A Preliminary Multi-Level Service Blueprint of End-User Development in Teleoperated Socially Assistive Robots

Saad Elbeleidy
selbeleidy@mines.edu
Colorado School of Mines
Golden, Colorado, USA

Alexandra Bejarano
abejarano@mines.edu
Colorado School of Mines
Golden, Colorado, USA

Tom Williams
twilliams@mines.edu
Colorado School of Mines
Golden, Colorado, USA

ABSTRACT

End-user development (EUD) in Human-Robot Interaction (HRI) aims to expand access and applicability of robotics by allowing people with minimal robotics expertise to use and benefit from robots. To develop these systems, HRI researchers often rely on human-centered design methods that focus on the robot user. These design methods justifiably center robot users as the ones for whom to design. However, focusing solely on the user can miss the bigger picture of resources, dependencies, and other people involved in the process. Recent research in teleoperated Socially Assistive Robots suggests that authoring tasks in preparation may be performed through invisible labor. This highlights the need for researchers to consider and design for the bigger picture of their robots' impact. Service design methods such as service blueprinting support designers in outlining the bigger picture of any system. In this paper, we use teleoperated Socially Assistive Robotics as a case study for End User Development of Teleoperated Robots. We use service blueprinting of our use case to outline the visible and invisible layers of End User Development of Teleoperated Robots and present preliminary results. Our service blueprint highlights the different roles that users can take on and defines various capabilities needed to support a smooth delivery of robot interactions.

KEYWORDS

Robot End-User Development, Robot End-User Programming, Teleoperated Socially Assistive Robots

1 INTRODUCTION

End-user development (EUD) in robotics extends robot applications beyond the expertise of the robot developers. Tools for robot programming empower people to apply their domain expertise and use robots in new contexts. However, this requires tool developers to account for the various levels of expertise and contexts in which people may use their tools. To appeal to a broad audience, developers rely on several modalities, such as visual programming [1, 3] or using natural language directly [1], to ease EUD in robotics.

In a survey on end-user robot programming, Ajaykumar et al. [1] identified three types of end-users based on expertise: general users, domain specialists, and robotics programmers. Most end-user robot programming systems focus on general users. However, many include more complex capabilities in the hope that general users may learn to leverage more robot capabilities [1]. The expert user of these systems is one with domain expertise and robotics knowledge. However, since these expert users are not the majority, many systems either try to support domain specialists in learning more robotics knowledge or support robotics experts in learning domain knowledge [1]. We present these user types mapped based on their expertise in Table 1.

User Type	Domain Expertise	Robotics/Tool Expertise
General User	Low	Low
Domain Expert	High	Low
Robotics Expert	Low	High
Expert User	High	High

Table 1: Mapping of users to relevant expertise.

Understanding users is necessary for designing robots. Research in HRI typically prioritizes a user-centric design approach that empathizes with users and aims to empower them. Researchers must also understand the broader impact of the robots they design in society and how robots may perpetuate existing inequities [8]. Service design methods provide a way to design the tasks that are invisible to the user and design with the politics, such as labor relationships and environmental impact, in mind [10]. Service design methodologies have been successfully used in robotics research to provide a big picture of how robots work [7, 11]. Service blueprinting is a service design method that looks beyond the user journey to define the touchpoints users have with others, and the necessary work that happens behind the scenes to meet users' needs [2]. Service blueprinting results in a service blueprint diagram that can serve as a single point of alignment on processes across an organization. When using service blueprints, designers typically produce a service blueprint that represents the current state of a service (if it already exists), identify weaknesses in the process, and design an ideal or target service blueprint.

While the majority of the time that users spend interacting in EUD systems is focused on authoring, EUD systems need to provide several additional capabilities upon which authoring depends. These capabilities include robot initialization/setup, program execution, editing and debugging, and verification [1]. Recent research in teleoperated socially assistive robotics suggests that tools sometimes overlook the importance of initialization or of connecting to a robot, and that general users often struggle with just getting started [5]. With teleoperated SARs used in therapy, authoring is often overlooked since the focus is on the robot's teleoperation, behavior editing, and verification [4, 6].

