17th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceava, Romania, May 23-25, 2024

From Smart Buildings to Smart Vehicles: Mobile User Interfaces for Multi-Environment Interactions

Laura-Bianca BILIUS, Alexandru-Tudor ANDREI, Radu-Daniel VATAVU
MintViz Lab, MANSiD Research Center, Stefan cel Mare University of Suceava, Romania
laura.bilius@usm.ro, tudor.andrei@usm.ro, radu.vatavu@usm.ro

Abstract—**We introduce the concept of multi-environment interactions in the context of smart cities, spanning smart environments of diverse kinds, from smart buildings to smart vehicles. The primary goal of multi-environment interactions is effective control of computer systems in those environments in a manner that is uniform and consistent to their users. In our approach, users interact with and control devices and services from smart environments through one application running on their personal mobile devices. We present the engineering details of our mobile application and discuss its multi-environment orientation through the prism of design heuristics relevant for the user experience of ubiquitous computing applied to smart cities. Furthermore, to demonstrate the utility of multi-environment interactions in this context, we present two practical use cases involving smart buildings and smart vehicles, respectively, where users control ambient light, sound, and airflow.**

Keywords—smart buildings, smart vehicles, smart cities, smart mobility, multi-environment interactions, ubiquitous computing, ambient intelligence

I. INTRODUCTION

Smart buildings and smart mobility represent two key parts of the concept of a smart city, where the constituting elements, such as smart environments, smart living, smart economy, and smart governance, come together to create a technologically advanced and sustainable urban environment [1]. These elements are supported by a diversity of digital devices and ubiquitous computing infrastructure, tailored to meet specific needs and deliver specific user experiences [2], [3]. However, while smart devices and environments, such as smart home appliances and smart vehicles, are continuously evolving in terms of the features and services offered to their users, technical challenges arise related to their inter-connectivity and, thus, the complexity of controlling and interacting with them uniformly [4]. In consideration of such challenges, a potential technical solution is represented by the design of mobile user interfaces that streamline control of smart devices across multiple, different environments.

Considering the ubiquitous integration of smart systems, devices, and sensors across physical environments, we introduce in this paper the concept of multi-environment interactions, which we demonstrate with a mobile application that enables users to control devices, from smart buildings to smart vehicles. For example, users can customize lighting and airflow settings in one environment, which can be subsequently transferred to other environments, such as the smart vehicle. In this paper, we present the engineering details of our mobile application and illustrate practical use cases involving smart ambient light, sound, and airflow. Furthermore, to characterize the multi-environment orientation of our application, we adopt design heuristics of the user experience in ubiquitous computing environments [5].

II. RELATED WORK

We overview prior work in the area of ubiquitous computing with a focus on smart buildings and smart mobility [6]-[15]. We specifically focus on these two smart city components due to the increasing prevalence of connected devices, ranging from smart home appliances to smart automotive interactive technology, that render these environments "smart."

In the area of interactions with smart appliances and digital devices in smart homes and smart buildings, prior studies have explored various interaction techniques [12]-[15]. For example, Ahmadi-Karvigh et al. [13] presented individual differences that influence users' preferences for the control of smart lighting systems. Their findings indicated that income and education levels, but also personality traits may affect users' preferences for particular automation levels in a smart environment. Other studies focused on specific user groups. For example, Contreras-Castaneda et al. [14] introduced a multimodal system designed to facilitate interaction with smart home devices and appliances for users with limited mobility. Their system was based on a smartphone application featuring various input modalities, including touch, voice, and a brain-computer interface. Ungurean and Vatavu [16] conducted an ability-centered examination of smart interactive television, and revealed accessibility challenges and potential technical solutions for assistive technology addressing users with upper-body motor impairments towards more accessible smart home entertainment environments.

Related to smart mobility, the scope of our work includes interactions with smart vehicles. Previous research in this space has explored a variety of techniques designed for both inside and outside-the-vehicle interaction, including conventional car controls, touchscreens, mid-air gestures, voice commands, and multimodal input [6]-[9], enabling both drivers and passengers [10] to control light, sound, and airflow. For example, Gheran and Vatavu [17] proposed gesture input performed with a smart ring as an alternative to steering wheel controls, while voice input was proposed by Alvarez et al. [18] integrated into a conversational user interface. These scientific and technical contributions can be addressed uniformly from the perspective of a smart vehicle representing a specific kind of smart environment [19]. We refer readers to [3] for an overview of input modalities for in-vehicle consumption of interactive media.

