in the field, technical feasibility, time to MVP/Schedule risk, in budget, ethical privacy & security, scalability, maintainability, and client feedback & adoption likelihood.

	Simulation Wristand	KoboClicker	Learning Kits	Pop-Up Shelters	E-Cash
Mission impact	4	5	5	3	4
User value in the field	3	5	3	4	4
Technical feasibility	4	5	5	3	3
Time to MVP / schedule risk	4	4	4	1	3
In Budget	3	3	4	2	3
Privacy & security risk	4	3	5	5	3
Scalability & maintainability	5	5	4	4	4
Client adoption likelihood	2	5	2	3	4
Total	29	35	32	25	28

Table 6. The Pugh matrix captures the scores of the final five project ideas, evaluated under the various categories named above.

Taken together, the scoring results made the KoboClicker the clear choice for further development. It consistently ranked highest across feasibility, cost, development time, and likelihood of adoption, while also demonstrating strong potential for humanitarian impact relative to the other concepts, and got the highest score in client feedback & adoption likelihood. Notably, the KoboClicker directly addresses a documented gap in field data collection, aligns with HHI's operational priorities, and is realistic to prototype within the scope of this course. For these reasons, our team selected the KoboClicker as the project to advance, and the next section provides an overview of its purpose and potential value in humanitarian settings.

VII. KoboClicker Overview

The KoboClicker is a rugged, multi-metric hardware device designed to bring real-time, high-integrity data collection into humanitarian field operations. Built for low-resource and high-pressure environments, it enables responders to capture multiple variables at once, quickly and discreetly, without relying on fragile or distracting mobile devices. By pairing seamlessly with KoboToolbox via low-energy Bluetooth, the KoboClicker enables instant, secure data transmission from the field to decision-makers, reducing errors, eliminating transcription delays, and expanding what can be collected during rapid assessments. As a new modality for frontline data capture, it strengthens the entire humanitarian workflow while improving the speed, accuracy, and usability of existing digital tools. This sets the stage for understanding how KoboToolbox operates today and why integrating a purpose-built device like the KoboClicker significantly enhances its impact.

KoboToolbox is an open-source digital platform designed to help humanitarian organizations collect and analyze data more rapidly and effectively. It enables users to build, deploy, and visualize surveys while streamlining data workflows and enhancing security. Its ability to operate fully offline and sync automatically when the internet becomes available makes it especially well-suited for crisis environments and low-resource settings. By replacing slow, paper-based methods and reducing reliance on expensive proprietary tools, KoboToolbox has significantly improved the speed, accessibility, and reliability of humanitarian data collection. Today, the platform supports more than 32,000 organizations, including UN agencies, NGOs, and local responders, across more than 200 countries, and facilitates more than 20 million survey submissions each month [62].

As KoboToolbox becomes the default data platform for humanitarian response, it is accelerating needs assessments, strengthening evidence-based decision-making, and improving coordination across agencies and countries. Its expanding user base has also enabled greater inclusion of local organizations in global data efforts, ensuring that frontline actors can contribute directly to shared information systems. This widespread adoption now presents a clear opportunity: enhancing the user experience and streamlining the transition of data from field collection to the platform will further increase the value and scalability of KoboToolbox.

KoboToolbox currently faces several limitations in real-world data collection. Field staff often rely on smartphones or tablets, which introduces operational challenges such as short battery life, vulnerability to damage, and distraction during live observations. Constantly looking down at a screen slows counting and reduces situational awareness. A dedicated clicker would address this by replacing screens with simple tactile buttons that allow rapid, unobtrusive data entry. In addition, KoboToolbox is built primarily for touchscreen or web-based inputs, offering limited support for external hardware. This restricts its usefulness in projects that require fast, real-time data capture. Integrating a clicker through custom middleware would translate button presses directly into structured data fields and create a path for future hardware integrations. Finally, KoboToolbox's survey-style forms allow entries to be altered or mistyped, compromising data integrity. A clicker system would generate time-stamped, immutable logs, reducing errors and strengthening the reliability of field data [63].

To define a specific market gap the KoboClicker fills, our team researched similar products currently used in humanitarian work. The standard tally counters used in the field are simple analog devices with a count button and a reset button. The UNHCR, one of the most prominent users of these counters, uses robust metal devices with a unit cost of $4 USD and weighing 2.5 to 3.5 ounces [61]. Counters are most often used as part of rapid needs assessments, broad evaluations conducted by humanitarian aid groups within the first 48 to 72 hours after deployment, to gauge the severity of a crisis and the immediate needs of the population [64]. The counters have several use cases: 1) population estimation in camps and at borders; 2) density sampling, especially in refugee camps; 3) camp perimeter and area measurements; 4) map generation in remote areas.[1] This data is combined into an initial report, which is often shared among different aid organizations and is used to guide strategy development. Tally counters are versatile and fast data-collection tools, making them well-suited to the acute time pressure and broad scope of a rapid needs assessment. Other notable strengths include cost-effectiveness, a small footprint, and robustness, enabling use across a wide range of climates with minimal maintenance. To be successful, the KoboClicker will have to build on these strengths while addressing the problems discussed below.

[1] For the latter two cases, aid workers will count their footsteps when walking the perimeter of a camp or between landmarks and use the reading from the tally counter to back-calculate distances based on their foot length.

A major weakness of conventional counters that our team seeks to address is their inability to collect multiple data points at once. According to the UNHCR, five main data points related to demographics are collected in most rapid needs assessments: 1) age, 2) gender, 3) ethnicity/race, 4) household composition, and 5) special needs or the presence of vulnerable groups [61]. Having different aid workers talk to the same individuals to collect these data points introduces a redundancy that costs valuable time in the critical first days after deployment. With a multi-metric counter like the KoboClicker, the data collection efficiency of each aid worker increases several times, initial reports can be generated faster, and more personnel can spend their time executing on aid programs.

A second challenge with current tally counters is that they are not integrated with online data collection and processing tools. Aid workers must manually enter their tallies into platforms like KoboToolbox, which costs valuable time in the field and introduces transcription errors. Having a tool that populates data directly into KoboToolbox with the click of a button reduces errors and enables analysis and decision-making far more quickly.

Finally, because conventional clickers lack metadata, it is difficult to catch mistakes and validate results. Aid workers in rapidly evolving crises are prone to distractions while executing data-collection tasks, such as tracking the movement of refugees across a border. While a lapse in counting cannot be tracked by current counters, timestamped data on the KoboClicker could detect a significant deviation from the average click rate and alert the user to a potential error. Similarly, accidental clicks could be detected and undone on the KoboClicker without having to reset the counter completely. Generating metadata allows mistakes to be caught and helps workers have greater confidence in the results of their rapid needs assessments, which inform the entire aid strategy. Importantly, timestamping data points enables extrapolation, which is valuable for estimating results such as the number of people crossing a border daily. Metadata opens a new frontier in data processing, yielding valuable insights.

VII. Design and Testing

A. Initial Design Considerations & Challenges

After we decided on pursuing KoboClicker as our project, we worked on laying out some foundational considerations and challenges for the mechanical, electrical, and software aspects of the project. These considerations served as the launch point for initial design brainstorming and our technical design specifications. Therefore, the details below around the design and components of the clicker mechanism and the associated software are preliminary and are included to provide insight into the initial design process. Later sections will detail our comprehensive sets of technical specifications and design processes.

Mechanical Design Considerations

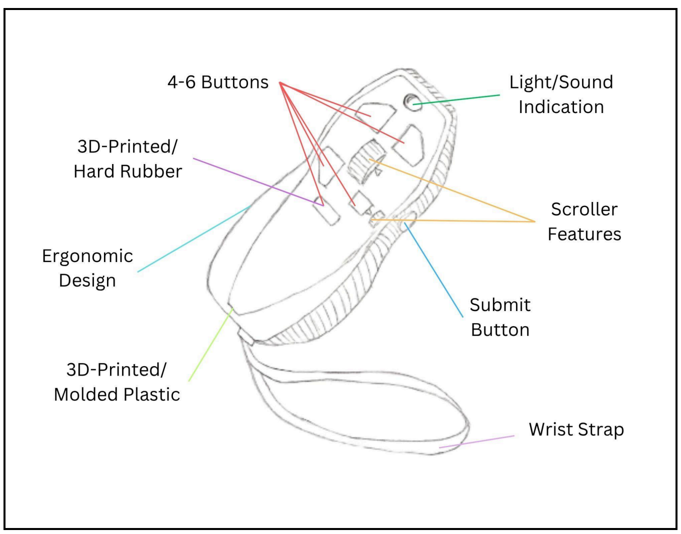


Figure 4. The mechanical design of KoboClicker resembles a small handheld remote or gaming controller made of 3D-printed filament or molded plastic. The clicker is worn on the wrist and has four to six buttons, along with a few scroll features. Once the associated input components have been toggled for a data entry, the user can press the submit button, which will be accompanied by a light, sound, and/or vibrational indicator.

The mechanical design of the KoboClicker consists mainly of the exterior components, including the physical body, the buttons, and other non-electrical components. The clicker is handheld and can be operated with one hand, and the shape is envisioned as similar to that of TV remote controls, single-handed gaming controllers, or ergonomic mice. Because of the repetitive button-clicking and the overall intensity of collecting data in the field, the design should be ergonomic so that KoboClicker is comfortable to hold and use for long periods. To allow for multitasking, the clicker would be attached to a wrist strap (Fig. 4), allowing the user to drop the clicker if they need to free their hands up to do another task, such as jotting down notes on paper, and then easily pick the clicker back up to resume usage. Having the strap looped around the wrist also serves as a deterrent to theft.

To enable data collection, the clicker will have between four and six programmable buttons (Fig. 4), depending on the categories of data the user hopes to collect. Some potential categories could include demographic variables such as gender, age range, injury status, and the number of children with an adult. KoboClicker could also include other input features, such as scrollers (Fig. 4), which are more useful for more quantitative data inputs, such as weight or age ranges. To specify the range for a data entry, the scrollers could have sections in different colors, and each color is programmed by the user to indicate a specific range of values. Thus, the user can scroll to the associated color section, and when they click the submit button on the side of the clicker, that specific value range – as well as any other buttons that were pressed – will be recorded under a single data entry. Hitting the submit button after each data entry will be accompanied by a light flash, a sound indication, and/or a slight vibration to signal to the user that the data entry was recorded (Fig. 4). Including different submit indicators will ensure that KoboClicker is user-friendly and accessible to users with potential impairments.

When considering the material of the mechanical components, the comfort and feel of holding the clicker and using the input components, and the clicker's cost and ease of manufacture must be considered. Another consideration is the material's durability and waterproofness, as the clicker will likely be exposed and used across multiple crisis settings. Therefore, for the clicker body, we may use 3D printing filament or molded plastic (Fig. 4), and the material of the buttons or scrollers could also be 3D printing filament or hard silicone rubber (Fig. 4), as the soft rubber of the TV remote buttons often gets worn down. These specifications must also consider weight as a parameter, as the clicker should be heavy and sturdy enough to withstand breakage and fit within a weight range that feels comfortable for a person to hold for long periods of time [54].

Mechanical Design Challenges

One challenge in the mechanical design of the KoboClicker is the shape of the clicker body, which must enhance user comfort while providing sufficient interior space to accommodate all electrical components. To navigate this challenge, the initial designs will likely draw on existing device designs, mentioned in the previous section, as well as on other ergonomically minded devices. Another consideration is the cost and manufacturing of the clicker, both in terms of ease of creation and replicability for HHI. The design should be affordable and feasible for manufacturing so that they can deploy them for future use. These considerations also play into the electrical components. Still, for the mechanical portion, we can optimize materials to keep costs as low as possible and keep the manufacturing process as simple as possible.

Additionally, the clicker must optimize the number of buttons to give the user more flexibility, allow more data to be collected, and simplify the interface so it is not too confusing to use. Before deciding on the number of buttons and other potential input options beyond buttons and scrollers, we will meet with HHI to receive their input, such as on the most valuable data categories that researchers and other data collectors record, especially for rapid needs assessment (as that is a common context for using a manual clicker). Implementing different hotkey functions could also be an option to increase the number of inputs available.

A sub-challenge with the buttons is the possibility of the mechanical components failing. If the body is damaged, a small failure will likely not be detrimental to continued function; however, if the button interface fails, especially at the interface between the physical buttons and the button circuitry, this would impair or completely compromise the clicker's functionality [54]. Anticipating and addressing potential failures is a consideration across the mechanical, electrical, and software aspects of the clicker. One idea is to configure the clicker body so it opens easily, allowing the user to examine the circuitry with minimal tools. A test procedure could also be developed in which the user, before taking the clicker onto the field, inputs a few test data entries using the various buttons and scrollbars and checks whether the entries are recorded accurately.

Electrical Design Considerations

The KoboClicker must be both durable and long-lasting to function effectively in field conditions. For this reason, the electrical design of the KoboClicker is kept as simple and lightweight as possible to maximize battery life and reduce the likelihood of internal errors. The prototype will run on a small programmable microcontroller, such as an Adafruit development board, allowing easy modification of both the software and hardware. Many of these smaller devices operate using 5 V logic rather than the standard 3.3 V logic. This means that a high signal (or logical "1") is in the range of 5 V, while a low signal (or logical "0") is ground. While this could, in theory, affect power consumption, the small size of the board and the resulting reduced current draw will most likely offset this concern. The microcontroller has a limited number of pins available for digital logic input from peripheral devices. Each button or scroll wheel will be connected to a designated pin on the microcontroller such that it remains low by default and transitions high when a button is pressed. The few output devices, such as LEDs or sound indicators, will be connected in parallel to the same microcontroller pin. This approach reduces the total number of pins required, which is a major constraint when using small microcontrollers, while still maintaining appropriate voltage across each device.

Another critical aspect of the electronics is the inclusion of two additional integrated circuit components: the Bluetooth module and the microSD card reader. Both components must communicate data to and from the microcontroller. To minimize pin usage, these devices will communicate using the I2C protocol. The secondary devices are connected to the same two communication lines, and each receives only the data intended for its specific address. The first line is the clock signal, which allows for synchronization of data transmission. Simply put, the clock is a square wave. The second line is the data line, which transmits packets of data to the downstream devices, beginning with the device’s I2C address.

Finally, the device needs a long-lasting, easily rechargeable battery. Another major consideration is the cost, weight, and space. Batteries are often chosen based on the latter factors mentioned. Testing will be required to find the right balance between a long-lasting battery that fits in the KoboClicker and keeping costs low. For the prototype, a lithium-ion polymer battery seems to be the best option. This allows for an easily integrated recharging circuit and native support for microcontroller connectivity. In addition to being rechargeable, for electrically lightweight projects such as this one, a 4Ah battery may last for days without recharging.

Electrical Design Challenges

Many of the electrical design challenges stem from balancing size, device cost, battery life, and other inherent device limitations. The final balance will only be determined from the design and testing process throughout the rest of the semester.

The internal electronics are limited in size by the physical device design. This is the first component in the balance. As mentioned in previous sections, the KoboClicker is a small, handheld device, leaving minimal room for internal components. Much of this space will need to be taken up by the battery to ensure a long-lasting battery life on a single charge. Additionally, the wiring and spatial arrangements of the three boards will need to be carefully mapped out in the mechanical design process to ensure that all essential electronics fit properly. In addition to being able to fit, the inside elements also need to be easily accessible. In rugged field conditions, internal components are bound to break or malfunction. Without easy access to the internal components, the KoboClicker would be useless. Thus, the internal electronics need to be designed with repair in mind.

As outlined by HHI's parameters, cost is a major factor in the device. While the prototype will cost more because it lacks a scaled manufacturing process and facility, the end-stage design must be cheap and mass-producible. This poses a significant challenge for selecting electrical components and ties into battery selection. The most expensive element of the device will most likely be the battery. To ensure the correct battery is chosen for the device, battery tests and comparisons will have to be conducted with size, cost, battery life, and durability as factors.

There are also a few more minor limitations that pose challenges to the larger design. Given the size of the microcontroller required, there are significant limitations on the number of input and output devices that can be used, specifically, the number of buttons available to track survey data. This also means there are speed limitations in inter-device communication under the I2C protocol. Finally, there are many challenges in what we do not yet know about electronic components and essential protocols, including the actual I2C design and how to effectively integrate Bluetooth into the system. Challenges with the electrical design will continue to appear and be addressed as the design process continues.

Software Design Considerations

At the start of the project, the team defined a set of specifications to ensure that the KoboClicker software subsystem supported reliable, secure, and repeatable data collection. The core requirement was the ability to construct structured rows of button data with consistent formatting and timestamps. Each row needed to follow a fixed-length binary format that could be exported directly into CSV for downstream use. The firmware had to detect button presses, assign timestamps, and provide a straightforward method for discarding incorrect entries. On the Android side, the system needed to receive these structured rows, store them in persistent memory, visualize them in a tabular layout, and export them for external analysis or uploading to KoboToolbox. Both real-time Bluetooth transmission and offline SD card logging were specified as essential from the beginning.

Performance and scalability goals were also established early. The microcontroller needed to buffer a large number of rows without loss, even if the Android device was not actively connected. Bluetooth throughput needed to support the rapid transmission of rows at the pace at which users would collect data. CSV export on the Android device needed to scale to large datasets while remaining fast and predictable. The application also needed to support multiple sessions or files without risking data overwriting, and its internal data structures needed to support at least five core columns, with the possibility of expansion in future versions.

Security requirements guided several foundational choices. Each row transmitted from the microcontroller needed to include a token so that the application could confirm that the data originated from an authorized device. The KoboToolbox API uploader required authentication using secure API tokens, and these credentials needed to be protected within the application. The software also had to prevent more than one device from connecting simultaneously to avoid mixed data streams. These measures ensured that data would maintain integrity and authenticity throughout the collection and upload process.