To define the invisible work upon which EUD in HRI depends, we apply service blueprinting to teleoperated Socially Assistive Robots used in therapy with children. We present our preliminary service blueprint for teleoperated SARs. We provide recommendations for researchers of EUD in HRI to consider and guidelines for how to modify our service blueprint to other domains of EUD in HRI.

2 DEFINING THE SERVICE BLUEPRINT

To define the service blueprint, we list the processes involved, typical user journeys, and various roles that are necessary to perform or support these journeys. After doing so, we connect all of these together to form the service blueprint.

2.1 Processes

As a first step to outlining a service blueprint, we enumerate the processes that occur across all user roles. We identify the following processes:

- **Authoring Setup**: any initial setup required before authoring can begin [1].
- **Authoring**: Robot behaviour definition/programming; the primary focus of EUD systems.
- **Verification**: verifying the authored program will work correctly [1].
- **Program Sharing**: the process of sharing defined programs with individuals who will execute them [5].
- **Execution Setup**: any initial setup required to execute authored programs [5].
- **Execution**: any tasks related to actively executing authored programs. Complexity may depend on the robot's level of autonomy.
- **Editing and Debugging**: program modification that may occur pre-execution, post-execution, or during execution [1].
- **Evaluation**: program review that occurs post-execution to evaluate the program [4].
- **System Development**: Building the EUD system itself.

2.2 User Journeys in Teleoperated Socially Assistive Robots

In this section, we define the typical user journeys that occur in controlling Socially Assistive Robots in education. A content developer begins by setting up their robot for authoring. The content developer defines any robot content expected to be delivered in the classroom. In the context of teleoperated Socially Assistive Robots, the dialogue that a robot would verbalize during an interaction represents the key *content* that must be authored to enable that interaction. Content developers verify the content is expressed as desired by practicing content on the connected robot. The content developer shares the authored content with the educator(s) who will use the robot. Educators begin by setting up their robots in the classroom. They review the content, test the content on their setup, and make edits where appropriate. The educator runs the program in the classroom and then reviews any analytics provided by the robotics system afterward. The educator may also share reporting with the content developer to inform improvements that can be made to the content.

2.3 Roles

Based on the processes and example user journey for teleoperated SARs, we identify the following roles:

- **Robot User or Assisted Individual**: directly interacts with the robot.
- **Robot Teleoperator or Caregiver**: sets up and operates the robot. The level of complexity of the tasks performed here will highly depend on the robot's level of autonomy.
- **Robot Programmer or Assistance Content Author**: develops content, program or behaviors to be performed by the robot.
- **EUD Tool Developer**: develops the EUD tool.

2.4 Mapping the Service Blueprint

Service blueprints typically focus on a single user and define the touchpoints, and tasks that are necessary to support the user's journey. In the case of teleoperated SARs, we found that delineating the various roles resulted in several interdependent user journeys that each require resources, tasks, and processes to support them. We initially developed three separate service blueprints, one for each role, to capture and present the full picture of teleoperated SARs. These service blueprints helped us determine the necessary high-level tasks and processes relative to each role. However, a service blueprint is typically used as a single document that can provide alignment across an organization around all that is necessary to provide a service. Requiring separate role-specific documents thus fails to meet that goal.

To arrive at a single document that captures all the roles and each role's journey, necessary tasks, and requirements, we extend service blueprints to simultaneously show multiple levels and journeys. Our extension to service blueprints is inspired by work in multi-level service design [9] and multi-actor service blueprints [12]. Service blueprints include a user journey, a set of front-stage touchpoints visible to the user, and a set of backstage processes invisible to the user. When incorporating multiple roles, visibility becomes role-specific. For example, content sharing between the content developer and the robot teleoperator is invisible to the robot user but visible to both the teleoperator and content developer. When applying the service blueprint to a particular EUD tool, it is important to highlight where the lines of interaction and lines of visibility exist from the perspective of each user. We discuss this in more detail in Section 3.

We present our preliminary service blueprint in Figure 1. Note that each role has a "Physical Evidence" row that defines the role's environment and an "Actions" row that defines the role's actions as performed in order. The bottom row lists the support services necessary to perform any actions that occur at the same time.