This work is supported by the NetZeRoCities Competence Center, funded by European Union – NextGenerationEU and Romanian Government, under the National Recovery and Resilience Plan for Romania, contract no.760007/30.12.2022 with the Romanian Ministry of Research, Innovation and Digitalisation through the specific research projects P3, Smart and sustainable buildings and P4, Smart mobility and infrastructure.

III. Design Requirements for Mobile Applications and Multi-Environment Interactions

We introduce in this work the concept of *multi-environment interaction*, empowering users to control smart devices in different environments, from smart buildings to smart vehicles, through the same mobile application. In this context, multi-environment interaction aligns to the set of Poslad's [5] user experience design heuristics (DH$_1$ to DH$_{17}$) for ubiquitous computing, as follows:

DH$_1$ *Safe exploration*. This heuristic specifies that applications enable users to explore the interface without the risk of making mistakes. In this line, multi-environment interactions enable users to experiment with different settings (*e.g.*, light intensity, sound volume, etc.) as they transition between different environments and can undo any changes if needed.

DH$_2$ *Satisficing*, *i.e.*, searching for adequate interaction options rather than aiming for the best possible choice. According to this heuristic, design for multi-environment interactions prioritizes clarity in the user interface.

DH$_3$ *Changes in midstream* assume that users may change their decisions or actions during the interaction. Thus, the design of multi-environment interactions should enable dynamical adjustment of device output based on previously configuration of user preferences.

DH$_4$ *Deferred choices* depict the situation where users may opt to postpone making certain decisions within the application. For example, the design of multi-environment interaction features turning on/off output devices through one user interface and mobile application.

DH$_5$ *Incremental construction*. According to this heuristic, tasks or processes evolve and require repeated attempts to get right. In the case of multi-environment interactions, this heuristic is reflected in the automatic adjustment of settings for output devices based on users' preferences for input across various smart environments.

DH$_6$ *Habituation or unification* means that users could make incorrect choices because of varying input modalities and techniques available across various applications. Thus, multi-environment interactions are performed with the user's preferred mobile devices through one user interface.

DH$_7$ *Constituted actions*. Within a specific context, lower-level actions combine to form a higher-level one. Following this heuristic, user preferences that propagate across multiple smart environments contribute to higher-level configuration of the corresponding interactions.

DH$_8$ *UI proxy*. Users might need to employ several individual devices and interfaces to achieve their goal. However, the UI design for multi-environment interaction facilitates control for multiple output devices using the preferred input device or interface (*e.g.*, tablet, smartphone, laptop, smartwatch, smart glasses).

DH$_9$ *Context-based and memory-recall*. This design heuristic specifies that users interact with objects based on the context of when and where they previously used them. Thus, the design of multi-environment interactions leverages predictable contexts due to settings transfer, resulting in a uniform user experience across various environments.

DH$_{10}$ *Prospective memory and context-aware trigger*. According to this heuristic, users tag objects to remind themselves to address later. Multi-environment interactions could enable access to the history of environments users have interacted with and reuse previous settings. Moreover, users can control devices from one smart environment while being physically located within another.

DH$_{11}$ *Situated help*, *i.e.*, even though help access is customized, it may not always align with users' understanding or experiences. Thus, multi-environment interactions should feature an intuitive mobile interface, easily understandable and informative for users at all levels.

DH$_{12}$ *Instant feedback*. The absence of feedback may lead users to interrupt their tasks or provide redundant input. In this line, user interface design for multi-environment interactions should enable real-time confirmation of user action across different environments, *e.g.*, users can easily observe the effects on various output devices.

DH$_{13}$ *Context-driven and explanations*. The reasons for unavailable options in the user interface or malfunctioning due to this unavailability are often unclear. Thus, multi-environment interactions must ensure minimization of potential user confusion by guaranteeing interface accessibility. This desideratum can be achieved by enabling control from the user's personal mobile device.

DH$_{14}$ *Prospective, anticipated actions*. According to this heuristic, as a task progresses, it becomes clearer which subsequent tasks are necessary to achieve the goal. Consequently, the design of multi-environment interactions should align and foster intuitiveness by making it clear which device will respond as user input progresses.

DH$_{15}$ *Streamlined replay* aims at improving user productivity by detecting redundant actions and simplifying the replay process. According to this design heuristic, multi-environment interactions are to be used across environments with different characteristics, facilitating streamlined replay of user action through the same mobile application instead of switching to other interfaces.

DH$_{16}$ *Streamlined input*. Following this heuristic, multi-environment interactions afford streamlined input by leveraging users' personal devices.

DH$_{17}$ *Streamlined output*. Multi-environment interactions streamline output by providing immediate feedback about the controlled device or service in a smart environment.