Integration with KoboToolbox sets additional requirements. The CSV schema had to match expected field formats and naming conventions. Timestamps needed to be converted to ISO 8601 format during API upload to ensure compatibility with KoboToolbox. The GUI-based CSV to form-field mapping tool had to be easy to use and capable of supporting any valid KoboToolbox form. The API uploader needed to run as a standalone tool on both Windows and macOS, allowing cross-platform deployments without installing additional dependencies. Several assumptions also shaped the initial specifications. Bluetooth was expected to be the primary method for real-time data transfer, and the Android emulator was assumed to be sufficient for early development before hardware became available. SD logging was expected to support offline workflows where Bluetooth might not be feasible, and fixed-length binary rows were assumed to simplify both processing and data consistency. These assumptions resulted in a specification focused on reliability, simplicity, and adaptability across a wide range of field scenarios.

Software Design Challenges

The software design for the KoboClicker presented a set of challenges rooted in the need to balance simplicity, reliability, field usability, and strict technical constraints across both the microcontroller and the Android application. From the beginning, the team needed to balance reliability with simplicity, Bluetooth latency with battery life, offline SD logging with real-time streaming, and data integrity with the limited resources of a small embedded device. These competing priorities required disciplined architectural choices that supported a straightforward user workflow while still meeting high standards for consistency and fault tolerance.

Uncertainty also shaped the early stages of the design process. It was unknown whether Bluetooth Low Energy would remain stable in the field, where interference, distance, and device variability can disrupt communication. It was also unclear whether an Android app could reliably parse, store, and display incoming data in real time. Compatibility with KoboCollect or KoboToolbox was another open question because no official pipeline existed for sending structured hardware-generated logs into either system. Finally, the number of characteristics an enumerator might need to track was not fixed, so the software needed to be flexible enough to scale beyond the initial format without losing performance.

Several practical constraints were present from the start. The team initially had little or no access to the final hardware and had to simulate Bluetooth data to begin development. Android imposed strict permission requirements on scanning and connecting to BLE devices, especially on newer versions of the operating system. On the firmware side, the microcontroller had minimal RAM, which required efficient buffering and lightweight string handling. Field users would not be able to navigate complex menus, so all logic needed to follow the simple NEXT, ERROR, and POWER workflow. Security considerations also shaped early decisions because the system needed to validate each row received from the device and prevent unauthorized Bluetooth devices from interacting with the app.

Reliability expectations added further difficulty. From day one, the system had to guarantee that no data would be lost, even if Bluetooth disconnected or the device reset unexpectedly. Each transmission needed confirmation before the row could be removed from the buffer. The firmware had to recover gracefully from partial writes and power interruptions, while the Android app needed to remain responsive at all times and handle growing datasets without slowing down. Together, these requirements demanded a software architecture designed for resilience and stability in the variable conditions expected in humanitarian field contexts.

B. Gantt Chart

To effectively manage and plan for each stage of the engineering process and the steps required to construct a deliverable, we created a Gantt chart with six stages. Many of the stages in this chart are discussed in detail in their own section of this report. This chart outlines the stages and subtasks assigned to individuals or groups. The Gantt chart was an essential tool for tracking progress on subtasks, monitoring the status of dependencies (tasks that must be completed to move forward on other tasks), and dividing group work to create tangible deliverables. Stages one and two are our investigation stage and ideation stage, which worked on developing an understanding of HHI as well as identifying a problem to be solved. Stage three is the design stage, where each subteam (mechanical, electrical, and software) developed technical specifications and initial designs for the device. Stages four and five are the initial prototyping stage and the iteration stage, respectively. In these stages, the device was created, tested, and then recreated multiple times with different form factors, features, and specifications. Finally, our last stage focuses on the delivery of our solution–our device, our presentation, and this report. Taken together, these stages illustrate how an initially broad problem over an extended period of time was transformed into a refined set of smaller problems to focus on. The Gantt chart not only outlined this progression but also helped maintain accountability and momentum throughout the project. Its role in guiding our time was central to delivering the polished products.

C. Mechanical

Design Analysis

When first approaching the design process, the mechanical team laid out practical and supplementary considerations for the design and fabrication of the clicker shell, button interface, and internal housing components. For the exterior shell and buttons, we wanted to ensure the device was comfortable to use for long workflows and high-volume data entry, and resilient to potentially harsh climatic and weather conditions, including moisture, dust, and extreme weather. We also wanted to ensure that the exterior of the clicker would be non-reactive with common clinical reagents, as emphasized during a meeting with HHI, especially in times when the crisis is tied to a disease or outbreak, any equipment would have to be easily sanitized. As for the internal housing, we aspired to create a complementary structure for the electrical features to properly secure them and provide sufficient durability to prevent them from breaking. However, the internal buildout would still provide easy access to necessary electrical features, such as the charging port and SD card slot, and would enable simple assembly and disassembly if an electrical component, such as the battery, needed to be accessed.

Final Technical Specifications

The mechanical design considerations of the KoboClicker focus on ensuring durability, reliability, and user comfort while operating in demanding environments. To quantify these considerations, a list of mechanical technical specifications was developed to guide design decisions throughout the development and testing stages.

These specifications were organized into two primary categories: non-negotiable mechanical specifications (Table 7) and negotiable mechanical specifications (Table 8). The non-negotiable specifications represent critical requirements that contribute directly to the core functionality, safety, and longevity of the device, and therefore remain fixed throughout the design process. The negotiable specifications were considered secondary priorities and allowed for flexibility during design iteration, optimization, and trade-off analysis, provided that the non-negotiable requirements were satisfied.

Table 7. Non-Negotiable Mechanical Specifications

Function	Specification
Water Resistance	High-pressure water resistance (IPX-6)
Actuation life	10 M+ clicks
Drop Rating	Withstand 1000N Impact Force
Thermal Resistance	IEC 60068-2-1
Weight	< 140g
Number of Inputs	8
Disinfectant resistance	Non-reactive with common disinfectants
Size	9-16 cm x 4-6 cm x 1-3 cm (l x w x h)

Table 8. Negotiable Mechanical Specifications

Function	Requirement
Serviceability	< 2min
Tactile Feedback	ISO 9241-9
Button Size	ISO 9241-9

Button Separation Distance	ISO 9241-9
Ergonomics Balance	Center of gravity within 25 mm of grip center
Ergonomics comfort rating	>4/5
Vibration Resistance	IEC 60068-2-6

To determine the estimated energy for the minimum amount of force that the clicker would most likely go through, we reasonably calculated the impact energy and associated force for a drop of the device from a 3 foot tall handheld position (Fig. 5).

Impact energy:

$$E = mgh = (0.111\,\mathrm{kg})(9.81\,\mathrm{m/s^2})(0.91\,\mathrm{m}) = 0.989\,\mathrm{J}$$

Estimated force (worst-case):
A conservative stopping distance of $d = 1\,\mathrm{mm} = 0.001\,\mathrm{m}$ gives:

$$F = \frac{E}{d} \approx \frac{0.989}{0.001} = 989\,\mathrm{N} \approx 1000\,\mathrm{N}$$

Thus, a force of approximately $1000\,\mathrm{N}$ is a reasonable lower bound for a 3-ft drop.

Figure 5. Impact force calculation for the drop of the KoboClicker at a height of 3 feet.

Prototype Design Process

The prototype design process was guided by an initial survey of existing handheld clicker devices to identify the structural and functional elements that enable lightweight construction, reliable actuation, and ergonomic usability. These findings informed our design constraints and established baseline expectations for device size, durability, and internal component organization.

The development process follows the three main elements of consideration for the physical design: (1) outer shell design to integrate the inputs interface and allow for comfortable user experience, (2) designing functional and feasible methods for joining components to achieve reliable assembly and manufacturing strategies, and (3) designing reliable internal structures allowing for mounting and considerations for tolerancing, wire routings, and structural reinforcements. These stages manifested in the three-part integration, focusing on the shell (external form factor), the buttons (mechanical actuation between the user and the electronics), and the internal mounting structures (security features for the electronics). The following sections outline the design decisions, trade-offs, and performance characteristics associated with each design strategy for each component that informed the final prototype, and provide an overview of the final design choice.

Clicker Shell Background Research

Background research on existing handheld devices was conducted to identify the ergonomic factors that influence user comfort and reliable actuation, as well as the mechanical strategies that enable these devices to function effectively. This benchmarking approach allowed us to identify repeatable design patterns rather than replicate specific products. From an ergonomic standpoint, we observed that successful handheld interfaces consistently center their primary inputs within the natural range of motion of the thumb and index finger. Devices that combined top-facing and side-mounted buttons demonstrated improved reachability and reduced the need for hand repositioning, while layouts that relied on ring- or little-finger actuation proved unreliable [65]. This research also underscored the importance of maintaining a sufficiently large front face to support intuitive targeting and of designing button distributions that remain usable regardless of handedness.

Separately, the functional analysis of these devices revealed practical engineering strategies for packaging electronics within compact geometries. Internal retention was commonly achieved through cylindrical standoffs, compression features that secure the PCB, or compliant silicone overlays that distribute loads and maintain electrical integrity. In particular, guidance for how to allow functional aspects to key components that often need servicing, such as battery packs, and components that might otherwise be exposed to the weather, such as micro-SD ports, was particularly useful for generating ideas on how to protect these regions while still allowing accessibility. Inspiration for allowing spaces for internal volume, not obstructing wiring paths and button travel, was also noted. We also noted that devices marketed as water-resistant typically rely on controlled seam geometry, minimized openings, and gasketed interfaces rather than thickened walls or complex enclosure shapes. Various fastening and sealing methods, including screws, snapfits, and elastomeric membranes, were applied selectively. Screws were often used to assemble components such as cylinders, pushing the PCB into place. Snap fits were used for the coverings over battery packs. Elastomeric membranes were cast and integrated to provide a softer interface with the top-facing buttons on the remotes.

These learnings provided a valuable reference for our own structural interface decisions, particularly in guiding which structure-mounting features to explore and in selecting assembly methods. However, the ergonomic principles identified in the background research remain primary and represent a significant opportunity for refinement in future prototyping iterations. As the internal architecture matures, these ergonomic considerations should guide further optimization of button placement, hand fit, and overall device geometry.

Shell Design

In developing the outer shell, our objective was to balance user interaction with the functional requirements of housing the electrical subsystem. To support rapid, collaborative development, the shell was modeled in Onshape, enabling simultaneous CAD iteration and reliable version control. Simultaneously, each iteration was fabricated on either a Prusa MK4S or

Bambu Lab X1-Carbon FDM printers, using general purpose PLA, to verify dimensional accuracy and internal clearances and validate overall assembly interactions. This workflow produced four sequential shell prototypes that reflected evolving design priorities. The first two, based on stopwatch and VR-remote geometries, emphasized ergonomic exploration and detailed surface shaping, while the electrical components were still in flux. As component sizes increased and assembly constraints from the other subsystems became clearer, we shifted our priorities to integrating critical assembly features to low-fidelity prototypes based on basic geometries - a coffin-shaped cylindrical shell and a rectangular shell - that prioritized internal surface area, assembly feasibility, and internal mounting considerations. Collectively, these prototypes established the design envelope for the final housing and informed the mechanical strategies adopted in later iterations.

Stopwatch Design

The stopwatch-inspired shell evaluated a circular housing with a large front surface area and distributed rim geometry (Fig. 6), allowing simultaneous use of the thumb and up to three fingers for actuation. This configuration provided flexibility for both top and side-mounted button placement and offered a favorable sealing boundary due to its continuous edge profile. In particular, after testing, the geometry's form factor naturally emphasizes top-facing button placement, where thumb actuation and spacing are most effective. The enlarged circular footprint also afforded generous PCB area, enabling exploration of more complex button layouts (Fig. 6). Ergonomic testing indicated that diameters in the 67–68 mm range accommodated a broad range of hand sizes and allowed stable multi-finger control. However, scaling the enclosure to accommodate the rectangular breadboard and associated wiring significantly increased depth, reducing controllability and preventing comfortable single-handed operation. Central buttons became difficult to reach, and the increased mass distribution compromised stability during extended use. The circular form also proved inefficient for internal packing of non-cylindrical components. While the geometry demonstrated clear ergonomic benefits for low-button-count devices with shallow internal architectures, it was ultimately incompatible with our component stack-up and one-handed usability requirements.

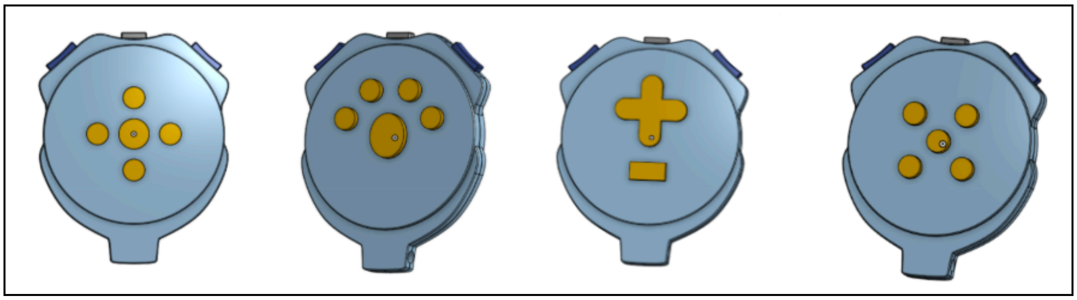


Figure 6. Stopwatch design. Pictured are four different button configuration possibilities.

VR Remote Design

The VR-remote–inspired geometry (Fig. 7) examined an elongated shell with sculpted exterior surfaces designed to align with natural hand contours and improve stability during prolonged use. This form enabled intentional placement of top- and side-mounted buttons (Fig. 7) that offloaded frequent actuation from the thumb to the index and middle fingers, consistent with human-factors principles. The design provided improved grip quality over the stopwatch model and allowed more controlled finger access across the device's length. However, the nonuniform exterior geometry required high printing precision that our Prusa could not meet. It exhibited limited tolerance to scaling, with each adjustment to internal cavity dimensions disrupting external contours and button alignment. Additionally, maintaining uniformly flat seating locations for button interfaces proved difficult across curved surfaces. Although this geometry achieved strong ergonomic performance and remains the most promising candidate for a future user-oriented iteration, its sensitivity to dimensional variation and limited internal packing efficiency made it unsuitable for rapid prototyping under evolving electrical constraints.

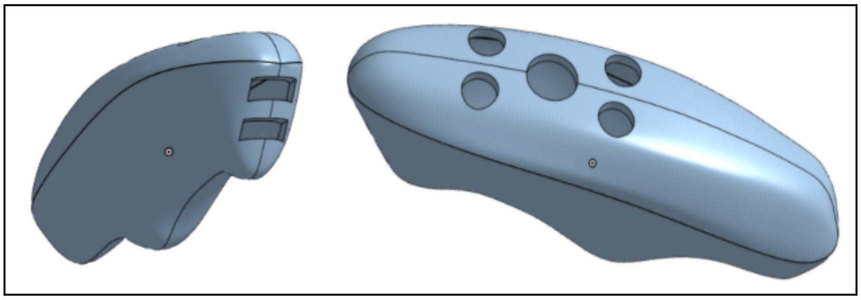


Figure 7. VR remote design. Top isometric view (left). Bottom Isometric view (right).

Coffin and Cylinder Design

While working on an ergonomic design and developing the final form of our product, we wanted to create a simplified design to iteratively prototype from and use for integration with the electrical subteam. As a simplified model of the stopwatch design, we developed a cylindrical shell geometry. The coffin design was created as a simplified model of the VR remote-inspired design (Fig. 8). While we continued to maximize the ergonomics of the VR-inspired shell exterior, we also wanted a simple geometry to begin integrating with the clicker's electrical components and give us a better understanding of the physical dimensions. The simplified design made it easier and faster to print. It gave us a feel for the general shape and what it would be like to hold. We placed stickers over the shell to represent buttons to compare button placement layouts. It was also helpful in beginning to integrate with the electrical team, as they were deciding which electrical components to use and where to place them within the shell. It also helped us develop an attachment strategy for fixing the shell base to its lid. We used three screws placed around the edges of the shell to better understand the tolerances of 3D-printed PLA and the material's strength. The sharp edges were uncomfortable to hold. With the stopwatch-inspired cylindrical design, it was challenging to fit the necessary electrical components into the shell, given the breadboard's dimensions. The stopwatch inspired the cylindrical design. The VR remote inspired the coffin design.

We placed stickers in various configurations on the shell to simulate potential button layouts. We wanted the buttons to be comfortable to press, as well as intuitive for the user, so that the placement made sense with the function. After discussing comfort levels of different configurations with the broader team, we decided that the most ergonomic and user-friendly layout involved one button on the front face of the device, three on the side, three on the top, and one towards the base.

Based on this first printed model, we were able to test screw tolerancing for 3D printing. When printing with PLA, we determined that an additional 0.5mm was needed to allow the screws to fit into our print. We also determined an ideal wall thickness for our shell. While 1mm was too flexible and flimsy, thicknesses above 2mm were appropriately rigid.

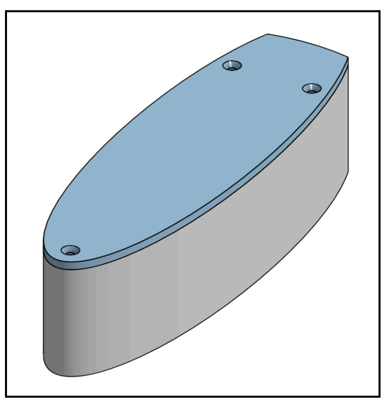


Figure 8. Coffin and cylinder design from a top isometric view.

Simplified rectangular design

Once the electrical team had decided on the prototypical components they would use for our final design, we decided to create a scaled-up, simplified version of the form we had previously envisioned. Since we had decided on an oblong design more like the VR remote, we modeled a simple rectangle sized to fit the long rectangular shape of the breadboard (Fig. 9). The simplified geometry made the shell much quicker and easier to model and print since it did not involve any organic geometries or supports while printing. Scaling up the shell was crucial to integration with the electrical subsystem, since the components they used to prototype the device would not have fit into our envisioned shell. The design was much bulkier than intended and not very comfortable to hold.

We wanted to model and fabricate the simplest possible geometry that would still roughly resemble the device's final form and accommodate the necessary electrical components. Since the electrical components (breadboard, battery, microSD) were all rectangular in shape, the natural choice was to build a rectangular shell. This was also easier to print and rapidly prototype than a more oblong/organic shape, as it was less likely to fail during printing and required fewer support structures.