3 DISCUSSION

In defining a service blueprint for teleoperated Socially Assistive robots, we outlined the processes that occur, the roles that individuals play, the interactions across roles, and the necessary capabilities to support teleoperated SARs. We believe that this service blueprint can be used to inform EUD tool design in robotics in general. We suspect that EUD tool designers would benefit from copying our service blueprint and defining or highlighting the following:

Roles & Necessary Skill Sets. For each EUD tool, the skill set necessary to perform each process may vary. Designers may benefit from outlining whether roles are expected to be performed by the same or different individuals. For each process, it would also be

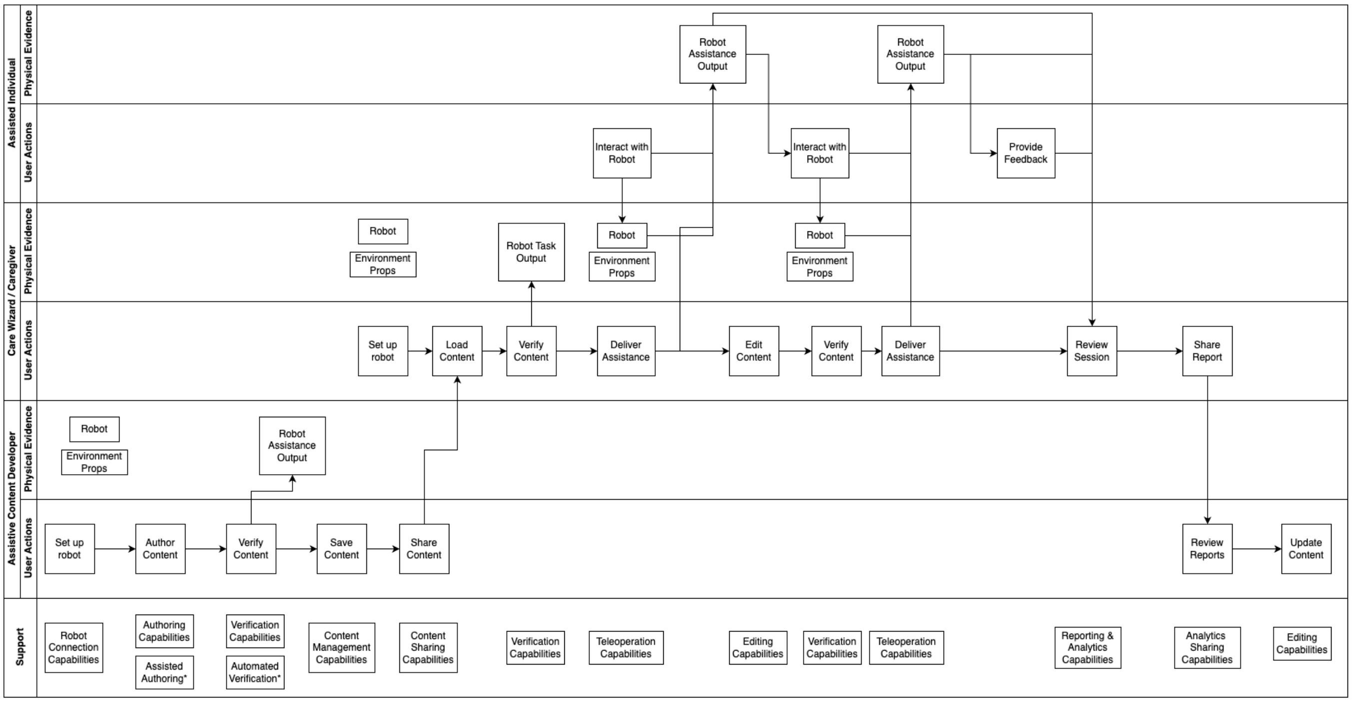

Figure 1: The service blueprint of teleoperated SARs

beneficial to outline the target expertise level and skill sets required to inform the design of EUD tool capabilities for that task.

Lines of Interaction & Visibility. For each EUD tool, designers may want to further define where the lines of separation for interaction and visibility would occur. In cases where robots are intended to be perceived as autonomous despite being controlled by an individual, robot users may not interact with the robot teleoperator. In some domains, robot teleoperators may not interact with robot program authors. We have intentionally not included the lines of interaction and visibility so that designers can define those for their applications.