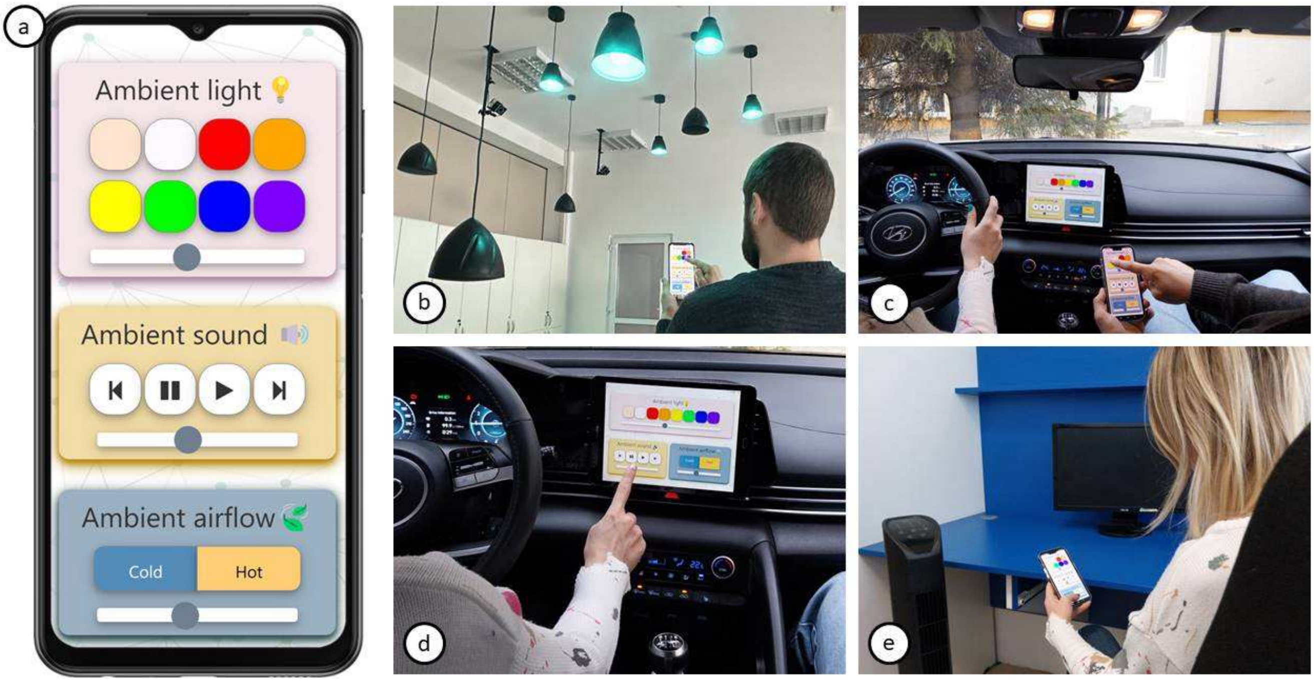

Fig. 1 The mobile application developed to demonstrate multi-environment interactions (a), and photographs that illustrate practical use case scenarios with users transitioning between several indoor and outdoor smart environments (b-e).

IV. Mobile Application and User Interface for Multi-Environment Interactions

To demonstrate our concept of multi-environment interactions, we designed and developed a mobile application and two practical use cases involving smart environments represented by smart buildings and smart vehicles, respectively. In this context, the relationship connecting smart buildings and smart vehicles is represented by users transitioning these environments, within which they have access to and control various functionalities, *i.e.*, ambient light, sound, and airflow. Fig. 1a depicts the user interface of our mobile application featuring distinct control sections corresponding to these output devices. The one-screen user interface was designed to deliver an intuitive user experience fostering safe exploration, habituation, prospective memory, and streamlined input/output, according to the design heuristics of ubiquitous computing [5] applied to multi-environment interactions, which were discussed in the previous section.

A. Engineering Details

We designed the application for a mobility context of use and portability across various mobile platforms and devices. Facilitated through a Flask server application, developed using Python, and a client application based on Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript, and the Bootstrap front-end framework, the user interface of the mobile application dynamically adapts to varying device characteristics. The user interface is divided into three sections corresponding to the type of output devices that can be controlled: ambient light, ambient sound, and ambient airflow. For example, users can configure the color and brightness level for the ambient light, if the environment permits such a feature.