To test electrical functionality within the physical shell encasement, we decided to simplify our button layout and focus on top-face-mounted buttons for our prototype. From this model, we discovered internal spacing issues, mainly stemming from the need to use jumper wires to connect electrical components. This allowed us to adjust our electrical housing structure and add additional space for wiring. We were also able to test the functionality of our top-mounted buttons, the electrical push buttons, and the corresponding software.

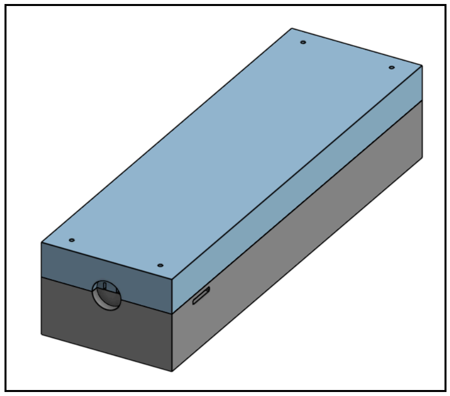


Figure 9. Simplified rectangular design from a top isometric view.

Button Caps and Mounting

Button Caps

The original plan for the buttons was to buy button caps from online vendors; however, this plan was discarded because each commercial cap typically worked only with a specific associated button, rather than being universal across multiple button variations. For those buttons, most required a high-volume order, so after determining that pre-made button caps would not be feasible given our time frame and budget, we wanted to create our own 3D-printed button caps that would fit over the electrical switches we had used. We looked at existing designs for button caps and modeled ours on them using a general purpose PLA filament. These designs used two cantilever snaps extended from the bottom of the cap and two support columns to grip the switch (Fig. 10). Unfortunately, the print quality from the Prusa MK4S and the Bambu Labs X1-Carbon using PLA prevented us from applying the cantilever design to our device, as the caps would not reliably or consistently fit the switches. The same model would produce prints that sometimes fit the electrical switches, but sometimes were too loose or too tight.

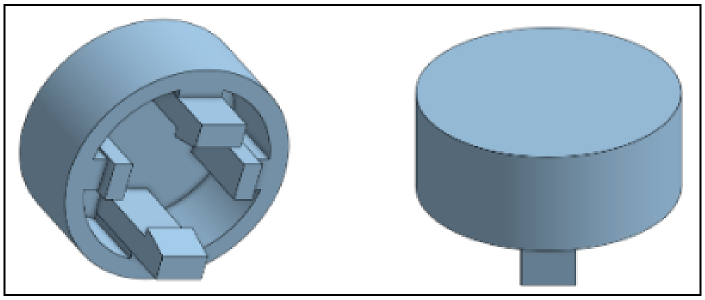


Figure 10. Button caps. Bottom isometric view (left). Top isometric view (right).

Button Mounting

Originally, when we were still pursuing the snap-fit button caps design, we were planning to mount the buttons to the inner surface of the clicker. That way, the location of the button and the holes would not have to be aligned perfectly, as the button location would be set directly underneath the hole for which the button cap would emerge through. We went through several iterations of button mounting mechanisms, for which can be assigned into two categories. The first was a snap fit mechanism, which consisted of four hooks encircling the perimeter of the button hole (Fig. 11). To mount, the button would be pushed past the hooks so that the bottom of the button box sat on top of the hooks. However, finding the accurate dimensions for the distance between the hooks as well as the length of the hook itself was difficult, because if the hooks were spaced too far apart, the button would not be supported by anything and would just fall right back through the hole. However, if they were too close, the hooks would not have much bend before they would snap when trying to push the buttons through.

Therefore, the snap fit mount was discarded, and a second category of button mounts were pursued. This category consisted of variations of a basket support for the bottom of the button, and would be secured after the button is already put into place. As a first iteration, a slim extrusion was modelled so that the middle portion of the extrusion would support the bottom of the button box, but the sides of the extrusion contained small cavities so that the button cap could slide downwards when force is applied. Therefore, the movement of the cap without moving the button box would cause the button to click. We then doubled the extrusion to create a thin “X”-shaped basket (Fig. 11), which worked with the button when we held the extrusion in place, but we found that the basket was too thin and would snap when we attached it to the clicker shell with superglue. A few variations were made to try and solve this problem, including thickening the legs of the basket, creating a screw-in securement mechanism to attach the basket to the shell without superglue, and using screws and nuts to secure the basket (Fig. 11). However, unique challenges arose for each design variation, such as incorrect button components being in contact with the basket or the wires of the button interfering with the button mount.

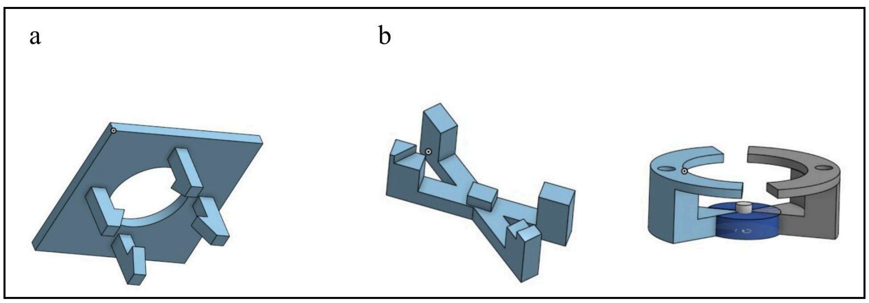


Figure 11. a) The snap fit mount design consisted of four hooks encircling the button hole. b) On the left, the X-shaped basket mount has a small ledge in the middle to support the bottom of the button box. On the right, parts of the basket mount have been filled in and extended to provide more surface area contact with the clicker shell. Screw holes have also been added.

Cylindrical Push Buttons

Ultimately, due to time constraints and the problems detailed above with the button caps, we chose to mount the buttons directly to the breadboard. Instead of using button caps that enveloped the entirety of the button box, extended cylindrical pillars were adhered to the push buttons using J-B Weld epoxy adhesive, which binds well to PLA-printed parts (Fig. 12). This direct connection ensures that the force transmitted to the top of the cylindrical pillar transfers down to the button. The simplicity of this design made it easy to fabricate, as it required less precise dimensioning than the two categories of button mounts considered above. This also mitigated alignment issues arising from variations in wiring thickness or slight shifts in component placement. However, the cylindrical pillars cannot be used for side-facing buttons because they must extend perpendicular to the breadboard surface (however, the slot-fit design below presents a workaround to this design constraint). Yet, the cylindrical pillars are pretty reliable and require minimal assembly, so the advantages of utilizing them for the top-facing buttons significantly outweigh any drawbacks.

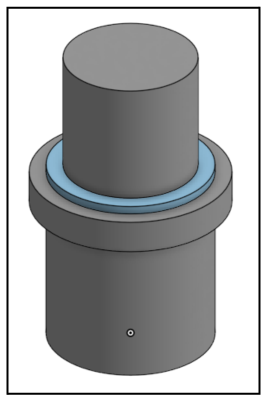


Figure 12. The cylindrical pillars rest on top of the buttons, allowing for the button to be pressed when the user presses the top of the pillar. They have a thin slot for the O-ring to prevent water seepage.

Slot-Fit Side Buttons

Our design initially involved multiple side-mounted buttons, which we would hope to implement on future iterations of the device. We developed a slot-in method to mount buttons to the device's side faces (Fig. 13). The two prongs on the electrical push button would slide into two slits at the edge of the shell's base, and the top of the shell would hold the button in place, stopping it from sliding out.

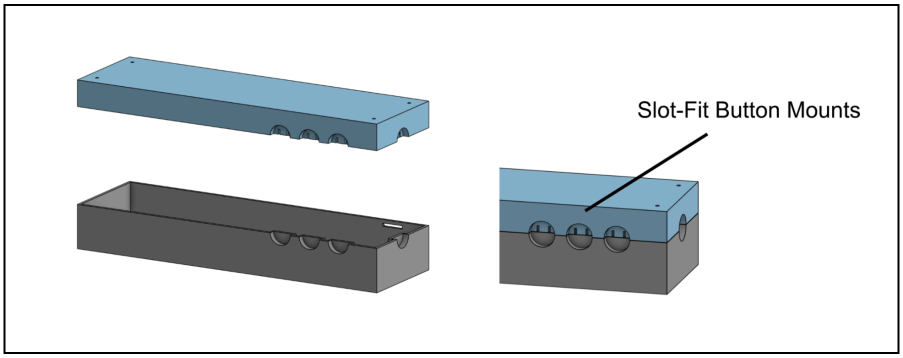


Figure 13. Rectangular shell design with slot-fit button mounting. Pieces not assembled (left). Assembled (right).

Assembly Considerations

Our main objective was to satisfy our design requirement that the device could be serviced and taken apart with one tool in minimal time, and put back together intuitively.

Base-Lid Assembly Assembly Design

The assembly method for the lid, base, and internal mounting frame was a critical aspect of the housing design. The mechanism needed to enable repeatable alignment of internal components, secure the electronics, and facilitate straightforward assembly during prototyping. Two primary assembly strategies were evaluated: (1) screw-based joining methods and (2) a snap-fit system between the lid and base.

Screw Based Assembly

Our initial assembly concept relied on mechanical fasteners to join the lid and base. This method was pursued for its simplicity and for its potential to integrate electrical alignment into the enclosure itself. Two variations of this screw-based approach were developed.

Direct Screw Columns

The first method incorporated vertical screw columns printed into the lid and base. Screws would pass through the top lid and engage with embedded columns in the base, clamping the enclosure together. This configuration was easy to manufacture and assemble, but it required tight tolerances on wall thickness and column diameter. Small deviations during printing led to misalignment, compromising thread engagement and sealing.

Screws Through the Breadboard Mounting Holes

The second method utilized the breadboard's inherent 3 mm through-holes to align both the internal electronics and the exterior housing. In this configuration, screws passed through the breadboard and into the base, with the lid fastening down over the same screw path. The intent was to use the breadboard itself as a datum, ensuring that electrical components remained aligned with the enclosure. However, this introduced multiple interdependent tolerances where the screw path needed to pass precisely through the breadboard holes, the lid and base needed to align within fractions of a millimeter, and the screw length needed to be able to extend minimum values of over an inch and a half (38mm) with screws with thread diameters that were smaller than 3mm, which were less commercially available in lengths that long.

Among the screws available during prototyping, #4–40 fasteners were the only viable option capable of meeting the required depth and head geometry. Even with appropriate screw selection, the reliance on the breadboard as an alignment feature created low margins for positional error. Minor deviations in print accuracy, wire routing, or component height frequently led to misalignment, complicating assembly and requiring repeated adjustments. For these reasons, screw-based assembly was ultimately deemed unsuitable for the final design.

Snap-Fit Lid and Base

To eliminate the compounding tolerance dependencies of the screw-based configurations, the final assembly method adopted a snap-fit interface between the lid and the base. The lid was designed with a recessed channel, and the base slides upward into this recess until it seats securely in place. The snap-fit geometry provides a controlled deformation during engagement, enabling reliable closure without the need for fasteners.

This design offered several advantages from the problems that arose with screws. These included reduced dependence on strict screw alignment and requirements, simplified user assembly by requiring no tools, and improved internal flexibility to allow the frame and electronics to float slightly before snapping into place.

The snap-fit mechanism also allowed the lid and base to be molded with consistent wall thicknesses, reducing print variability and improving structural integrity across the enclosure. The design was validated through repeated assembly cycles, demonstrating repeatable engagement without material fatigue in the PLA components.

Mounting Electrical Components

The housing for the electrical system was designed in multiple iterations with reliable component mounting, mechanical load protection, and accessibility for the system maintenance at the forefront. The final design of a loaded grid structure utilizes a combination of a redesigned PLA grid for the primary electronics and adhesive mounting for auxiliary components. These decisions were driven by the need for mechanical robustness and compatibility with variable electronic layouts while trying to determine the footprint of KoboClicker. Each iteration refined the internal mounting strategy based on manufacturability constraints and evolving electrical layouts, necessitating flexible component placements.

Integrated Grid Base

Initial housing methods incorporated an integrated PLA grid measuring 53 mm × 183 mm, designed as an extension of the base. This grid reinforced the enclosure and provided a continuous surface for mounting electrical components, with the goal of distributing force throughout the remote. The grid also included recessed extrusions, also supported by grid functionalities, to restrict the movement of free-floating electrical components, the battery, and the SD card. While structurally robust, the integrated grid limited flexibility as the electrical design evolved.

Layered Assembly

The second design adopted a more modular, layered assembly approach. The 53 mm x 183 mm grid was retained but was modified to be a separate layer from the existing grid, such that it could slide into the base/lid shell. This was intended to enable rapid printing of the device and lower the failure rates associated with building the full print through FDM PLA 3-D printing. The layered assembly enabled more rapid prototyping of height changes.

The layered assembly approach was also modified to leverage the breadboard's inherent 3 mm screw holes, enabling direct, secure mounting to the grid. To position the free-floating electronic components, the precisely dimensioned extrusions were maintained on the grid. These features fixed components in the desired locations and ensured stability during operation.

Open Frame

The final frame design was developed following full-system integration of the breadboard, soldered electrical components, wiring harnesses, and shell. During this integration, the rigid grid geometry and extrusion-based component placements proved incompatible with the need for real-time adjustments to wiring thickness and component routing. To address these limitations, the grid was removed and replaced with an open frame sized to match the finalized breadboard footprint of 53 mm × 103 mm. A tolerance of 0.5 mm was maintained around all mounting points to ensure proper alignment with the breadboard's screw holes without interference.

The selection of mounting hardware also introduced constraints. The screws available on hand were limited in both head count and length range, requiring the fasteners to fit within a 2–4 head configuration and achieve a minimum embedded depth of 12.7 mm to secure the breadboard reliably. After evaluating screw diameters relative to the breadboard through-holes, surrounding frame wall thicknesses, and available thread engagement, 4×40 fasteners were chosen as the optimal solution. Ensuring proper tolerancing of the screw-hole width and spacing was critical, as overly thick walls could prevent insertion, while overly thin walls could cause structural failure during tightening. The finalized hole pattern provided consistent alignment while maintaining sufficient material around each screw for mechanical stability.

The introduction of open sidewalls in the new frame allowed wires to be maneuvered freely, reducing strain at soldered joints and preventing pinching during assembly. Local thickened regions were added in areas of low wire density to preserve mechanical rigidity while avoiding restrictions on cable routing. This final configuration provided the adaptability necessary for the evolving electrical system while maintaining the structural robustness required for prototype operation.

Waterproofing

To waterproof the clicker mechanism, we considered multiple options with varying levels of coverage. Mainly, there were two openings we wanted to ensure would not be prone to water seepage: the separation between the lid and the base of the clicker shell, and the gap between the clicker shell and the button caps. Initially, we planned to waterproof the device by creating a silicone casing for the shell, either lining the interior or the exterior of the clicker. However, we decided against creating a full waterproof casing because for both scenarios, cutouts would have to be made to certain parts of the casing for access to certain electrical components or portions of the shell, such as the charging or the microSD port. We also decided against an inner silicone casing specifically to protect the button interface as the design would be heavily dependent on the button mounting, and additionally, creating a mold for the silicone casing would also be time consuming.

Therefore, we looked into how similar commercial products are waterproofed, and after we disassembled a waterproof stopwatch, we gravitated towards implementing silicone O-rings for both the connection between the clicker lid and base, as well as around the rim of the cylindrical push pillars for the buttons (Fig. 14). The O-rings function by creating a seal between two surfaces, therefore blocking out any water that may seep into the cracks. For the O-rings, we decided to use two different casting methods. For the overall shell we designed a mold based on the dimensions of our device using onshape, and 3D printing the mold. For ease of removal we added a thin layered extrusion on top of the ring on the mold which we used an Exacto knife to cut off. We had a thin cavity surrounding the rim of the clicker body, and this is where we would situate the O-ring so that it created a seal between the two halves of the clicker body when they are joined together.

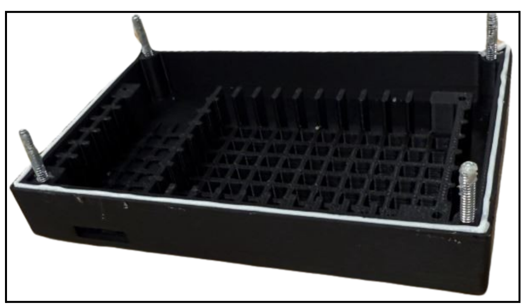


Figure 14. The O-Ring lays in a cutout along the rim of the clicker base and creates a seal when the base is joined with the clicker lid.

When waterproofing the buttons we took a different approach. When editing the buttons to become waterproof we cut out the 1mm thick and 1mm deep ring around the part of the button you press. For spacing of buttons, the ring had to be small, which made it difficult to use the method for the wheel and have it fit correctly. We decided to dip the buttons into the silicone, allowing it to populate the cut out, allowing for a perfect fit, then wiping excess off of the top. The silicone we used is called *Dragon Skin* which is the firmest silicon offered in the lab. The choice of the firm silicon was crucial for the overall shell o-ring or it would not keep its shape to fit into the device. While we also used *Dragon Skin* for the button, using the dipping method would also work with softer silicone.

Commercially available micro-SD and USB-C compatible silicon plugs were purchased to waterproof the outlets. Because these components are flush with the outer region of the shell, they are pertinent to ensuring that the elements are protected when the elements are in the field and not in use. To add an extra layer of protection, we also applied epoxy to the inner surface of the clicker shell to prevent water seepage caused by the porosity of the 3D-printed filament.

Mechanical Testing and Validation

A series of tests were performed to verify that the mechanical shell met the structural and environmental performance requirements established for this device. All tests were conducted on the shell alone, without electronic components installed, to isolate mechanical behavior and avoid unintentional damage to active circuitry. These tests included (1) waterproof testing, (2) drop and durability evaluation, and (3) analytical validation of material performance relative to required operational loads.

Waterproof Testing of Shell

Waterproof testing was conducted to determine whether the enclosure could withstand conditions comparable to IPX6-level exposure, which corresponds to high-pressure water spray. While not performed as a full certification procedure, this test sought to demonstrate that the device could be subjected to rainfall, splashing, or incidental immersion in shallow water (e.g., a puddle) without significant fluid ingress.

