PHUI-kit: A Tool for Physical User Interface Layout

Michael D. Jones
Zann Anderson
Casey Walkder
Kevin Seppi
Brigham Young U.
Provo, UT, USA
m@byu.edu
zannderson@gmail.com
caseywalker117@gmail.com
k@byu.edu

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Copyright held by the owner/author(s).
CHI'18 Extended Abstracts, April 21–26, 2018, Montreal, QC, Canada
ACM 978-1-4503-5621-3/18/04.
<https://doi.org/10.1145/3170427.3186518>

Abstract

PHUIs are interfaces built from tangible widgets arranged on the surfaces of physical objects. PHUI-kit is tool for making physical user interface (PHUI) design more like graphical user interface (GUI) design in the context of 3D printed object housings. The tool includes a drag-and drop interface to place, reposition and delete physical widgets on a curved housing in a 3D modeling tool. Automatic cable routing simplifies assembly planning and execution. The objects are printed, the widgets snapped onto a cable and snapped into the housing to create an interactive object. A cable guide printed on paper guides the maker while snapping widgets onto the cable.

Author Keywords

tangible interfaces; user interface layout tools

ACM Classification Keywords

H.5.m [Information interfaces and presentation (e.g., HCI)]:
Miscellaneous

Introduction

Graphical user interface (GUI) layout tools allow a person to create a GUI by clicking an interface element in a menu and then dropping the element into the layout. This paradigm appeared as early as 1983 [2] and has carried through to modern integrated development environments

(IDEs) in which a developer can click on interface elements, drag them onto a window, and place them in an interface. This approach to GUI layout can be found in AndroidStudio¹, WindowBuilder for Eclipse² and the XCode Interface Builder³.

Inspired by drag-and-drop GUI editors for screens, we seek to create a similar interface layout experience for physical interactive devices. We call an interface for a physical device a “physical user interface” or “PHUI” (pronounced “foey”) from physical user interface by analogy to “GUI” for graphical user interfaces.

In this demonstration, participants will use PHUI-kit to layout interfaces on curved housings. Our demonstration encourages reflection of the role of perspective in PHUI layout in that PHUI-kit allows makers to place widgets on the object exterior rather than placing components on the object interior.

PHUI-kit

PHUI-kit which is a palette-based interface for PHUI layout on digital models of curved 3D objects. PHUI-kit includes a plugin for the Rhinoceros 5 modeling tool and a set of small custom-built electronic components connected by a flat ribbon cable.

Placing Widgets

The PHUI-kit interface is shown in Figure 1. To place a widget, the maker selects a widget from the toolbar, selects an object on which to place the widget, and then places the widget on the outside of the object where it should appear in the user interface. The widget is constrained to lie on and be oriented to the housing surface during placement.

¹<https://developer.android.com/studio/index.html>

²<https://eclipse.org/windowbuilder/>

³<https://developer.apple.com/xcode/>

Figure 1: To place a widget, a maker clicks the widget name on the left and drags it onto the object surface on the right—in much the same process one might use to layout a graphical user interface. The tool hides implementation details associated with building physical user interfaces.

After the maker selects an object to edit, PHUI-kit embeds the widget to lie on and be perpendicular to the surface of the object. Constraining the widget to lie on the enclosure surface is a simple task but it is a critically important part of the PHUI layout experience.

The widget is partially rendered during placement to allow the maker to visualize where the widget will go. The resulting process is smooth and fluid.

When the maker clicks the surface, the widget is placed at that point on the object by translating and rotating the model loaded from the server. The maker believes they are placing a widget, but PHUI-kit actually places the geometry needed to modify the housing to support implementing a widget at the selected location.

After placement, we check if the widget collides with other

widgets, or if the placement is infeasible because it lies partially outside the enclosure. If so, the widget is colored red to indicate that the placement is infeasible.

For widgets with exposed moving handles, such as knobs, the maker also sets the diameter and height of the handle bounding volume. That bounding volume is used later to trim bumps in the enclosure surface which might impede movement.

Widgets with caps will need mounting holes and posts added to the cap so that the cap snaps onto the component. A maker creates these holes and posts by clicking the “cap” button for that widget type and then clicking the geometry to be used as a cap. The cap geometry is then placed as just described.

When the design is complete, the maker clicks print prep, instructing PHUI-kit to prepare the housing for 3D printing and to route the ribbon cable between widgets.

PHUI-kit generates a cable wiring diagram based on the generated cable route. The maker lays the ribbon cable on a paper printed copy of the wiring diagram and crimps headers at the appropriate lengths. Widgets snap into headers and then snap into their brackets in the housing.