In the case of ambient sound, users have the option of adjusting audio volume and play, pause, and resume sound. Ambient airflow is controlled in terms of the temperature setting and airflow intensity. When the user performs an action in the application, an HTTP request is sent to the server. Listing 1 exemplifies JavaScript code implementing an event listener for controlling the airflow intensity provided by a smart tower fan.

```
req(cmd, value) {
 let actual_value = this.transform(cmd,value);
 return {
  method: 'PUT',
  headers: {'Content-Type':'application/json',
   'Accept': 'application/json'},
  body: JSON.stringify({command: cmd,
        value: actual_value}) };
}
send_command(command, value) {
 let request = this.req(command, value);
 fetch(this.URL, request).then(response =>
       console.log(response.json()) );
}
fanRangeIntensity.addEventListener("change",
   () => { fan.send_command(Fan.COMMANDS.GEAR,
   fanRangeIntensity.value); });
```

Listing 1. Javascript code creating a JSON object with the settings for controlling the airflow provided by a smart tower fan.

The server parses the HTTP request received from the client application to extract the command and its associated parameters. Subsequently, the server creates a new request, following the output device API specification, *e.g.*, the Govee API [20] in our example involving a smart fan; see Listing 2.

```
class Fan:
  def command(self, cmd, value):
    headers ={"Govee-API-Key":API_KEY,
      "Content-Type": "application/json"}
    content = {"device": DEVICE_MAC,
      "model": MODEL,
      "cmd": {"name": cmd, "value": value},}
    requests.put(URL, headers=headers,
                 json=content)
```

Listing 2. Python code creating a HTTP request to control an output device in the smart environment, in this case, a smart tower fan.

B. *Preliminary Evaluation Results*

To validate our technical implementation, we deployed the mobile application on two mobile devices (Samsung A23, Dual-core 2.2 GHz Kryo 660 Gold & Hexa-core 1.7 GHz Kryo 660 Silver CPU and 6GB RAM, running Android 14, and iPhone 14 Pro, Dual-core 3.46 GHz Everest & Quad-core 2.02 GHz Sawtooth CPU and 6GB RAM, running iOS 17.4) in two environments represented by a smart room and inside a vehicle, respectively. In the smart room scenario, we utilized Philips Hue A60 smart lamps to demonstrate control of ambient light, Power Dynamics PDS40W speakers for ambient sound, and a Govee Smart Tower Fan H7101 for ambient airflow. In the in-vehicle environment, we installed a tablet device running our mobile application; see Fig. 1b-e for several photographs taken during our testing. As users transition between different environments, the application maintains connectivity with the output devices via Bluetooth and the server via Wi-Fi/HTTP. For example, the user connects to the ambient sound system in the smart room (Fig. 1b) and adjusts the light color to cyan. When inside the vehicle (Fig. 1c), the same application connects to the in-vehicle infotainment system (our tablet device during the testing) to set the background light to the same color. A second user employs the mobile application to navigate through the music list inside the vehicle (Fig. 1d). When in the smart building, the same mobile application enables control of the airflow (Fig. 1e), according to our example from the previous subsection.

V. CONCLUSION

We introduced multi-environment interactions enabled by applications running on personal devices as users transition between different smart environments. To this end, we employed user-centered design heuristics for ubiquitous computing and designed and developed a mobile application that can be used to control ambient light, sound, and airflow in two environments represented by a smart room and smart vehicle. Our work has several limitations. First, our demonstrator is limited to the specific smart devices employed during testing, and further work is needed to extend functionality to other devices. Second, since our goal was to demonstrate the concept of multi-environment interaction, we adopted a simplified design and engineering approach. Future work will consider dedicated software architecture for smart environments, enabling our application to run on other personal devices, such as smartwatches and smart glasses. Also, future work will consist of conducting a user study to evaluate various dimensions of the user experience across transitions between different smart environments.

REFERENCES

[1] P. Saputra and A. Ramadhan, "*Smart building trend, role, and position: a systematic literature review*," International Journal of Reconfigurable and Embedded Systems (IJRES), 12, pp. 29–41, 2023.

[2] L. Alghamdi, M. Akter, C. S. Cardenas, D. A. Cruces, J. Wiese, J. Kropczynski, H. Lipford, and P. J. Wisniewski, "*Mi casa es su casa ("misu"): A mobile app for sharing smart home devices with people outside the home*," in Companion Publication of the 2022 Conference on Computer Supported Cooperative Work and Social Computing, CSCW'22 Companion, p. 184–187, ACM, 2022.

[3] L.-B. Bilius and R.-D. Vatavu, "*A synopsis of input modalities for in-vehicle infotainment and consumption of interactive media,*" in Proceedings of the 2020 ACM International Conference on Interactive Media Experiences, IMX '20, p. 195–199, ACM, 2020.