Procedure:

1. Shell weighed prior to exposure
2. Shell subject to continuous high-pressure water jet for 1 minute
3. Shell mass was recorded again after wiping excess water from the exterior surface.

Table 9. Mass of shell before and after exposure to waterproof test.

Mass Before	Mass After
110.7g	111.0 g

The 0.3 g difference is consistent with the known hygroscopic nature of polylactic acid (PLA), the material used for the prototype. PLA absorbs small amounts of moisture into microvoids and the polymer matrix during exposure (Table 9). Because no liquid was observed inside the housing and mass gain remained within expected absorption limits, the result indicates minimal risk of water penetration during typical outdoor use. This validates that the shell meets the project requirement for environmental splash and incidental exposure resistance, though not full submersion.

Drop and Durability Testing

The drop test evaluated whether the enclosure could withstand typical mechanical shock encountered during handheld operation. The device mass was approximately 111 g, and drops were performed from a height of 3 ft (0.91 m) with zero initial velocity.

The shell was dropped multiple times on multiple different faces: (1) front-facing button regions, (2) back surface, (3) sidewalls, (4) vertices and edges.

After each impact, the shell was visually inspected for cracking, buckling, delamination, or surface fatigue. No damage was observed across all drops, indicating that the enclosure provides adequate structural integrity under realistic accidental handling conditions. This result aligns with theoretical expectations given PLA's tensile strength of 50–60 MPa, allowing it to withstand the transient stresses produced by handheld-height drops without reaching fracture thresholds.

Material Performance Validation

While experimental testing confirmed adequate real-world performance, additional engineering justification was conducted using the known material properties of PLA relative to expected loading scenarios (Table 10).

a. *Tensile and Yield Strength*

Table 10. Typical PLA properties for the prototype materials.

Property	Value
Tensile Strength	50–60 MPa
Tensile Modulus	3.0–3.5 GPa
Flexural Strength	80–100 MPa
Density	1.25 g/cm^3

Given the device mass of 111 g, the stresses imparted to the shell during normal use (button pressing, handling forces < 20–30 N, and minor torsion during grip) are orders of magnitude smaller than PLA's failure thresholds. Even the ~1000 N drop impact distributes across localized contact regions, producing momentary stresses that remain below yield for the wall thicknesses used (typically ≥2 mm in your design).

b. Heat Resistance

PLA exhibits a heat deflection temperature (HDT) of ~55–60°C, meaning deformation could occur at elevated temperatures, such as inside a car on a very hot day. Given the intended handheld and outdoor use profile, short-term exposure to ambient temperatures (0–40°C) poses no risk of deformation. However, sustained temperatures above ~55°C may soften the shell. This limitation should be noted for future material upgrades (e.g., PETG, ABS, or nylon).

c. Vibration Resistance

Given PLA's modulus (~3 GPa) and the compact geometry of the housing where the shell behaves as a stiff body, the natural frequencies falling well above the range induced by handheld operation or walking, and typical vibration amplitudes less than 1 mm impart negligible cyclic stress. Thus, no resonant or fatigue-induced failures were observed during handling or repeated assembly cycles, suggesting that vibration does not pose a structural risk under expected use conditions.

Final Shell Design

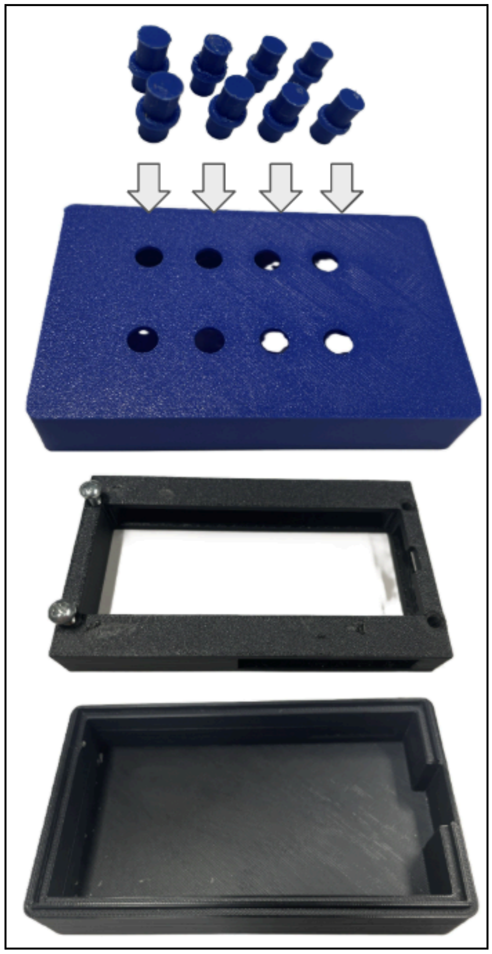


Figure 15. The full schematic of fabricated base, grid, lid, and button caps.

The final prototype is organized top-down around a mechanical stack that prioritizes reliable user interaction, robust assembly, and flexible electrical integration (Fig. 15). At the user interface, top-mounted buttons are actuated through direct cylindrical touch extensions, selected to ensure consistent actuation regardless of wiring variability and to preserve the tactile precision needed for repeated use—reflecting the emphasis placed on intuitive, repeatable button access throughout ergonomic evaluation. These interfaces are integrated into a snap-fit lid that forms the primary enclosure boundary; the snap-fit was chosen over screw-based methods to reduce tolerance stacking, simplify assembly, and support rapid iteration while maintaining consistent wall geometry around the shell. Beneath the lid, an open internal frame and wire-routing region were incorporated to accommodate varied wiring paths and evolving component layouts, acknowledging the need for reconfigurability identified in earlier prototypes. This region provides controlled spatial organization without over-constraining the electrical architecture. At the base of the assembly, the rectangular footprint is dimensioned directly around the breadboard to maximize internal packing efficiency, and the board is rigidly mounted using a screw-in attachment strategy that provides the stability required for dependable electrical performance. Together, these elements reflect a design process that balanced user-interface clarity, manufacturability, and internal adaptability, resulting in a prototype optimized for functional testing and iterative refinement.

D. Electrical

Design Considerations

The operational requirements of the KoboClicker require a system that is reliable and straightforward while having a long run time and robust data integrity. To meet these needs, the electrical design specifications were organized into three primary categories: Power (Table 11), Storage (Table 12), and Haptics/Interface (Table 13). Within these categories, some specifications were identified as non-negotiable, as they contribute to the core functionality of the device. These were bolded in the tables below. Other specifications were deemed more flexible, allowing room for optimization, cost trade-offs, or design iteration. These were left unbolded.

Final Technical Specifications

Table 11. Power source specifications.

Function	Requirement
Run time	**8+ hours**
Nominal voltage	**3.3V**
Maximum current	100 mA
Maximum size	**26 mm diameter x 65 mm height**
Charging method	**USB-C**
Form factor	Cylindrical
Cycle life	1000+ cycles
Temperature range	-20°C to 50°C
Maximum cost	$10 USD

One of the non-negotiable power requirements for the KoboClicker is its run time. We set an 8-hour minimum so the device could last a full work day without needing a recharge. To support this specification, the 3.3 V operating level and the 100 mA maximum current draw required that the electronics chosen were efficient, safe, and low-power. Another non-negotiable specification was a USB-C charging port so that the KoboClicker would be compatible with most existing charging systems. We also limited the physical size of our electrical components to 26 mm in diameter and 65 mm in height to keep the KoboClicker compact, easy to hold, and discreet to use.

Table 12. Storage and other hardware specifications.

Function	Requirement
Memory type	Flash

External flash capacity	**20MB**
Maximum events stored	**100,000 Events**
Flash endurance	≥100k cycles
Maximum upload time	**3 minutes**
Sustained BLE transfer rate	90 KB/s
Maximum bytes per event	100 bytes/event
SRAM	150 KB
Connection protocol	Bluetooth 5.0 (GATT)
BLE connection range	2 Meters
Memory reset	Available via reset button
Anti-corruption mechanism	Cyclic redundancy check *CRC-8 requires 1 byte
Data file format	Binary log or CSV
Cost	$15 USD

For storage, we chose to use flash memory because it is non-volatile, which is critical for protecting the data when the device is turned off or if the battery dies in the field. Through our research and conversations with HHI, we decided that we needed the ability to store at least 100,000 data events at a time. To put that in perspective, in a rapid needs assessment, the device would be able to record information about 100,000 people before streaming the data over BLE or writing to the microSD card, after which the memory would be wiped. We decided to limit the amount of memory needed to store each event to 100 bytes, with 50 bytes being a practical target. This event size provides enough room for key details such as event type, operator ID, timestamps, and error-checking, without needlessly using memory. Based on this specification, we determined that the KoboClicker would need 20 MB of flash memory. To outline the rough calculation: storing 100,000 events at about 100 bytes each would require 10 MB. Accounting for wear leveling and file-system overhead with a safe margin brings us up to 20 MB. Given that data would be streamed often between the KoboClicker and the host device, we wanted to achieve a maximum BLE upload time of 3 minutes for 20 MB, requiring the wireless connection to maintain about 90 kbps.

Table 13: Haptics and Interface Specifications

Function	Requirement
Maximum button contact resistance	**100 mΩ**
Minimum button insulation resistance	**100 MΩ**
Button debounce time	**5-10ms**
Button actuation force	200g
Haptic motor “save” indicator	Motor driver connected to I2C bus
LED “save” indicator	LED driver connected to I2C bus
LED “low power” indicator	Below 20%
Cost	$10 USD
Haptic motor min rev/minute	2,000
Maximum size	7.8 mm diameter

The KoboClicker is built to be able to collect a variety of data points extremely quickly, meaning that the buttons and feedback system have to be extremely reliable. First, we set strict electrical requirements for the buttons. It had to have a low contact resistance and very high insulation resistance, together with five to ten millisecond debounce time and a 200-gram actuation force. These specifications ensure that every press is intentional and correctly detected.

Design Decisions

The next section will outline several major design decisions that the electrical team made based on the design specifications described above.

Microcontroller Selection: nRF52840 Xiao Sense

Our first major design decision was to choose a microcontroller for the KoboClicker. The microcontroller would determine the efficiency and range of our Bluetooth and the charging system and size footprint of our device. As we later found, it would also dictate which software libraries we had access to in Arduino IDE. Our design specifications placed an emphasis on reliable and fast Bluetooth, a small footprint, and a universal charging system such as USB-C. While conducting research, we looked for microcontrollers that met these three major requirements. Our research yielded a shortlist of three microcontrollers: 1) the nRF52840 Xiao Sense from Adafruit, 2) the ESP32-C3 Xiao from Espressif, and 3) the Arduino Nano 33 BLE from Arduino. We determined that the 21 mm x 17.5 mm Xiao boards were more favorable than the 45 mm x 18 mm Arduino Nano 33 BLE for a small, handheld device.

Through further research, we determined that the nRF52840 board had a superior Bluetooth range on its on-board BLE module compared to both other boards and was compatible with existing Bluetooth scanning applications like nRFConnect. nRFConnect gave us a way to validate our circuits while the Software team worked on developing an app to receive Bluetooth data. For these reasons, we decided to move forward with the nRF52840 Xiao Sense (Fig. 16).

We were aware of one major drawback of the microcontroller we chose: it gave us access to a maximum of 11 GPIO pins. From a rough understanding of our circuit, we needed four pins for an SPI protocol, two pins for any I2C peripherals, and potentially two more pins for a linear voltage regulator. This would leave us only three GPIO pins for data buttons. Assuming our circuit required one pin per button, this would constrain our device to a maximum of three buttons. This pin limitation inspired research into circuits that would minimize the number of pins needed for buttons and our charging IC, leading to the analog button ladder and the QT Py BFF.

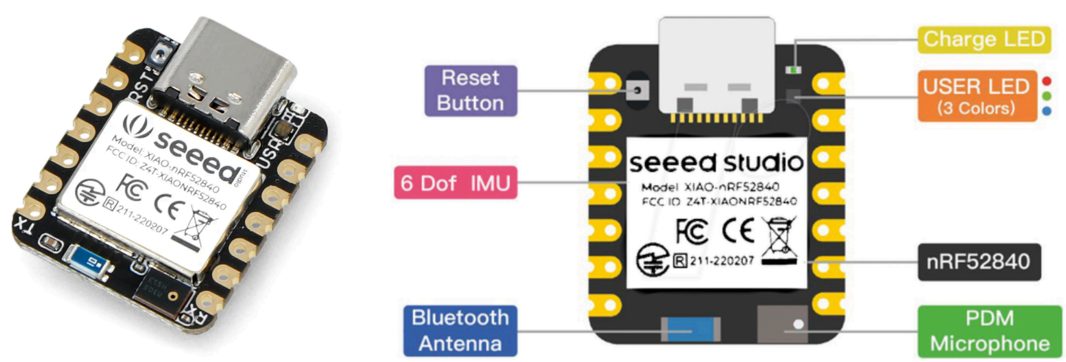


Figure 16: physical board layout and USB-C form factor (left). Labeled diagram identifying onboard components, including the nRF52840 SoC, Bluetooth antenna, 6-DoF IMU, PDM microphone, user and charge LEDs, and reset button (right).

Analog Button Ladder

As mentioned above, the limited pins on our microcontroller led us to search for alternative circuits that would allow us to detect multiple buttons with the same pin. Instead of using digital pins that would read high or low depending on button presses, we decided to experiment with an analog button ladder (Fig. 17). The circuit uses voltage dividers to create several loops where current could flow, each with a different amount of resistance. Each loop has a button corresponding to it, and all of these loops are open circuits (no current) when the button is not pressed. Pressing the button closes the circuit, and the 5V input voltage drops a certain amount, which corresponds to the unique resistance of the loop. Analog pins can read these voltages, and our Arduino code can determine which button was pressed based on the voltage that was read. For example, a button further along the ladder would return a smaller voltage when pressed, because the initial 5V input from the microcontroller would drop across more resistors.

We were able to validate this circuit on a breadboard and detect button presses from eight different buttons using only two GPIO pins.

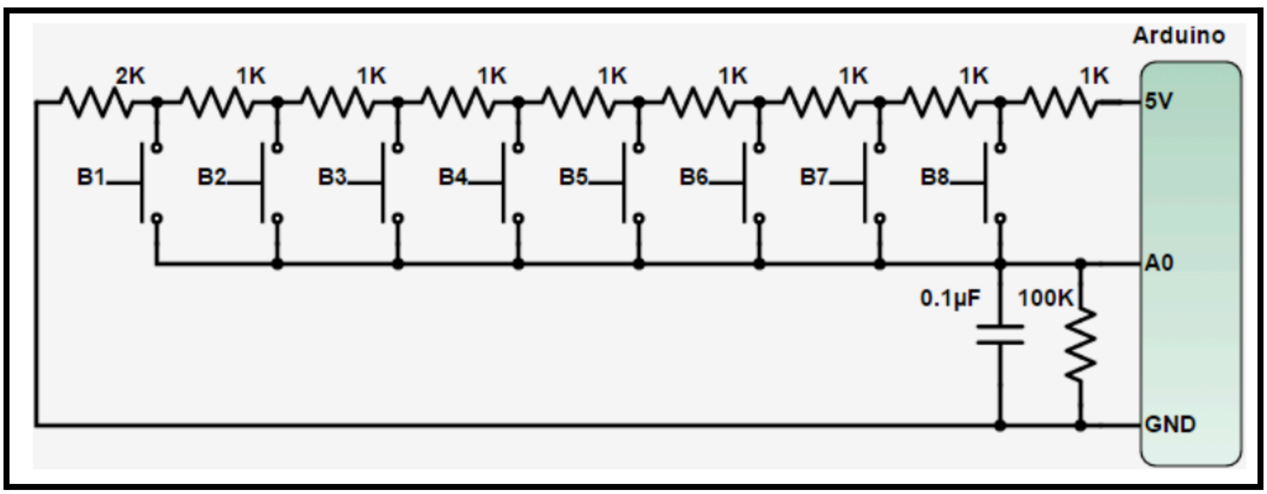


Figure 17: An schematic of an analog button ladder which we based our circuit on. Since the ladder's resistors are all 1kΩ and there are eight buttons, the voltage drops 0.5V for each consecutive button.

Charging IC: QT Py BFF

A reliable power subsystem was essential to the electrical design of the KoboClicker, as the microcontroller and all peripheral components required a stable 3.3 V rail for correct operation. The system also needed to support a rechargeable Li-ion pouch battery to enable portable field use. Early in prototyping, the electrical team explored several discrete approaches to voltage regulation and battery management, including low-dropout (LDO) regulators and compact buck converters. While theoretically viable, these components proved impractical for our timeline and fabrication constraints. The buck converters we sourced were too small for effective use on a solderable breadboard, and both LDO- and buck-based solutions would have required additional wiring, external protection circuitry, and careful layout to ensure stable operation. Each option introduced unnecessary design complexity and consumed valuable physical space inside the compact enclosure.

After consultation with SEAS instructional staff, particularly Leo Gomez, the team adopted the QT Py BFF as the integrated charging and regulation solution (Fig. 18). Produced by the same manufacturer as our microcontroller (the Seeed Studio nRF52840 Xiao Sense), the BFF is designed as a “backpack” module that matches the Xiao’s footprint and pin layout exactly. This allowed the board to be soldered directly beneath the microcontroller with minimal effort, effectively merging the two into a single compact assembly. The BFF provides regulated 3.3 V output, a JST-PH connector for the Li-ion pouch battery, and built-in charging and protection circuitry, all without the need for external components.