Feature Definition. Designers may benefit from outlining which features or capabilities their tools provide and to whom. The following capabilities may be optional in many EUD tools: assisted authoring, automated verification, program or analytics sharing (if the author and teleoperator are the same individual), and analytics and reporting capabilities. In other cases, it may be an intentional choice, for example, to remove editing capabilities from being accessible to robot teleoperators.

Domain Specific Programs. The service blueprint would likely vary greatly depending on the particular domains supported by each EUD tool. Designers ought to expand on the robot user's experience and user journey based on the EUD tool capabilities. This may lead to further domain-specific features that the EUD tool ought to support.

4 CONCLUSION

End-user development in robotics can empower users to benefit from robots with minimal technical knowledge and/or burden. In addition to designing with an understanding of the user journey, we applied service blueprinting to uncover the various resources and tasks that occur behind the scenes to support the use journey. We applied service blueprinting to a use case of teleoperated SARs to inform the development of robot EUD tools in general. We provide a preliminary service blueprint that outlines user journeys, roles, and necessary EUD tool capabilities to support them. We recommend EUD tool designers apply and expand on our service blueprint, and outlined important steps for doing so.

REFERENCES

[1] Gopika Ajaykumar, Maureen Steele, and Chien-Ming Huang. 2021. A Survey on End-User Robot Programming. *ACM Comput. Surv.* 54, 8, Article 164 (oct 2021), 36 pages. https://doi.org/10.1145/3466819

[2] Mary Jo Bitner, Amy L Ostrom, and Felicia N Morgan. 2008. Service blueprinting: a practical technique for service innovation. *California management review* 50, 3 (2008), 66–94.

[3] Enrique Coronado, Fulvio Mastrogiovanni, Bipin Indurkhya, and Gentiane Venture. 2020. Visual Programming Environments for End-User Development of intelligent and social robots, a systematic review. *Journal of Computer Languages* 58 (2020), 100970. https://doi.org/10.1016/j.cola.2020.100970

[4] Saad Elbeleidy, Terran Mott, Dan Liu, Ellen Do, Elizabeth Reddy, and Tom Williams. 2023. Beyond the Session: Centering Teleoperators in Robot-Assisted Therapy Reveals the Bigger Picture. In *Proceedings of the ACM Conference On Computer-Supported Cooperative Work And Social Computing.*

[5] Saad Elbeleidy, Terran Mott, Dan Liu, and Tom Williams. 2022. Practical Considerations for Deploying Robot Teleoperation in Therapy and Telehealth. In *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN).* IEEE, 977–984.

[6] Saad Elbeleidy, Elizabeth Reddy, and Tom Williams. 2023. The Invisible Labor of Authoring Dialogue for Teleoperated Socially Assistive Robots. In *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN).*

[7] Min Kyung Lee and Jodi Forlizzi. 2009. Designing adaptive robotic services. *Proc. of IASDR'09* (2009), 1–10.

[8] Anastasia K. Ostrowski, Raechel Walker, Madhurima Das, Maria Yang, Cynthia Breazea, Hae Won Park, and Aditi Verma. 2022. Ethics, Equity, Justice in Human-Robot Interaction: A Review and Future Directions. In *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. 969–976. https://doi.org/10.1109/RO-MAN53752.2022.9900805

[9] Lia Patrício, Raymond P Fisk, João Falcão e Cunha, and Larry Constantine. 2011. Multilevel service design: from customer value constellation to service experience blueprinting. *Journal of service Research* 14, 2 (2011), 180–200.

[10] Lara Penin. 2018. *An introduction to service design: designing the invisible*. Bloomsbury Publishing.

[11] Kseniya Tarasova. 2015. Service design in social robotics: Involving elderly care professionals into co-creation. (2015).

[12] Thomas Wreiner, Ingrid Mårtensson, Olof Arnell, Natalia Gonzalez, Stefan Holmlid, and Fabian Segelström. 2009. Exploring service blueprints for multiple actors: A case study of car parking services. In *First nordic conference on service design and service innovation*. 213–223.