[4] D. Wang, K. Sugiura, and Y. Murase, "*Design and implementation of user-centered home appliance controlling service environment*," in Proceedings of Workshop on Mobile Video Delivery, ACM, 2018. http://dx.doi.org/10.1145/2579465.2579473

[5] S. Poslad, *Ubiquitous Computing: Smart Devices, Environments and Interactions*. United Kingdom: John Wiley & Sons, 2009.

[6] L. Angelini, F. Carrino, S. Carrino, M. Caon, O. A. Khaled, J. Baumgartner, A. Sonderegger, D. Lalanne, and E. Mugellini, "*Gesturing on the steering wheel: a user-elicited taxonomy,*" in Proc. of the 6th Int. Conf. on Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI '14, p. 1–8, ACM, 2014.

[7] L. Angelini, F. Carrino, S. Carrino, M. Caon, D. Lalanne, O. A. Khaled, and E. Mugellini, "*Opportunistic synergy: a classifier fusion engine for micro-gesture recognition,*" in Proceedings of the 5th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI '13, p. 30–37, ACM, 2013.

[8] K. R. May, T. M. Gable, and B. N. Walker, "*Designing an in-vehicle air gesture set using elicitation methods*," in Proc. 9th International Conference Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI '17, p. 74–83, ACM, 2017. https://doi.org/10.1145/3122986.3123015

[9] G. Shakeri, J. H. Williamson, and S. Brewster, "*May the force be with you: Ultrasound haptic feedback for mid-air gesture interaction in cars,*" in Proc. of the 10th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, ACM, 2018. https://doi.org/10.1145/3239060.3239081

[10] L.-B. Bilius, R.-D. Vatavu, and N. Marquardt, "*Smart vehicle proxemics: A conceptual framework operationalizing proxemics in the context of outside-the-vehicle interactions*," in Human-Computer Interaction – INTERACT 2021, (Cham), pp. 150–171, Springer, 2021.

[11] M. T. Tombeng, R. Najoan, and N. Karel, "*Smart car: Digital controlling system using android smartwatch voice recognition,*" in 2018 6th Int. Conf. on Cyber and IT Service Management, p:1–5, 2018.

[12] A.-T. Andrei, A.-I. Siean, and R.-D. Vatavu, "*Tap4light: Smart lighting interactions by tapping with a five-finger augmentation device,*" in Proceedings of the 13th Augmented Human International Conference, AH2022, ACM, 2022.

[13] S. Ahmadi-Karvigh, A. Ghahramani, B. Becerik-Gerber, and L. Soibelman, "*One size does not fit all: Understanding user preferences for building automation systems,*" Energy and Buildings, 145, 2017.

[14] M. A. Contreras-Castaneda, J. A. Holgado-Terriza, G. Pomboza-Junez, P. Paderewski-Rodriguez, and F. L. Gutierrez-Vela, "*Smart home: Multimodal interaction for control of home devices,*" in Proceedings of the XX International Conference on Human-Computer Interaction, Interaction '19, ACM, 2019.

[15] N. Almeida, S. Silva, A. Teixeira, M. Ketsmur, D. Guimaraes, and E. Fonseca, "*Multimodal interaction for accessible smart homes,*" in Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-Exclusion, p. 63–70, ACM, 2018. https://doi.org/10.1145/3218585.3218595

[16] O.-C. Ungurean and R.-D. Vatavu, "*Ability-centered examination of people with motor impairments' interaction with television towards more accessible smart home entertainment environments,*" in Proceedings of the 13th International Symposium on Ambient Intelligence, pp. 32–43, 2022.

[17] B.-F. Gheran and R.-D. Vatavu, "*From controls on the steering wheel to controls on the finger: Using smart rings for in-vehicle interactions,*" DIS' 20 Companion, p. 299–304, ACM, 2020.

[18] I. Alvarez, A. Martin, J. Dunbar, J. Taiber, D.-M. Wilson, and J. E. Gilbert, "*Voice interfaced vehicle user help,*" in Proc. of the 2nd Int. Conference on Automotive User Interfaces and Interactive Vehicular Applications, AutomotiveUI '10, p. 42–49, ACM, 2010.

[19] O.-A. Schipor and R.-D. Vatavu, "*Empirical results for high-definition video and augmented reality content delivery in hyper-connected cars,*" Interacting with Computers, vol. 33, no. 1, pp. 3–16, 2021, https://doi.org/10.1093/iwcomp/iwaa025

[20] Govee Developer API, https://govee-public.s3.amazonaws.com/developer-docs/GoveeDeveloperAPIReference.pdf