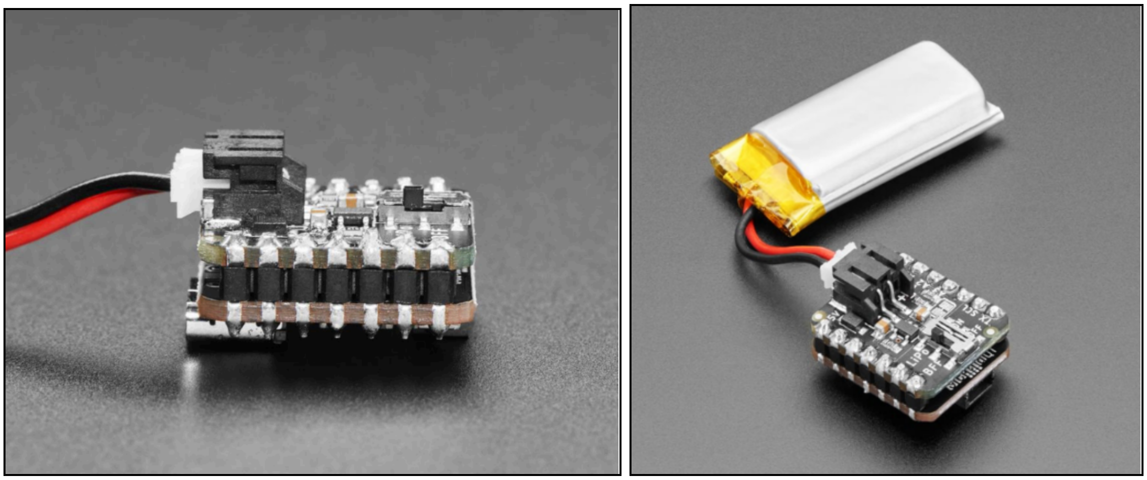


Figure 18. QT Py BFF module with Li-ion pouch battery (left) and side view of the board and connector (right).

In practice, the QT Py BFF offered a highly reliable power platform. Current draw measurements during BLE transmission remained stable, and the system operated without voltage dips during SD-card writes or button-matrix activity. While the team did not characterize brownout behavior or test charging while active, the BFF's ease of integration, low footprint, and strong compatibility with the Xiao made it the optimal choice for the prototype.

Chronology of Electrical Prototyping

The electrical team's prototyping process proceeded through four major phases, beginning with foundational Bluetooth validation and culminating in the full integration of the button interface, power system, and data-logging subsystem. This chronology outlines the iterative development we went through, the challenges we encountered, and the progression toward the final working prototype.

Early BLE Prototyping and Initial Button Interface

Prototyping began with a proof-of-concept test using an Arduino Uno, a single button, and the nRF Connect mobile application. The goal in this initial phase was to confirm that

Bluetooth Low Energy (BLE) transmission from a microcontroller was feasible for rapid button-press detection, and to understand BLE latency, message formatting, and device discoverability. Early tests surfaced several usability challenges: inconsistent BLE advertisement, cryptic hexadecimal strings representing button states, and faulty logic where the device reported “pressed” even when unpressed. These issues required us to refine the Arduino code to stabilize BLE output and ensure accurate button detection.

As the design evolved, we moved on to creating prototypes of the analog button ladder on the Arduino Uno, moving through two to three iterations with varying resistor and capacitor values. These revisions sought to maximize voltage separation between button states while avoiding overlap. We verified each configuration using the Arduino IDE serial monitor, ensuring that each button produced a distinct, repeatable analog reading.

Migration to the nRF52840 Xiao Sense and Expanded Button Ladder

Upon arrival of the Xiao microcontroller, the team began migrating both BLE functionality and the ladder interface from the Uno. This transition proved significantly more challenging: BLE libraries for the Nordic board differed substantially from those used on Arduino boards, and an attempt to upload new bootloader firmware to solve these incompatibilities resulted in the first board being rendered unusable. With a replacement microcontroller, the team successfully implemented BLE advertising and button-press reporting. The resistor ladder behaved as expected on the Xiao, permitting expansion to the full eight-button interface using two parallel four-button ladders routed to two analog pins. This expansion required revisiting the physical arrangement of components in anticipation of later integration on a solderable breadboard.

Power-System Development with QT Py BFF

In parallel with button-interface development, the team evaluated options for voltage regulation and battery charging. Preliminary consideration of discrete LDOs and buck converters revealed practical limitations: footprints were too small for rapid prototyping, and custom power-management circuitry risked introducing instability. Consultation with SEAS staff led to the adoption of the QT Py BFF, a compact “backpack” module designed to mount directly beneath the Xiao. After soldering the BFF to the Xiao, the system operated reliably without requiring additional regulation stages. The team verified stable current draw during BLE transmission but did not extensively characterize battery-life performance or brownout behavior.

microSD Prototyping Using PMOD Module

The data-logging subsystem was developed using a PMOD microSD module connected over SPI. Prototyping began by manually wiring the module to the Xiao using header jumpers. Testing focused on file creation, write reliability, and persistence across power cycles. Early issues arose from SPI pin misalignment and occasional wiring errors, but once resolved, write operations were consistent and did not produce corrupted files. The team did not directly test simultaneous BLE transmission and SD writes, which remains an area for future verification.

Subsystem Integration on Solderable Breadboards

With all major subsystems functioning independently, the team proceeded to integrate them onto a solderable breadboard. Across approximately four board iterations, the electrical team refined the layout of the eight-button ladders, resistors, power connections, and microcontroller assembly. Some boards exhibited unstable or “wonky” voltage readings due to soldering imperfections, necessitating repeated re-soldering and reconstruction. Physical integration introduced additional complications: the wires connecting the PMOD microSD module were thin and fragile, frequently breaking when bent to fit within the mechanical enclosure. These mechanical stresses were a recurring issue and required multiple rounds of reinforcement and re-soldering. The software team concurrently implemented software-level debouncing and adjusted button-detection thresholds to accommodate minor variances in analog voltage readings.

By the end of the integration phase, the electrical subsystem achieved full functionality. The final prototype supported all eight buttons through dual resistor ladders, reliably logged data to the microSD card, transmitted button events via BLE, and operated from the Li-ion battery through the QT Py BFF power module. Some advanced features, such as the DRV2605L haptic driver and LRA motor, were not integrated before the deadline but remain compatible with the current electrical architecture.

Final Circuit

All components in the circuit operate together through the QT Py, which distributes regulated 3.3 V power and coordinates communication across SPI, I^2C, and analog inputs. The SD-card module exchanges data with the microcontroller over SPI, while the haptic driver receives commands via I^2C and delivers motor actuation without loading the logic rail (Fig. 19a, 19b). The resistor-ladder input network feeds uniquely encoded voltages into a single ADC pin, allowing the QT Py to interpret button presses with minimal wiring. Because every subsystem shares a common power domain and stable ground reference, the microcontroller can reliably manage storage, haptic feedback, and user input simultaneously, with each peripheral contributing to an integrated and electrically coherent system.

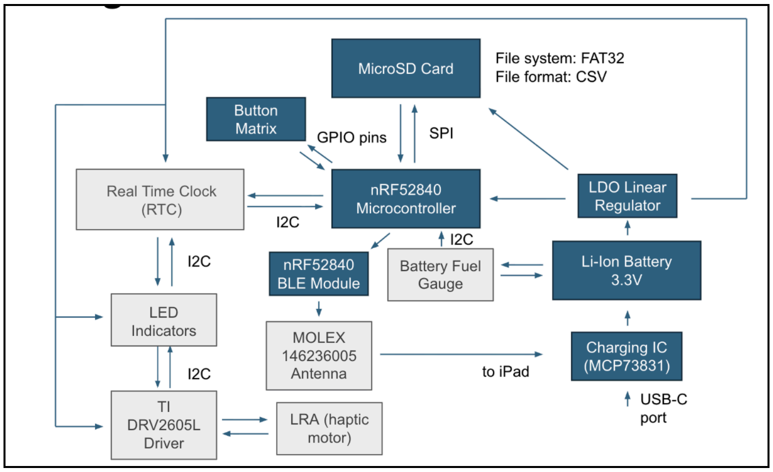


Figure 19a. Block diagram of the circuit components that were integrated (blue) and recommendations for further integration (grey).

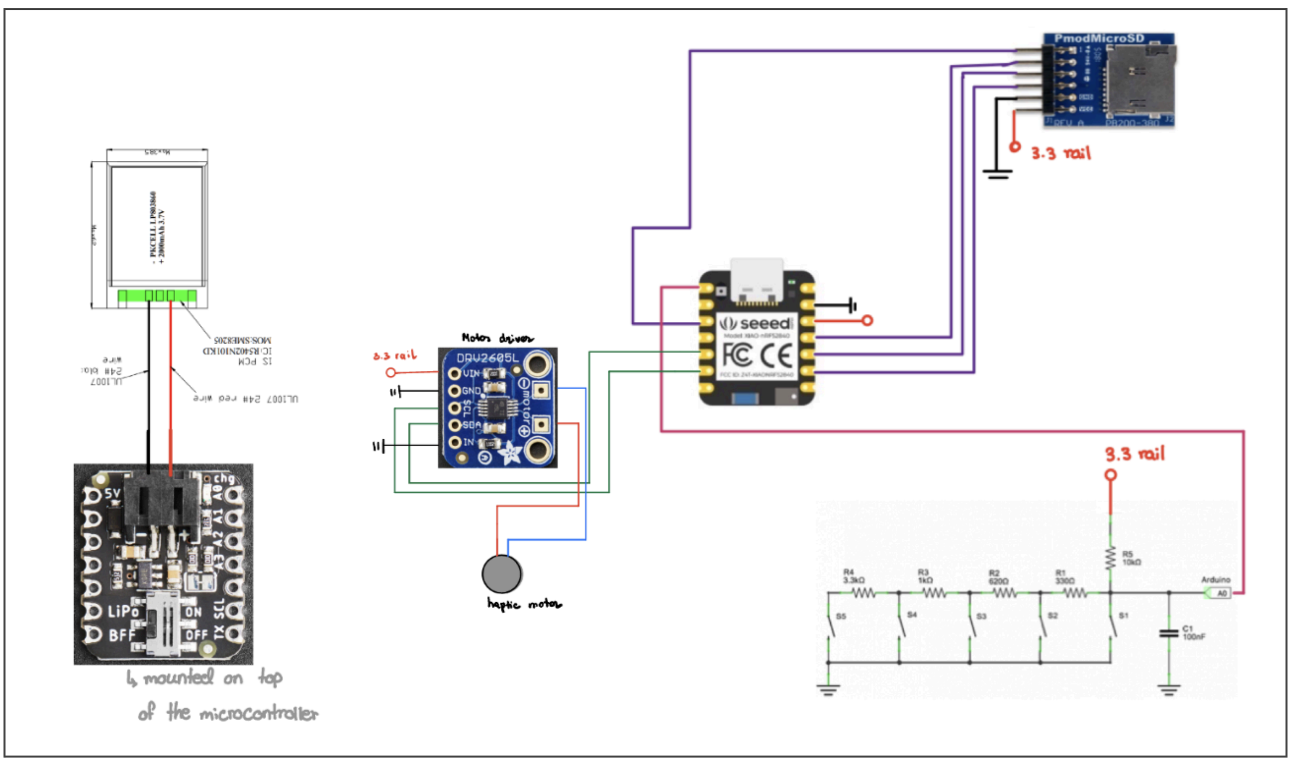


Figure 19b: Full circuit schematic.

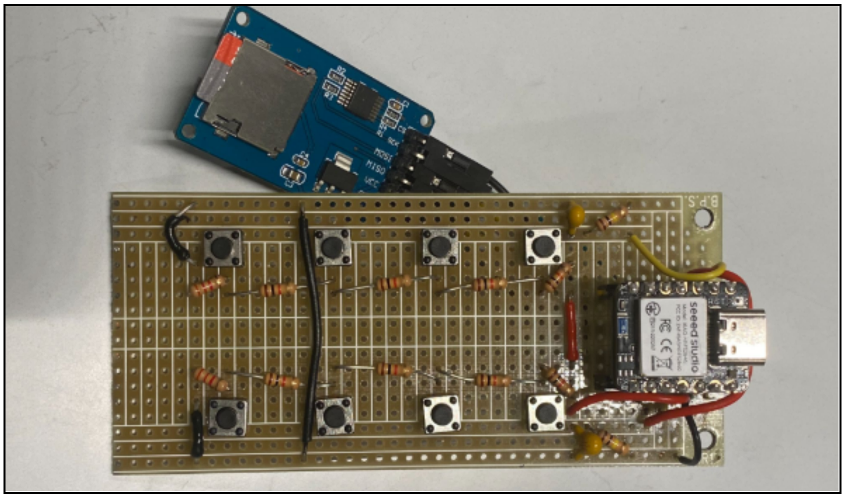


Figure 20. Assembled prototype showing the QT Py microcontroller mounted to the perfboard, the resistor-ladder button array, and the external microSD module connected for data logging.

The completed prototype in Figure 20 brings all system elements onto a single perfboard, illustrating the final electrical footprint of the device. The QT Py microcontroller is mounted directly on the board alongside the resistor-ladder button array and associated wiring, showing how the inputs, power rails, and signal routes physically converge in the final design. The microSD breakout, visible behind the main assembly, interfaces through header pins and demonstrates how external storage integrates cleanly with the core circuit. As a whole, the assembled hardware provides a clear view of how the subsystems fit together spatially and electrically in the finished implementation.

E. Software

Design Considerations

The software component of our system is designed to facilitate data collection from the microcontroller-based clicker device and ensure seamless integration with KoboToolbox. We approached this through two parallel arms (Fig. 21). The first arm involves a custom Android application that acts as a proof-of-concept interface for collecting button press events and updating a dashboard in real time. Initially, we considered using KoboCollect, the official mobile data collection app for KoboToolbox. However, since we found difficulty with accessing KoboCollect's source code, we developed our own app to demonstrate how the clicker device could interact with a mobile interface while maintaining flexibility for integration with KoboToolbox in the future.

The second arm leverages the KoboToolbox API to allow users to upload CSV data directly from a micro SD card. This provides an alternative for cases where Bluetooth connectivity is unavailable or impractical. As illustrated in the flowchart, the microcontroller records button presses to a CSV file, which can then be parsed either by the Android app for live updates or by the API upload tool to convert the CSV data into XML and submit it directly to KoboToolbox. This dual-path approach ensures both real-time and offline data collection options, making the system versatile and robust.

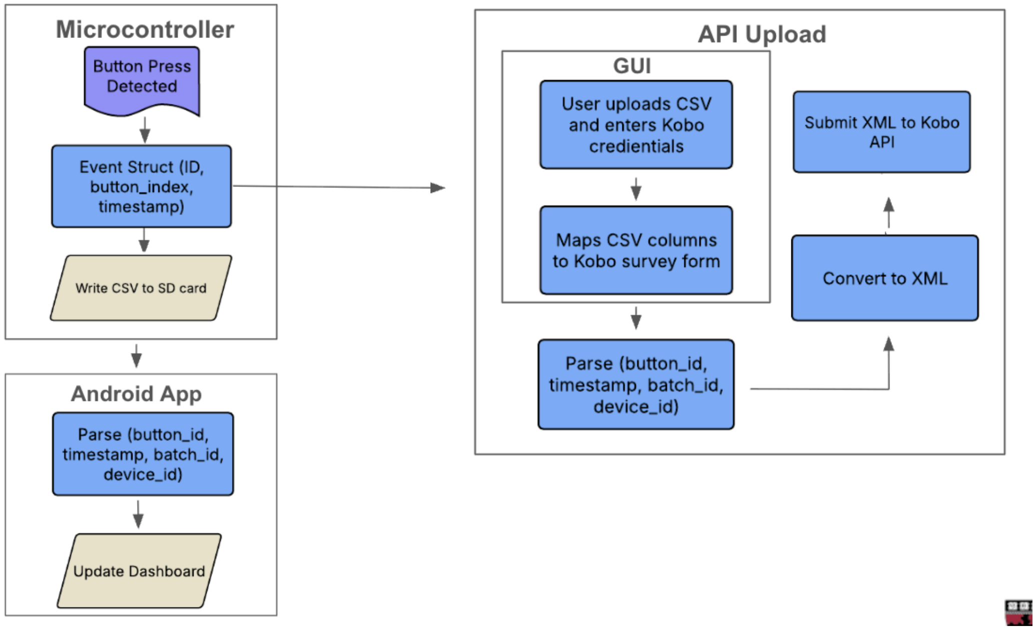


Figure 21. Overview of the software data flow. The microcontroller records button press events to a CSV file, which can either be parsed by the Android app for real-time dashboard updates or processed through the API upload tool to convert the CSV to XML and submit it to KoboToolbox.

Final Technical Specifications

The software subsystem was designed to reliably capture button-press events, store them in non-volatile memory, and transmit them via Bluetooth to a mobile gateway for upload to KoboToolbox or a spreadsheet backend. The specifications emphasize data integrity, fault-tolerant transmission, and low-overhead logging, ensuring that no events are lost even in the presence of intermittent connectivity or power resets. The following subsections detail the required system behaviors and constraints. Within these categories, some specifications were identified as non-negotiable, as they contribute to the core functionality of the device. These were bolded in the tables below.

Event Capture and Logging

The event capture pipeline must ensure that every interaction is recorded with precise timing and unique identification, without duplication or omission (Table 14). Events are written immediately to non-volatile memory to preserve data even in the event of unexpected shutdown or BLE disconnection.

Table 14. Event capture specifications.

Function	Requirement
Unique event tracking	**Each button press must generate a unique event ID**
Metadata	Software must record button ID, UTC timestamp, and event payload
Timestamp accuracy	±50 ms precision
Immediate persistence	Events must be queued in non-volatile memory upon capture
Concurrency handling	Must avoid duplicate or lost records during simultaneous presses

To guarantee reliability, the software logs each button press as an atomic event containing a unique identifier, a high-accuracy timestamp, and button metadata. Events are immediately written to non-volatile storage to prevent data loss, ensuring that the system can withstand resets or mid-use power interruption. The firmware also includes safeguards to eliminate duplicates in cases of rapid or overlapping user interactions

Data Transmission & Sync Protocol

Data transmission is performed via Bluetooth Low Energy and a companion app that relays submissions to KoboToolbox. The upload protocol emphasizes robustness in unstable field conditions, including acknowledgment checking, retry logic, and resumable transfers.

Table 15. Data transmission specifications narrative description.