Moving and Deleting Widgets

Another key feature of GUI design tools is that they allow the maker to re-position and delete interface elements. This is difficult in PHUI layout tools based on sculpting because re-positioning widgets involves moving clay [4] or reprocessing scanned sculptures [6].

However, moving widgets in virtual representations of the interface is simple. To move a widget in PHUI-kit, the maker clicks the “move” command, selects a widget, selects the

object on which to place the widget and then places the widget as described earlier. Deleting a widget is similar except the widget is not re-positioned.

Related Work

Others have explored the placement of electrical components in enclosures for personal fabrication of interactive devices but primarily from the internal perspective. That is from the the *inside-out* rather than the *outside-in*. For example, Villar et al. [7] allow a maker to place a *component in the enclosure interior*. We support the maker placing a *widget on the enclosure exterior*.

Our demonstration encourages reflection of the role of perspective in PHUI layout. From the maker’s perspective, we believe the difference between working on the exterior of the device rather than the inside is significant because ultimately the placement of widgets on the outside is what matters in interface layout.

Weichel et al. [8] also work from the outside of the enclosure, but only for objects comprised of flat laser-cut pieces. This work is important in that it automatically rearranges widgets when enclosure faces are resized, but it does not address problems related to working with enclosures made of curved surfaces using a 3D printer.

Our work is closely related to prior work in prototyping interfaces for physical interactive devices. BOXES [3], Switcharoo [1] and Calder [5] merge form and functionality early in the prototyping process so that a maker can build an interactive physical device with a certain shape. DisplayObjects [1] also merge form and function but do so by projecting images of interactive elements onto a foam model in a motion capture arena. We seek a similar early merger of form and functionality but at a higher-level of fidelity and expressiveness while seeking to maintain fluidity. 3D printing allows a

maker to easily create nearly any physical shape, but the details associated with adding interactive elements to the 3D models before printing remains tedious and we seek to remove that tedium.

Acknowledgement

This work supported by National Science Foundation grant IIS-1406578.

REFERENCES

1. Daniel Avrahami and Scott E. Hudson. 2002. Forming Interactivity: A Tool for Rapid Prototyping of Physical Interactive Products. In *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '02)*. ACM, New York, NY, USA, 141–146. DOI: <http://dx.doi.org/10.1145/778712.778735>
2. W. Buxton, M. R. Lamb, D. Sherman, and K. C. Smith. 1983. Towards a Comprehensive User Interface Management System. In *Proceedings of the 10th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '83)*. ACM, New York, NY, USA, 35–42. DOI: <http://dx.doi.org/10.1145/800059.801130>
3. Scott E. Hudson and Jennifer Mankoff. 2006. Rapid Construction of Functioning Physical Interfaces from Cardboard, Thumbtacks, Tin Foil and Masking Tape. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST '06)*. ACM, New York, NY, USA, 289–298. DOI: <http://dx.doi.org/10.1145/1166253.1166299>
4. Michael D. Jones, Kevin Seppi, and Dan R. Olsen. 2016. What You Sculpt is What You Get: Modeling Physical Interactive Devices with Clay and 3D Printed Widgets. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 876–886. DOI: <http://dx.doi.org/10.1145/2858036.2858493>
5. Johnny C. Lee, Daniel Avrahami, Scott E. Hudson, Jodi Forlizzi, Paul H. Dietz, and Darren Leigh. 2004. The Calder Toolkit: Wired and Wireless Components for Rapidly Prototyping Interactive Devices. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '04)*. ACM, New York, NY, USA, 167–175. DOI: <http://dx.doi.org/10.1145/1013115.1013139>
6. Valkyrie Savage, Sean Follmer, Jingyi Li, and Björn Hartmann. 2015. Makers' Marks: Physical Markup for Designing and Fabricating Functional Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 103–108.
7. Nicolas Villar, James Scott, Steve Hodges, Kerry Hammil, and Colin Miller. 2012. .NET Gadgeteer: A Platform for Custom Devices. In *Proceedings of the 10th International Conference on Pervasive Computing (Pervasive'12)*. Springer-Verlag, Berlin, Heidelberg, 216–233. DOI: http://dx.doi.org/10.1007/978-3-642-31205-2_14
8. Christian Weichel, Manfred Lau, and Hans Gellersen. 2013. Enclosed: A Component-centric Interface for Designing Prototype Enclosures. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13)*. ACM, New York, NY, USA, 215–218. DOI: <http://dx.doi.org/10.1145/2460625.2460659>