Function	Requirement
Sync latency	Upload must begin and complete within 5 seconds of sync initiation
Delivery guarantee	Software must verify server acknowledgment for each batch

Retry mechanism	**Failed transmissions must auto-retry until successful**
Duplicate prevention	Upload logic must not resend delivered events
Memory cleanup	Events are cleared only after confirmed delivery
Transport layer	Must support Bluetooth (BLE) transmission
Batching	Events must be batched by count or time interval (configurable)
Resumability	Uploads must resume partially delivered batches

As described in Table 15 above, the software implements a resilient synchronization protocol designed for intermittent connectivity. All transmissions require confirmation before local deletion, preventing silent data loss. Failed attempts trigger automatic retry logic, and partial uploads are resumed without duplication. Using TLS ensures confidentiality of all transmitted event logs, while BLE keeps power usage minimal.

Data Storage & Integrity Management

Table 16 lists the data storage specifications. Local storage is optimized for high-volume event buffering and corruption protection. A structured schema and integrity checks ensure that all stored events can be safely reconstructed and mapped to downstream systems.

Table 16. Data storage specifications.

Function	Requirement
Storage capacity	Must store up to 10,000 unsent events without loss
Data schema	Must use a structured XML schema compatible with KoboToolbox
Batch metadata	Each batch assigned a unique batch ID
Persistence	Must retain all unsent data through power loss and reset

The storage system is intentionally conservative, retaining all unsent events across crashes or restarts. XML formatting ensures compatibility with form field mappings and downstream ingestion tools.

Security & Authentication Requirements

Security controls ensure only authenticated clients can upload data and that credentials remain protected inside the device (Table 17).

Table 17. Security specifications.

Function	Requirement
Authentication	**Must authenticate to KoboToolbox via API tokens**
App protocol	**Must require a unique token for data reception**
Error tracking	**Log authentication failures for debugging**

API credentials are encrypted at rest to avoid extraction from memory and authentication is required for every submission cycle. These safeguards ensure the device cannot be used maliciously.

Time Management Requirements

Table 18 below lists timekeeping specifications for the software component. Accurate and consistent timing is critical for event ordering and analysis. The clock must maintain precision and be able to resynchronize when a gateway device becomes available.

Table 18. Timekeeping specifications.

Function	Requirement
Drift tolerance	System clock must stay within ±1 second / 24 hours
Time sync	Must auto-synchronize via BLE gateway when possible
Timestamp format	Must record events in ISO-8601 UTC

Accurate timing supports downstream analytics and cross-device synchronization. The firmware maintains low drift and standardizes timestamps for compatibility with spreadsheets, APIs, and form systems.

Output Formats & External Integration

To ensure compatibility with downstream analysis tools, the software exports all captured events in standardized CSV and XML formats. Each batch of uploaded data is packaged with consistent metadata fields and timestamp formatting, enabling seamless mapping to KoboToolbox forms or spreadsheet-based workflows (see Table 19).

Table 19. Output and integration specifications.

Function	Requirement
Export format	**Must export events to CSV with fixed headers**
Required fields	device_id, button_id, timestamp, press_type, batch_id, status
File generation	One CSV file per sync batch
KoboToolbox integration	Must generate KoboToolbox-compliant XML submission payloads
Form mapping	Must map each button to KoboToolbox form fields
Profile management	Supports one active KoboToolbox profile at a time

These structured outputs allow external systems to confidently parse, store, and analyze the event data without additional transformation. By enforcing strict schema compliance and predictable file generation, the software guarantees interoperability across both field and analytical environments.

Firmware Programming

To support reliable offline data collection, we developed firmware for the nRF52840 Xiao Sense that performs three main tasks: reading multi button input through two resistive ladders, constructing structured data rows that represent a participant's responses, and transmitting and logging those rows through both Bluetooth and SD card storage. The complete firmware can be found on GitHub at: https://github.com/jcurcio3105/KoboClicker.

At a high level, the microcontroller functions as a self contained data logger. Each button press contributes to one row of binary data, where each column represents one characteristic. Pressing the NEXT button (Button 1) starts a new row and assigns it a timestamp. Buttons 2 through 6 fill in that row by setting specific columns to 1, and each column can be set only once per row. When data entry for that row is complete, pressing NEXT again saves the current row and immediately starts a new one. This workflow mirrors how a field user collects a small number of attributes per individual before moving to the next person.

The firmware also includes an ERROR button (Button 7), which resets the active row to all zeros. This allows the user to discard and replace a row if the wrong buttons were pressed. If ERROR is pressed during an active row, the firmware saves a zeroed row and starts a new one automatically. If ERROR is pressed when no row is active, it simply starts a new zeroed row.

Power State and File Management

The POWER button (Button 8) supports two behaviors. A short press toggles the system between enabled and disabled modes. When disabled, all buttons except POWER are ignored and Bluetooth advertising is paused. This reduces power usage and prevents accidental input. A long press of POWER, defined as holding the button for at least two seconds, rotates the system to a new CSV file on the SD card. The first log file is DATALOG.CSV, and later files follow a sequence such as DLG0001.CSV, DLG0002.CSV, and so on. A new file is created with a fresh header, and the in memory buffer is cleared so that each logging session starts cleanly. Older files are never deleted, which allows long term field deployments without data loss.

Resistive Ladder Button Decoding

The device reads eight buttons using two analog inputs. Each input is a resistive ladder that produces a unique voltage for each button. The firmware interprets these voltages and maps them to button numbers. The code also includes debounce logic to avoid noise and accidental multiple triggers. Using resistive ladders reduces wiring complexity while still allowing the system to distinguish eight separate inputs.

Row Construction and Internal Buffering

Each row is stored in memory with five binary columns, a timestamp, and two status values that track whether the row has been successfully sent to Bluetooth and to the SD card. Rows are stored in a circular buffer, which prevents overflow and allows continuous logging even during high activity. Rows remain in memory until both delivery paths have confirmed successful receipt.

Dual Logging Path: Verified Bluetooth and SD Storage

The firmware implements a dual delivery system to ensure data integrity. Every row must be delivered successfully over Bluetooth and to the SD card before it is removed from the buffer.

Bluetooth Transmission

When a phone or tablet connects using Bluetooth, the firmware sends all unsent rows as CSV strings through a UART style characteristic. A row is marked as delivered only when the firmware confirms that it has been transmitted successfully.

SD Card Logging with Verification

The firmware writes data to the SD card in small batches. After writing, it reads back the entire file and checks that all newly added rows appear exactly as expected. This verifies that the SD write succeeded before marking rows as delivered. If a row is not found during verification, it stays in the buffer and will be retried. By requiring success on both the Bluetooth path and the SD card path, the system provides very strong protection against data loss even if Bluetooth connections drop or the SD card behaves inconsistently.

Security Token and Data Integrity

Each CSV row includes a constant token value at the end. The receiving application checks this value before accepting a row. This prevents partial transmissions, corrupted entries, or mixed logs from being interpreted as valid data.

Summary of System Behavior

The firmware ensures that:

1. Each participant is represented by one structured and timestamped CSV row.
2. Rows are created, updated, saved, or discarded through physical buttons only, so the workflow is simple in the field.
3. The power system supports short and long press actions for pausing the device and creating new log files.
4. All data is logged both over Bluetooth and to the SD card, and each delivery is verified.
5. Rows are deleted from memory only after both Bluetooth and SD card have confirmed successful delivery.
6. Every row is formatted in a consistent CSV structure for compatibility with KoboToolbox and other data systems.

The complete source code is available in the GitHub for review and reproducibility.

Android Application

When designing the Android application, we began by creating a simulation of how the final version of the app would interact with the KoboClicker. Because access to a dedicated Android device and finalized hardware was limited early in development, the initial focus was placed on building a software-side simulation that accurately represented the expected behavior of the completed system. This approach allowed development and testing to proceed in parallel with hardware planning, while ensuring the app's structure remained compatible with real Bluetooth data once a physical device became available.

The goal of this stage was to design and implement an Android application that simulates receiving structured data from a microcontroller over Bluetooth, logs it in tabular format, and exports it as a CSV file for later analysis or upload to KoboToolbox. The app was designed such that each individual is represented by a single row of binary data, with each column corresponding to a specific characteristic. Button presses on the microcontroller conceptually toggle values within that row from 0 to 1. Once data entry for an individual is complete, a dedicated "next" button transmits the completed row to the Android app and initializes a new row on the microcontroller. The Android application presented here was designed to closely mirror the real-world workflow, even though it currently runs entirely in an Android emulator, using simulated data rather than live Bluetooth input.

The application was developed using Android Studio and Jetpack Compose and is designed to run either on an Android emulator or a physical Android device. In its current form, the app is typically run in an emulator, which enables rapid testing, debugging, and iteration without dedicated hardware. The app uses modern Android features such as edge-to-edge rendering and reactive UI state management, ensuring the interface updates automatically in response to new data without requiring manual redraws or refresh logic. This design choice is especially important for applications that asynchronously receive streaming or batch-based data, such as Bluetooth transmissions from an external device.

At the core of the application is a simple but powerful data model in which each individual is represented by a single data object. Each data entry consists of a timestamp and a list of integers representing binary characteristics. Conceptually, this structure maps directly to a CSV row, where the timestamp occupies the first column, and each subsequent column contains a 0 or 1 indicating the presence or absence of a specific characteristic. By modeling the internal data structure to match the final CSV output format, the application avoids unnecessary transformation steps and ensures consistency between what the user sees on screen and what is exported. The use of integer values rather than booleans further simplifies aggregation and export, as summing values across rows directly yields meaningful totals for each category.

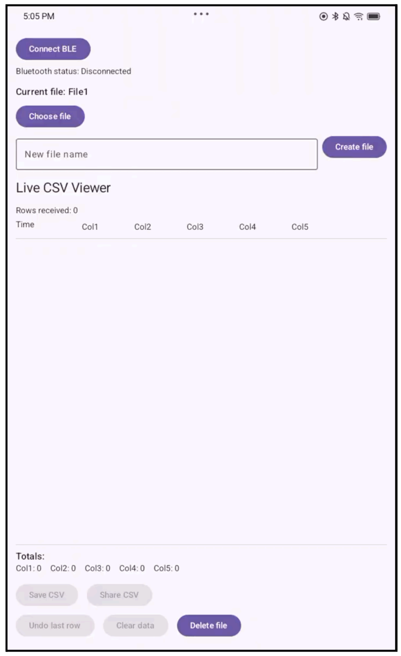
In place of a real Bluetooth connection, the app includes a simulation loop that mimics how data would be received from a microcontroller. At fixed time intervals, the app generates a new batch of data consisting of a timestamp and a randomly generated list of binary values, with one value per column. This simulated batch is then appended to the internal list of entries, exactly as real data would be upon receipt over Bluetooth. This simulation closely reflects the expected behavior of the real system. In the intended hardware implementation, the microcontroller would collect button presses for a single individual, construct a one-dimensional array of binary values, and transmit that array once the “next” button is pressed. The Android app would then parse that transmission and append it as a new row. Because the app already treats data as arriving in discrete batches rather than as individual button events, replacing the simulation with actual Bluetooth input would require minimal changes to the overall architecture.

The user interface displays incoming data in a scrollable, table-like layout (see Figure 22). Each row corresponds to one batch, representing one individual, and each column corresponds to a characteristic. Horizontal scrolling is used to accommodate multiple columns without compressing the layout, preserving readability even as additional categories are added.

Each row is rendered as a rounded, card-like element, visually separating individual entries and making the dataset easier to scan. The column headers are editable directly within the interface, allowing users to define or rename categories at runtime. This flexibility is particularly useful in field data collection scenarios, where the set of characteristics tracked may vary by study or context. The app also supports dynamically adding and removing columns, up to a defined maximum, while maintaining internal consistency across all existing rows.

In addition to displaying raw data, the application computes running totals for each column in real time. These totals are derived by summing the values in each column across all batches. Because the data is stored as integers, this aggregation is straightforward and efficient. The totals update automatically whenever new data is added or existing data is removed, providing immediate feedback to the user about overall trends in the dataset. This mirrors common post-processing steps that would otherwise be performed in spreadsheet software and demonstrates how basic analysis can be integrated directly into the data collection interface.

A key feature of the application is its ability to export collected data as a CSV file. Internally, the app converts the list of entries into a CSV-formatted string by first generating a header row from the column labels and then appending one row per batch. This CSV data can either be saved directly to the device's storage or shared using Android's built-in sharing mechanisms. Saving the CSV to device storage simulates writing data to an SD card in an embedded system, providing persistent local storage that can be retrieved later via a file manager or USB connection. Sharing the CSV allows the data to be transmitted immediately to other applications or services, such as email or cloud storage. Both export methods operate on the same underlying CSV generation logic, ensuring consistency regardless of how the data is ultimately used. Overall, our simulation-based development approach enabled meaningful



progress without hardware dependencies while ensuring that the final system can be extended to support real microcontroller input with minimal architectural changes.

Figure 22. Image of the Android application interface running on an Android tablet, illustrating the Bluetooth-enabled CSV logging system. This view represents the finalized application used to receive, verify, store, and export data transmitted from the microcontroller via Bluetooth Low Energy.

After validating the data model, user interface, and CSV export workflow using simulated Bluetooth data, the project transitioned from a purely simulated environment to a fully functional Android application capable of communicating with a physical microcontroller over Bluetooth Low Energy (BLE). This transition involved extending the original simulation-based architecture rather than replacing it, ensuring that the app's internal logic for batch handling, tabular display, and CSV export remained consistent. Development and testing were initially performed using the Android emulator, after which the finalized application was deployed onto an Android tablet to enable real Bluetooth communication with the microcontroller hardware.

The core behavior of the application remained the same as in the simulated version: each individual is represented by a single row of binary data, button presses on the microcontroller toggle values within that row, and a "next" action finalizes the row and transmits it to the app. However, several new features were added to support reliability, security, and real-world usage constraints. One major enhancement was the introduction of a security mechanism requiring a token for updating data via BLE. This prevents unauthorized devices from injecting or modifying data and ensures that only the intended microcontroller can communicate with the application. In addition, the app enforces a maximum of one connected device at a time, reducing the risk of conflicting transmissions and simplifying session management.

To improve robustness and error handling, explicit error-marking and recovery mechanisms were integrated. The microcontroller includes a dedicated button function that resets the current row to an empty state, allowing the user to correct mistakes before transmitting data. On the Android side, the app supports deleting the most recently received row, enabling rollback in the event of an incorrect transmission. The system also verifies successful reception at both ends by tracking confirmation flags for BLE transmission and SD card storage, ensuring that each row is not only received by the app but also successfully written to persistent storage on the microcontroller.

File management was expanded significantly to support real data collection workflows. The app now allows users to create new CSV files both on the Android device and on the microcontroller's SD card, enabling separate datasets to be maintained for different sessions or experiments. Old files can be deleted or cleared to manage storage space, and redundant data storage is used to reduce the risk of data loss. By maintaining synchronized copies of files on the mobile device and the SD card, the system ensures that collected data remains accessible even if one storage medium fails.

Power management considerations were also incorporated once the system moved beyond simulation. A dedicated power-related control allows the user to decide when the microcontroller advertises its BLE network, rather than broadcasting continuously. This significantly reduces power consumption and prolongs battery life, which is critical for extended field deployments. The Android app reflects this state by clearly indicating connection status and only attempting communication when advertising is enabled.

The final application preserves all functionality developed during the simulation phase, including dynamic table display, real-time aggregation, editable headers, and CSV export, while extending the system to support secure, reliable communication with physical hardware. As shown in Figure 22, the user interface now reflects the full feature set of the deployed app, including Bluetooth connection status, file selection and creation controls, a live CSV viewer, total counters, and data management buttons. This evolution from simulation to hardware-integrated application demonstrates how a carefully designed software architecture can scale from development testing to real-world deployment with minimal structural changes, while significantly expanding functionality and reliability.

KoboToolbox Upload

Bluetooth provides a convenient method for real-time data transfer, however it is not always accessible. Users may face hardware limitations, connectivity issues, or distance constraints that prevent them from using Bluetooth at the time of data collection. Additionally, some users may prefer to manage and review data offline before uploading it to the central repository. To address these scenarios, we developed an API-based CSV upload module, which allows users to export their device data onto a microSD card, convert it into a CSV file, and upload it directly to KoboToolbox through a graphical interface (see Figure 23). This alternative ensures that data collection remains flexible and reliable, even in environments where Bluetooth access is not feasible.

Figure 23: Graphics User interface for the API uploader. Users enter credential information in the first page and map their csv columns to form fields in the next page. Data is immediately uploaded to KoboToolbox.

The core functionality of the API upload module is around reading, processing, and submitting CSV data in a format compatible with KoboToolbox. The module begins by normalizing the CSV headers to maintain consistency, ensuring that column names are stripped of special characters and standardized for mapping. Timestamps recorded in milliseconds are converted into ISO8601 format relative to a user-specified start date, allowing precise chronological representation of events. This preprocessing step ensures that the temporal data aligns accurately with KoboToolbox's requirements and provides a reliable basis for further analysis. Once the CSV is prepared, the module interfaces directly with the KoboToolbox API. Users provide their credentials and survey link, from which the survey ID is extracted and used to retrieve the form definition. The graphical user interface provides a user-friendly workflow for mapping CSV columns to the corresponding KoboToolbox form fields. Users can assign columns representing buttons or other data points to the appropriate survey fields, while also designating a column to serve as the timestamp. After mapping, the module generates XML instances for each row in the CSV, embedding metadata such as start, end, instanceID, and versioning information. Each XML instance is then submitted to the KoboToolbox API over HTTPS, ensuring secure transmission. Simultaneously, all XML submissions are saved locally to the user-specified output folder, providing a backup and an audit trail in case of submission errors.

The module also provides real-time feedback during the upload process. As each row is submitted, the GUI displays the server response, indicating whether the submission was successful or if an error occurred. This immediate feedback allows users to quickly identify and correct any issues, such as missing fields or connectivity problems. The API upload module is distributed as a downloadable ZIP folder, containing a standalone executable for both Windows and macOS. Users can simply extract the folder and run the application without the need to install Python or any additional dependencies, making deployment straightforward and accessible across platforms.

Software Deliverable

The full software package developed for the KoboClicker system is available at https://github.com/jcurcio3105/KoboClicker, which includes the complete Arduino firmware for the nRF52840 Xiao Sense, the Android application source code, the KoboToolbox API upload tool, and all supporting modules. The repository also contains detailed README files outlining required libraries, installation steps, development environment setup, and instructions for running, integrating, and extending each component, ensuring full reproducibility and ease of deployment.

XII. Implementation and Next Steps

Integration

Bringing together the work of the mechanical, electrical, and software teams into a neatly packaged prototype was one of the most important and challenging stages of this project. We began by transferring the finalized electrical circuit from a conventional breadboard to a solderable breadboard (see Figure 20). With each iteration of our solderable breadboard, we ran several software tests before assessing compatibility with the mechanical shell (see Figure 24). The first of these tests was to ensure that both button ladders were working correctly. We ran code to read voltages off the two analog pins when different buttons were pressed and recalibrated our voltage thresholds for the new board. If we saw the button ladders printing 0V or 5V continuously or fluctuating randomly, it meant that there was a short circuit.

After ensuring both button ladders were working properly, we tested our protocol for writing to the microSD card and sending data to the app via BLE (see Figure 25). While the BLE usually worked, we faced issues writing to the microSD because of the integrity of our soldered connections. Instead of soldering the microSD down to the breadboard, we attached it via socket-to-header wires so it would be free-floating. While this allowed us to maneuver it into the designated microSD slot on the shell, the wires were flimsy and would often rip off as we conducted our tests. Given more time, we would solder a permanent microSD socket directly to the board, but after double-checking our soldered wires we were always able to get the SD card to initialize.

After these software tests were complete, the next step was to fit the board into the mechanical shell and redimension if necessary. We checked that the shell lid holes were aligned with the buttons on the breadboard, the USB-C opening was aligned with the port on our microcontroller, and the microSD was able to be slotted into its designated spot on the side of our shell. The microSD alignment was the most challenging aspect because of the flimsiness of the wires connecting it to the board. We had to increase the shell width several times to accommodate these wires and make sure they did not exceed their bend radius. We also had to redimension to make room for the wire connecting the battery to the QT Py BFF.

Going into the integration process, we were experimenting with two ways to close our shell: a screw-in design and a snap-fit design. The screw-in design was easier to incorporate with the O-rings for waterproofing while the snap fit design was sleeker and easier to open and close. We decided to move forward with the snap fit design during testing because it was necessary to remove the shell lid many times to rearrange components. However, due to its drop-test stability and O-ring compatibility, a screw-in design is a more viable long-term option (Fig. 26).

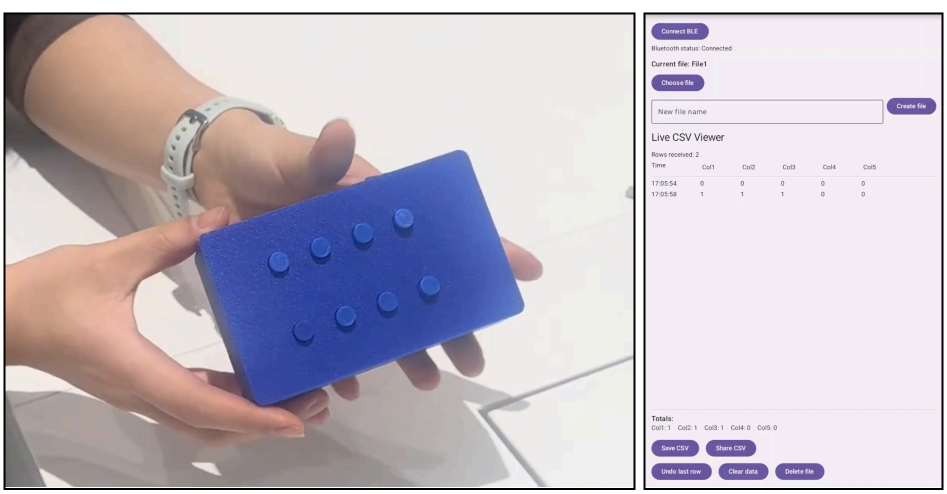


Figure 24: Pictures of the finalized shell design and button presses being populated in the data collection app. For testing purposes, data was streamed over Bluetooth every time the NEXT button (button 1) was pressed.

Timestamp	Col1	Col2	Col3	Col4	Col5
1765038964145	1	1	1	0	0
1765038970090	1	1	0	0	0
1765038974623	0	0	0	1	1
1765038979543	0	0	1	0	0
1765038984277	1	0	0	0	1
1765038987590	0	0	0	0	0
1765038988273	0	0	0	0	0
1765038991248	0	1	0	0	0

Figure 25: A CSV file containing timestamped data from button presses from five buttons logged on the microSD card. The microSD offers a physical, secondary backup of all data.

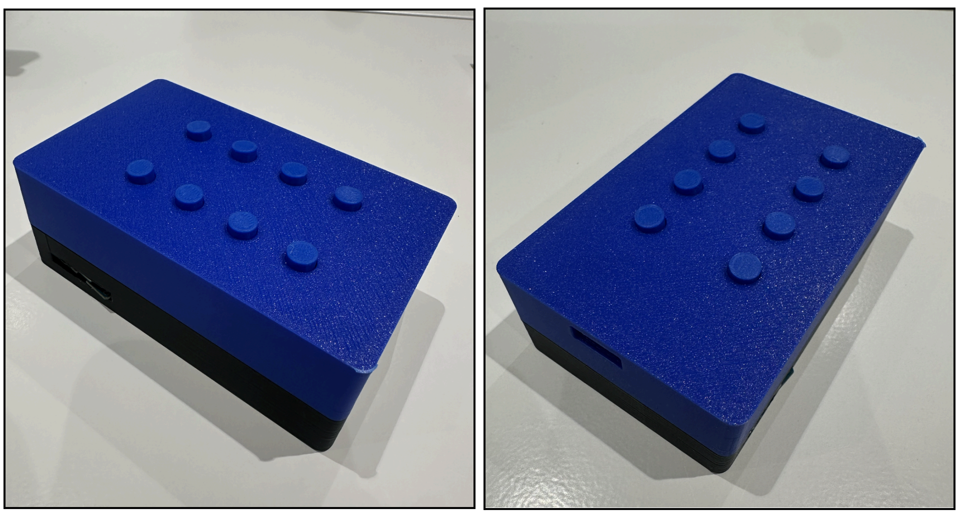


Figure 26: Images of the complete 3D printed shell with the microSD port (left) and the USB-C port (right) visible.

Next Steps

The current KoboClicker prototype demonstrates a complete proof of concept for translating physical button-based inputs into structured digital data compatible with KoboToolbox. Both real-time Bluetooth transmission and offline CSV-based data upload have been validated, confirming the feasibility of the overall system architecture. However, the prototype remains an early-stage implementation, with limitations in physical robustness, electronic integration, and software interoperability. Advancing the system toward field readiness requires focused improvements across the device's physical design, internal electronics, and software integration.

1. **Ergonomic and Weatherproof Clicker Design**

A key area for future development is the refinement of the clicker's physical form factor to better support real-world humanitarian use. The existing prototype relies on an enclosure and internal layout that prioritize functionality over ergonomics, durability, and environmental protection. Future iterations should focus on improving hand comfort, button placement, and tactile feedback, while maintaining a compact and lightweight device that can be operated repeatedly over long periods. Given the intended deployment contexts, the enclosure must be resistant to dust, moisture, and rough handling.

From a manufacturing standpoint, early-stage production should emphasize low-risk and flexible approaches that support rapid iteration. Modifying off-the-shelf enclosures with machined cutouts provides a cost-effective and fast path for pilot deployments while avoiding upfront tooling costs. As the design matures and the desired form factor becomes more defined, CNC-machined enclosures offer greater customization for small production runs. Injection molding becomes attractive only once the design is stable and production volumes are high enough to amortize tooling costs. Material selection should prioritize outdoor durability, with UV-stabilized ABS or ASA serving as practical defaults, while waterproofing should be addressed holistically through gasketed seams, controlled compression, and careful treatment of enclosure penetrations. Button interfaces, in particular, require careful design, as they are a primary source of potential water ingress. Detailed information on future mechanical considerations has been included in Appendix E.

2. Transition to a Custom Printed Circuit Board

Another important area for future development is transitioning from the current breadboard-based electronics to a custom printed circuit board. A custom PCB would significantly improve mechanical reliability, reduce internal wiring complexity, and enable a more compact internal layout. In addition to improving robustness, a dedicated PCB would support better power management, cleaner signal routing for button inputs, and more reliable integration of Bluetooth and storage components. This transition would also simplify enclosure design by reducing internal volume and enabling consistent mounting strategies. Designing the PCB with modular connectors and test points would further support debugging, maintenance, and future feature expansion during pilot deployments and beyond.

3. Integration with KoboCollect

Software integration remains a critical pathway for improving usability and adoption. While this project includes a custom Android application that demonstrates successful Bluetooth-based data capture from the clicker, this application was developed as a proof of concept. The source code of KoboCollect, the primary mobile data collection application used with KoboToolbox, was difficult to distinguish from ODK collect and edit, which limits direct integration during the scope of this work.

Future development should focus on enabling direct interoperability between the clicker and KoboCollect's data collection workflow. Achieving this integration would allow button press events to be recorded as native KoboCollect submissions in real time, eliminating the need for parallel applications and simplifying training and deployment for end users. In parallel, the API-based CSV upload workflow developed in this project provides a robust alternative for transferring offline data into KoboToolbox in low-connectivity or restricted-device environments. To support continued development, the project maintains a publicly available GitHub repository containing comprehensive documentation, source code, and system architecture details, providing a strong foundation for future teams to extend, refine, and integrate the KoboClicker into existing humanitarian data collection workflows.

X. Conclusion

Over the course of this project, our team developed the KoboClicker as a practical, proof-of-concept tool aimed at strengthening real-time data collection in humanitarian field environments. Guided by our initial investigation with HHI, we focused on a challenge that appeared consistently across programs: the difficulty of capturing fast, reliable, and structured field data using existing devices. The KoboClicker addresses this gap by offering a durable, low-cost input device that pairs tactile data entry with both Bluetooth transmission and offline storage. Together, these capabilities demonstrate a feasible pathway to extend KoboToolbox's utility in settings where speed, situational awareness, or device fragility pose limitations.

The prototype integrates mechanical, electrical, and software subsystems into a functioning device that can record button-based inputs, securely transmit data, and maintain a verified local log. While this version is not yet optimized for field deployment, it provides clear evidence of technical viability and outlines the design considerations necessary for future refinement. Mechanical testing confirmed that a compact, robust form factor is achievable, and electrical validation showed reliable multi-button detection, low-power operation, and consistent data logging. Likewise, the software workflow, with both an Android interface and an API-based upload tool, illustrates how hardware input could be incorporated into KoboToolbox's existing ecosystem.

Looking ahead, several focused steps would strengthen the device's readiness for operational use. These include transitioning to a custom PCB to improve reliability, refining the enclosure for ergonomics and environmental protection, and pursuing deeper integration with KoboCollect to streamline field workflows. Field testing with HHI program teams would also be essential for understanding user preferences, contextual constraints, and long-term durability needs.

Overall, the KoboClicker establishes a tangible starting point for expanding how humanitarian responders collect and manage quantitative data. By clarifying technical requirements and mapping a feasible development pathway, this project provides HHI with a foundation for future iterations toward a scalable, field-ready tool that supports evidence-based humanitarian action.

XI. References

[1] Harvard Humanitarian Initiative, "Harvard Humanitarian Initiative," [Online]. Available: https://www.hhi.harvard.edu/. [Accessed: 14-Sep-2025].

[2] United Nations Office for the Coordination of Humanitarian Affairs, "United Nations Office for the Coordination of Humanitarian Affairs," 2024. [Online]. Available: https://www.unocha.org/. [Accessed: 14-Sep-2025].

[3] UNDP, "United Nations Development Programme," [Online]. Available: https://www.undp.org/home. [Accessed: 14-Sep-2025].

[4] ICRC, "International Committee of the Red Cross," Aug. 31, 2016. [Online]. Available: https://www.icrc.org/en. [Accessed: 14-Sep-2025].

[5] United Nations, "Deliver Humanitarian Aid," 2021. [Online]. Available: https://www.un.org/en/our-work/deliver-humanitarian-aid. [Accessed: 14-Sep-2025].

[6] UNICEF, "The Humanitarian-Development Nexus," [Online]. Available: https://www.unicef.org/eu/humanitarian-development-nexus. [Accessed: 14-Sep-2025].

[7] J. H. S. Lie, "The Humanitarian-Development Nexus: Humanitarian Principles, Practice, and Pragmatics," *Journal of International Humanitarian Action*, vol. 5, no. 1, Dec. 2020. https://doi.org/10.1186/s41018-020-00086-0.

[8] F. Mulder, "The Paradox of Externally Driven Localisation: A Case Study on How Local Actors Manage the Contradictory Legitimacy Requirements of Top-Down Bottom-Up Aid," *Journal of International Humanitarian Action*, vol. 8, no. 1, Jul. 2023. doi: https://doi.org/10.1186/s41018-023-00139-0.

[9] United Nations High Commissioner for Refugees, “Cluster Approach,” [Online]. Available: https://emergency.unhcr.org/coordination-and-communication/cluster-system/cluster-approach. [Accessed: 09-Sep-2025]

[10] ICVA, “About OCHA,” [Online]. Available: https://www.icvanetwork.org/uploads/2021/08/Topic-Four-Briefing-paper-OCHA-and-NGOs-in-Humanitarian-Coordination.pdf. [Accessed: 14-Sep-2025].

[11] WHO, “What We Do,” 2025. [Online]. Available: https://www.who.int/about/what-we-do. [Accessed: 14-Sep-2025].

[12] UNICEF, “Our Values,” [Online]. Available: https://www.unicef.org/careers/media/1041/file/UNICEF%27s_Competency_Framework.pdf. [Accessed: 14-Sep-2025].

[13] WFP, “WFP at a Glance,” [Online]. Available: https://docs.wfp.org/api/documents/WFP-0000168147/download/?_ga=2.58674157.616840726.1757864775-132855539.1757864775. [Accessed: 14-Sep-2025].

[14] UNHCR US, “What We Do,” [Online]. Available: https://www.unhcr.org/us/what-we-do. [Accessed: 14-Sep-2025].

[15] WHO, “WHO EMRO: Child and Adolescent Health in Humanitarian Settings: Operational Guide,” 2023. [Online]. Available: https://www.emro.who.int/cah-guide/chapter1-2.html. [Accessed: 14-Sep-2025].

[16] Berkeley Library, “Library Guides: Non Governmental Organizations (NGOs): Humanitarian,” 2019. [Online]. Available: https://guides.lib.berkeley.edu/c.php?g=496970&p=3626027. [Accessed: 14-Sep-2025].

[17] OCHA, “Localization,” [Online]. Available: https://www.unocha.org/localization. [Accessed: 14-Sep-2025].

[18] N. Mitsuyoshi, et al., “The Army’s Role in Domestic Disaster Response: Preparing for the Next Catastrophe,” 2014. [Online]. Available: https://ctpp.sanford.duke.edu//wp-content/uploads/sites/16/2015/09/USAWCDUKECRP19May14_FINALTONICHOLS_.pdf. [Accessed: 14-Sep-2025].

[19] F. C. Cuny, “The Lost American — Use of the Military in Humanitarian Relief,” *PBS Frontline*, [Online]. Available: https://www.pbs.org/wgbh/pages/frontline/shows/cuny/laptop/humanrelief.html. [Accessed: 14-Sep-2025].

[20] European Commission, “Humanitarian Principles,” 2023. [Online]. Available: https://civil-protection-humanitarian-aid.ec.europa.eu/who/humanitarian-principles_en. [Accessed: 14-Sep-2025].

[21] The UN Refugee Agency, "Humanitarian Principles," Mar. 14, 2025. [Online]. Available: https://emergency.unhcr.org/protection/protection-principles/humanitarian-principles. [Accessed: 14-Sep-2025].

[22] Better World Campaign, "Before the First Delivery: Why Neutrality is the Foundation for Effective Humanitarian Assistance," July 15, 2025. [Online]. Available: https://betterworldcampaign.org/blog/before-the-first-bag-of-rice-why-neutrality-is-essential-for-effective-humanitarian-aid. [Accessed: 14-Sep-2025].

[23] P. McIlreavy, "Enough is Enough. It's Time to Protect Aid Workers," *The Guardian*, Sep. 23, 2016. [Online]. Available:

https://www.theguardian.com/global-development-professionals-network/2016/sep/23/enough-is-enough-its-time-to-protect-aid-workers. [Accessed: 14-Sep-2025].

[24] Harvard Humanitarian Initiative, “Open House,” Cambridge, MA, Sep. 11, 2025.

[25] Harvard Humanitarian Initiative, "Harvard Humanitarian Initiative Impact Report 2023," [Online]. Available: https://hhi.harvard.edu/sites/g/files/omnuum6866/files/final_2023_hhi_impact_report_3.pdf. [Accessed: 14-Sep-2025].

[26] United Nations, “Crisis and Emergency Response,” [Online]. Available: https://www.un.org/en/global-issues/crisis-and-emergency-response. [Accessed: 14-Sep-2025].

[27] Better World Campaign, “The Impact of Foreign Aid Cuts,” [Online]. Available: https://betterworldcampaign.org/impact-of-foreign-assistance-cuts. [Accessed: 14-Sep-2025].

[28] M. Barnett and T. G. Weiss, “Humanitarianism Contested: Where Angels Fear to Tread,” 1st ed. London: Routledge, 2011. doi: 10.4324/9780203829301.

[29] Harvard Humanitarian Initiative, “The Need for Humanitarian Research: Addressing Emerging Challenges in a Complex World,” *Harvard Humanitarian Initiative News*, [Online]. Available: https://hhi.harvard.edu/news/need-humanitarian-research-addressing-emerging-challenges-complex-world. [Accessed: 14-Sep-2025].

[30] P. Vesco et al., “The Impacts of Armed Conflict on Human Development: A Review of the Literature,” *World Development*, vol. 187, no. 187, p. 106806, Nov. 2024, doi: https://doi.org/10.1016/j.worlddev.2024.106806.

[31] United Nations Office for the Coordination of Humanitarian Affairs, “Humanitarian Access Snapshot – Gaza Strip,” Apr. 2024. [Online]. Available: https://www.unocha.org/publications/report/occupied-palestinian-territory/humanitarian-access-snapshot-gaza-strip-1-30-april-2024. [Accessed: 14-Sep-2025].

[32] Reuters, “Killing of Aid Workers Surges to Record High During Gaza War, UN says,” *Reuters*, Aug. 19, 2025. [Online]. Available: http://reuters.com/world/middle-east/killing-aid-workers-surges-record-high-during-gaza-war-un-says-2025-08-19/? [Accessed: Sep. 14, 2025].

[33] United Nation News, “World Humanitarian Day 2025: Aid Workers Mull Record Toll of Their Own,” *United Nations News*, Aug. 19, 2025. [Online]. Available: https://news.un.org/en/story/2025/08/1165678. [Accessed: Sep. 14, 2025].

[34] A. De Waal, “Mass Starvation: The History and Future of Famine.” *Wiley*, 2018. [Online]. Available: https://www.wiley.com/en-us/Mass+Starvation%3A+The+History+and+Future+of+Famine-p-9781509524662. [Accessed: Sep. 14, 2025].

[35] World Food Programme Staff, “All You Need to Know About the WFP Food Basket,” Jan. 8, 2025. [Online]. Available: https://www.wfp.org/stories/wfp-food-basket.[Accessed: Sep. 14, 2025].

[36] World Central Kitchen, “Chefs For Gaza: Let Us Cook,” May 13, 2025. [Online]. Available: https://wck.org/news/let-us-cook. [Accessed: Sep. 14, 2025].

[37] S. Kurdi, et al., “The Struggle to Get Food Aid from Egypt to Gaza: Insights from the Egyptian Food Bank,” *IFPRI Blog*, Aug. 4, 2025. [Online]. Available:

https://www.ifpri.org/blog/the-struggle-to-get-food-aid-from-egypt-to-gaza-insights-from-the-egyptian-food-bank/. [Accessed: Sep. 14, 2025].

[38] S. Beltrami, "Humanitarian Airdrops: Can Life-Saving Food Fall From the Sky?," *World Food Programme*, Aug. 20, 2025. [Online]. Available: https://www.wfp.org/stories/airdrops-humanitarian-emergency-un-world-food-programme-sudan-syria. [Accessed: Sep. 14, 2025].

[39] United Nations Office for the Coordination of Humanitarian Affairs, "Considerations For Delivery of Humanitarian Aid During Ceasefire — Gaza," [Online]. Available: https://www.unocha.org/considerations-delivery-humanitarian-aid-during-ceasefire-gaza. [Accessed: Sep. 14, 2025].

[40] World Bank, "Climate Change Could Force 216 Million People to Migrate Within Their Own Countries by 2050," Sep. 13, 2021, [Online]. Available: https://www.worldbank.org/en/news/press-release/2021/09/13/climate-change-could-force-216-million-people-to-migrate-within-their-own-countries-by-2050. [Accessed: Sep. 14, 2025].

[41] Oxfam America, "Reform Food Aid," Oxfam. [Online]. Available: https://www.oxfamamerica.org/explore/issues/humanitarian-response-and-leaders/hunger-and-famine/food-aid/. [Accessed: Sep. 14, 2025].

[42] K. Tierney, *Disasters: A Sociological Approach*, 2nd ed. John Wiley & Sons, 2025.

[43] "Somalia Suspends Talks With Government," *The New York Times*, Mar. 9, 2010. [Online]. Available: https://www.nytimes.com/2010/03/10/world/africa/10somalia.html. [Accessed: Sept. 21, 2025].

[44] S. Seddon, "What is USAID and why is Trump poised to 'close it down'?," *BBC News*, Feb. 7, 2025. [Online.] Available: https://www.bbc.com/news/articles/clyezjwnx5ko. [Accessed: Sep. 21, 2025].

[45] J.-K. Westendorf, J. Bian, M. Daigle, A. Potts, K. Jennings, M. Reddick, C. C. Massonneau, G. Gamhewage, and M. E. Mahmoud, "Sexual exploitation, abuse and harassment in humanitarian contexts," Bull. World Health Organ., vol. 102, no. 12, pp. 888–894, Sep. 25, 2024, doi: 10.2471/BLT.24.291655.

[46] I. Albanti, "Overview of the Harvard Humanitarian Initiative," presentation, Harvard Humanitarian Initiative, Sep. 10, 2025.

[47] I. Albanti et al., "Introduction to HHI programs," HHI Open House, Harvard Humanitarian Initiative, Sep. 11, 2025.

[48] I. Albanti et al., "Team Q&A session," Harvard Humanitarian Initiative, Sep. 18, 2025.

[49] Strucke and M. J. Cohen, "What Is Happening to U.S. Humanitarian Assistance? Will the United States Continue to Save Lives?" CSIS Analysis, Apr. 18, 2025. [Online]. Available: https://www.csis.org/analysis/what-happening-us-humanitarian-assistance-will-united-states-continue-save-lives. Accessed: Sep. 22, 2025.

[50] Harvard Humanitarian Initiative, "Geospatial Workshop," [Online]. Available: https://hhi.harvard.edu/geospatial-workshop. Accessed: Sep. 22, 2025.

[51] Organisation for Economic Co-operation and Development, *Evaluation Criteria: Adapted Definitions and Principles for Use*, DCD/DAC(2019)58/FINAL, 11 December 2019. [Online]. Available: https://one.oecd.org/document/DCD/DAC%282019%2958/FINAL/en/pdf

[52] CHS Alliance, “The Standard,” *Core Humanitarian Standard*, 2024. [Online]. Available: https://www.corehumanitarianstandard.org/the-standard

[53] Sphere Association, *Handbook: Humanitarian Charter and Minimum Standards in Humanitarian Response*. [Online]. Available: https://spherestandards.org/handbook/

[54] ES-96 Team, “Phase 2 Report Presentation” (oral presentation, Prof. David Mooney, Allston, MA, Oct. 6, 2025).

[55] 3DEEE, “3D-Printing Filament Length Weight Volume Calculator,” *tools.3deee.ch*. [Online]. Available: https://tools.3deee.ch/f_calculator

[56] Smooth-On, *Mold Star Silicone Making Rubber*. [Online]. Available: https://www.amazon.com/Mold-Star-Silicone-Making-Rubber/dp/B00ETAY8RI?th=1

[57] Hukado, “Adjustable Hand Wrist Straps Lanyard,” *Amazon*. [Online]. Available: http://amazon.com/Hukado-Adjustable-Lanyard-Flashlight-Keychains/dp/B07Q2QDG3Q/ref=sr_1_2_sspa

[58] Evan Designs, “Chip, Nano and Pico LED Lights,” *EvanDesigns.com*. [Online]. Available: https://evandesigns.com/products/chip-nano-pico-leds?variant=39985934172208

[59] Adafruit, “Mini Oval Speaker – 8 Ω 1 W,” Product ID 3923, *Adafruit Industries*. [Online]. Available: https://www.adafruit.com/product/3923

[60] Adafruit, “Vibrating Mini Motor Disc (Product ID 1201),” *Adafruit Industries*. [Online]. Available: https://www.adafruit.com/product/1201

[61] UNHCR, “United Nations High Commissioner for Refugees Official Website.” https://www.unhcr.org/

[62] KoboToolbox, KoboToolbox Official Website. https://www.kobotoolbox.org/

[63] Center for Disaster Preparedness, “Advantages and Disadvantages of Using KoboToolbox,” Inclusive Data Management System Guidebook. https://www.cdpf.org.ph/post/advantages-and-disadvantages-of-using-kobotoolbox

[64] Magalie Salazar, “Needs Assessments,” OCHA Knowledge Base. https://knowledge.base.unocha.org/wiki/spaces/hpc/pages/3992616975/Needs+Assessments

[65] Avient Corporation, “Ergonomic Design Guide,” Avient, 2021. [Online]. Available: https://www.avient.com/sites/default/files/2021-11/avient-design-ergonomic-design-guide.pdf

XII. Appendix

Appendix A: Acronyms

- CARE — Cooperative for Assistance and Relief Everywhere
- CCCM — Camp Coordination and Camp Management
- CSO — Civil Society Organization
- FAO — Food and Agriculture Organization of the United Nations
- HHI — Harvard Humanitarian Initiative
- IFRC — International Federation of Red Cross and Red Crescent Societies
- IMC — International Medical Corps
- INGO — International Non-Governmental Organization
- IOM — International Organization for Migration
- IRC — International Rescue Committee
- MSF — Médecins Sans Frontières (Doctors Without Borders)
- NGO — Non-Governmental Organization
- OCHA — Office for the Coordination of Humanitarian Affairs
- PIH — Partners In Health
- UN — United Nations
- UNDP — United Nations Development Programme
- UNHCR — United Nations High Commissioner for Refugees
- UNICEF — United Nations Children's Fund
- USAID/OFDA — United States Agency for International Development / Office of U.S. Foreign Disaster Assistance
- WFP — World Food Programme
- WHO — World Health Organization

Appendix B: Global Humanitarian Clusters and Lead Agencies

Cluster	Lead / Co-Lead Agency
Health	World Health Organization (WHO)
Water, Sanitation, and Hygiene (WASH)	United Nations Children's Fund (UNICEF)
Nutrition	United Nations Children's Fund (UNICEF)

Cluster	Lead / Co-Lead Agency
Food Security	World Food Programme (WFP) / Food and Agriculture Organization (FAO)
Education	UNICEF / Save the Children
Protection	United Nations High Commissioner for Refugees (UNHCR)*
Shelter	International Federation of Red Cross and Red Crescent Societies (IFRC) in natural disasters / UNHCR in conflict settings
Camp Coordination and Camp Management (CCCM)	International Organization for Migration (IOM) / UNHCR
Logistics	World Food Programme (WFP)
Emergency Telecommunications	World Food Programme (WFP)
Early Recovery	United Nations Development Programme (UNDP)

*Protection includes multiple Areas of Responsibility led by different agencies depending on the protection concern [14–15].

Appendix C: Harvard Humanitarian Initiative Programs Overview

Program	Primary Focus	Leadership	Key Activities	Geographic Scope
Emergency Health Systems (EHS)	Strengthening emergency & trauma care systems during peacetime to improve crisis response	Dr. Catalina Gonzalez Marquez	Emergency and trauma care training; development of locally led training models; exploration of virtual reality for remote training	Ukraine & other crisis-affected regions

Noncommunicable Diseases and Conflict (NDC)	Management of chronic diseases in humanitarian and conflict settings	Dr. Sylvia Kehlenbrink	Insulin delivery in crisis settings; cold-chain logistics research; improving care for noncommunicable diseases	Global
Children in Crisis	Pediatric health and mental health in humanitarian contexts	Carolyn Baer (partial leadership	School-based mental health education: focus on connectivity and access to electricity	Ukraine, Jordan, Ethiopia (Tigray region)
Resilient Communities	Disaster preparedness and community resilience	Dr. Enzo Bollettino	Early warning system development; nationwide crisis surveys; partnerships with local organizations	The Philippines and other disaster-prone regions
Humanitarian Geoanalytics	Spatial analysis of humanitarian crises and population movement	Dr. Erica Nelson	Geospatial analysis of migration, conflict, public health, and land use; ethical data governance	Global
Humanitarian Academy at Harvard	Education and training for humanitarian professionals and students	HHI Faculty and Staff	Humanitarian Response Intensive Course (HRIC); Monitoring and Evaluation Workshop; Urban Humanitarian Emergencies Course	Global
Transitional Justice	Intersection of humanitarian response, recovery, and development	Dr. Phuong Pham; Dr. Patrick Vinck	Development and piloting of KoboToolbox; data collection and evaluation methods; post-conflict recovery research	Global

Appendix D: Synthesis Mapping Exercise

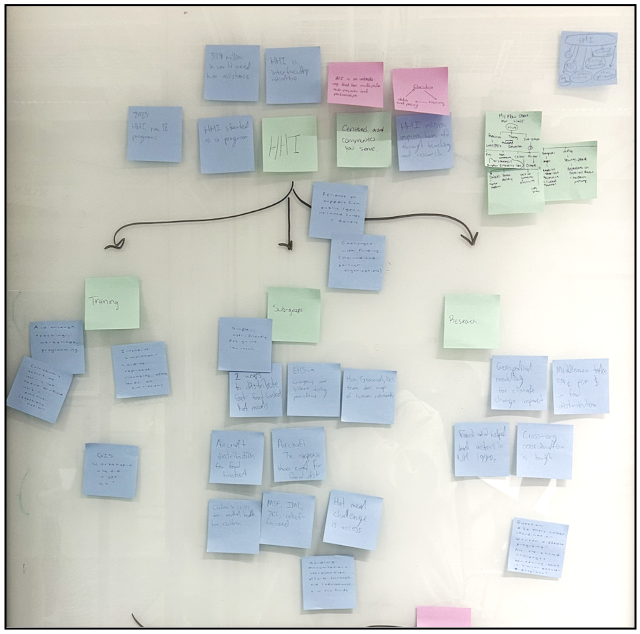


Figure D1: Visual categorization of challenges identified across HHI programs during the synthesis process. The mapping highlights data collection, analysis, and standardization as cross-cutting challenges affecting multiple programs, while education-related challenges are more program-specific.

Appendix E: Enclosure Manufacturing and Design Considerations

E.1 Scope and Design Assumptions

This appendix summarizes key considerations for transitioning the KoboClicker enclosure from prototype to production. The analysis evaluates manufacturing approaches, material selection, waterproofing strategies, button interfaces, logistics, and cost tradeoffs for a small weatherproof electronics enclosure. Baseline assumptions include a two-piece housing approximately 2 × 4 × 1 inches in size with a total mass of roughly 2 ounces. The objective is to identify realistic production pathways that balance cost, durability, and technical risk across different production scales. A central driver across all approaches is production volume. At low quantities, flexibility, speed, and minimal upfront cost dominate decision-making, whereas at higher volumes, per-unit cost efficiency justifies greater upfront investment. Across scenarios, early-stage production is best served by off-the-shelf enclosures with minor modifications, while custom injection molding becomes attractive only after the design stabilizes and quantities increase sufficiently to amortize tooling costs.

E.2 Manufacturing Pathways

Three primary manufacturing pathways were evaluated:

1. **Off-the-shelf enclosures with modifications**

   Commercially available project boxes or rated enclosures can be modified with cutouts for buttons, ports, and mounting features. This approach minimizes non-recurring engineering effort and avoids tooling costs, making it the fastest and lowest-risk option for prototypes and pilot deployments. Unit costs are typically in the single-digit dollar range, with additional costs driven by machining, labeling, and assembly.

2. **CNC-machined custom enclosures**

   CNC machining from plastic stock enables a custom form factor without mold tooling and supports rapid iteration with good dimensional control. This approach is well suited for small production runs where geometry is still evolving. For runs on the order of 100 units, typical costs fall in the low-teens to low-twenties per enclosure, depending on machining complexity and finishing requirements.

3. **Injection molding**

   Injection molding is the standard approach for high-volume production, offering consistent cosmetics, repeatability, and integrated features such as bosses, ribs, and snap fits. However, at low quantities, tooling costs dominate total cost, making injection molding unattractive for small runs. It becomes cost-effective only once production volumes reach several hundred units or more.

E.3 Material Selection and Waterproofing Strategy

Material choice primarily affects outdoor durability, UV resistance, and long-term appearance rather than per-unit cost at this scale. UV-stabilized ABS is an economical option for medium-duty outdoor use. ASA provides superior UV stability and color retention for prolonged outdoor exposure. Polycarbonate offers high impact resistance, making it suitable for harsh handling environments, though surface finish and chemical exposure must be considered. Material blends such as PC/ABS or PC/ASA balance toughness, moldability, and weatherability. Reinforced nylon offers high stiffness and thermal resistance but is generally better suited for internal structural components due to moisture absorption.

Waterproofing should be approached as a system-level design problem, as most failures occur at interfaces rather than through bulk materials. A practical and serviceable solution uses a perimeter gasket between enclosure halves with controlled screw compression, combined with sealed treatments for all penetrations. Tongue-and-groove seams increase leak path length and improve alignment. Permanent sealing methods such as ultrasonic welding provide robust protection but reduce serviceability. Environmental pressure changes should also be considered, as temperature-driven pressure cycling can draw moisture through weak points.

E.4 Button Interface Design

Button interfaces are a major contributor to both usability and ingress risk. Early-stage solutions include internal tact switches paired with external silicone boots, balancing simplicity and sealing effectiveness. Membrane keypads or graphic overlays offer strong water resistance and minimal mechanical complexity but provide softer tactile feedback. Silicone rubber keypads consolidate multiple buttons and provide reliable sealing through controlled compression but require custom tooling. Sealed metal or piezoelectric buttons offer the highest durability in harsh environments at increased cost. In early iterations, minimizing penetrations while maintaining acceptable usability is typically the most effective approach.

E.6. Cost Modeling and Scaling Considerations

A unit cost of an enclosure can be modeled as:

$$\text{Unit Cost} = \frac{(T+S)}{Q} + (C_m + C_p + C_f)$$

Where

T = tooling cost

S = setup or non-recurring cost

Q = production quantity

C_m = material cost per unit

C_p = processing cost per unit

C_f = finishing and packaging cost per unit

This model illustrates why low-volume injection molding is expensive: when *Q* is small, the tooling cost term dominates. A break-even comparison between CNC machining and injection molding can be expressed as:

$$C_{CNC(Q)} = S_{CNC} + u_{CNC} * Q$$

$$C_{IM(Q)} = T_{IM} + u_{IM} * Q$$

The break-even quantity Q is:

$$Q = \frac{(T_{IM} - S_{CNC})}{u_{CNC} - u_{IM}}$$

Overall, the recommended progression is to begin with off-the-shelf enclosures, transition to CNC machining if a custom form factor is required, and adopt injection molding only when design stability and volume justify tooling investment. Material, waterproofing, and button choices should be considered holistically, as real-world failures most often arise at interfaces rather than in the primary enclosure material.

Appendix F: Link to GitHub

The complete GitHub repository can be found at https://github.com/jcurcio3105/KoboClicker.

Appendix G: Link to STL Files

STL files for the various designs included in this report can be found at this link.