

UNIVERSITY OF CRETE  
DEPARTMENT OF COMPUTER SCIENCE  
FACULTY OF SCIENCES AND ENGINEERING

**Wizard of Aml: A system for building and  
enacting interactive prototypes in Intelligent  
Environments**

by

Dimitrios Arampatzis

MASTER'S THESIS

Heraklion, December 2020



UNIVERSITY OF CRETE  
DEPARTMENT OF COMPUTER SCIENCE

**Wizard of AmI: A system for building and enacting interactive  
prototypes in Intelligent Environments**

by **Dimitrios Arampatzis**

A thesis submitted in partial fulfillment of the  
requirements for the degree of Master of Science


**APPROVED BY:**



---

**Author:** Dimitrios Arampatzis

CONSTANTINOS  
STEPHANIDIS



Digitally signed by CONSTANTINOS  
STEPHANIDIS  
Date: 2021.01.22 13:44:42 +02'00'

---

**Supervisor:** Constantine Stephanidis, Professor, University of Crete




Digitally signed by ANTONIOS  
ARGYROS  
Date: 2021.01.22 15:04:14  
+02'00'

---

**Committee Member:** Antonis Argyros, Professor, University of Crete

Margherita Giovanna  
Antona




Digitally signed by Margherita  
Giovanna Antona  
Date: 2021.01.22 13:57:53 +02'00'

---

**Committee Member:** Margherita Antona, Principal Researcher, FORTH-ICS

Polyvios  
Pratikakis



Digitally signed by Polyvios  
Pratikakis  
Date: 2021.01.25 12:52:11  
+02'00'

---

**Director of Graduate Studies:** Polyvios Pratikakis, Assistant Professor, University of Crete



This thesis is dedicated to my family...  
to my father Giorgos, my mother Avgi,  
and my sister Asimina  
... who have always supported me



# Acknowledgments

First of all, I would like to thank my thesis supervisor, Professor Constatine Stephanidis, as he was the person who inspired me to work in the field of Human-Computer Interaction. In addition, I would like to thank him for giving me the opportunity to be a member of the HCI laboratory, and for his support throughout my Master studies. I am also grateful to my advisory committee members, Professor Antonis Argyros and Principal researcher Dr. Margherita Antona, who helped me improve my work with their valuable feedback.

Moreover, I owe my appreciation to my colleagues Dr. Asterios Leonidis, Dr. Maria Korozi, and Mr. Vasilis Kouroumalis for the valuable help during my master studies which contributed to the end result of this thesis. I would also like to thank all and each one of my colleagues, who, during the design and development process of the system, contributed either through a workshop, evaluation, or conversations with me.

My best thanks to my friends Katerina Valakoy, Vasilis Glabedakis, Maria Doulgeraki, Giorgos Kalaitzidis, Nikos Liapakis, Mike Loukakis, Nantia Manoli, Alex Mavromatis, Stavros Batzakas, George Nikitakis, David Parastatidis, Vagelis Poutouris, Irini Sykianaki, Evi Kamporoudi and Annie Tran who, in their own way, supported me during this thesis and transformed this period to a more fun and enjoyable experience.

Finally, I am grateful to my parents, George and Avgi, for the unwavering support they have given to me throughout my studies, who believe in me and help me whenever I need them without any doubt.





# Ευχαριστίες

Καταρχάς, θα ήθελα να ευχαριστήσω τον επόπτη της μεταπτυχιακής μου εργασίας, καθηγητή Κωνσταντίνο Στεφανίδη, καθώς ήταν ο άνθρωπος που με ενέπνευσε να ασχοληθώ με τον τομέα της Αλληλεπίδρασης Ανθρώπου Υπολογιστή και να φτάσω στο να έχω ολοκληρώσει αυτή τη μεταπτυχιακή εργασία. Επιπλέον τον ευχαριστώ που μου έδωσε την ευκαιρία να γίνω μέλος του εργαστηρίου Αλληλεπίδρασης Ανθρώπου Υπολογιστή και ταυτόχρονα για τη βοήθεια κατά τη διάρκεια των μεταπτυχιακών μου σπουδών. Είμαι επίσης ευγνώμων στα μέλη της εξεταστικής επιτροπής, τον καθηγητή Αντώνιο Αργυρό και την κύρια ερευνήτρια Δρ. Μαργαρίτα Αντόνα, που με βοήθησαν να βελτιώσω τη δουλειά μου με τα πολύτιμα σχόλια.

Στη συνέχεια θα ήθελα να εκφράσω την εκτίμηση μου στους συναδέλφους Δρ. Αστέριο Λεωνίδη, Δρ. Μαρία Κορόζη, και Βασίλη Κουρούμαλη για την πολύτιμη βοήθεια τους καθ' όλη τη διάρκεια των μεταπτυχιακών σπουδών, η οποία συνέβαλε στο τελικό αποτέλεσμα αυτής της εργασίας. Θα ήθελα επίσης να ευχαριστήσω όλους τους συναδέλφους μου, οι οποίοι κατά τη διάρκεια της σχεδίασης και υλοποίησης του συστήματος, βοήθησαν είτε συμμετέχοντας σε workshop, σε evaluation ή συζητώντας μαζί μου.

Ευχαριστώ θερμά τους φίλους μου Κατερίνα Βαλάκου, Βασίλη Γλαμπεδάκη, Μαρία Δουλγεράκη, Γιώργο Καλαϊτζίδη, Νίκο Λιαπάκη, Μιχάλη Λουκάκη, Νάντια Μανώλη, Αλέξανδρο Μαυρομάτη, Σταύρο Μπατζάκα, Γιώργο Νικητάκη, Δαβίδ Παραστατίδη, Βαγγέλη Πουτούρη, Ειρήνη Συκιανάκη, Εύη Καμπορούδη, και Annie Tran που με τον τρόπο τους βοήθησαν σε αυτή την εργασία και έκαναν την περίοδο εκπόνησης της πιο διασκεδαστική και ευχάριστη.

Τέλος, είμαι ευγνώμων στους γονείς μου, Γιώργο και Αυγή, για την αμέριστη υποστήριξη που μου έδωσαν καθ' όλη τη διάρκεια των σπουδών μου, που πιστεύουν σε εμένα και με στηρίζουν συνεχώς χωρίς δεύτερη σκέψη.



# Abstract

Intelligent Environments (IEs) have attracted considerable research interest in the recent years and are gradually becoming integral part of people's daily lives. Various environments such as classrooms, meeting rooms, living rooms and kitchens are being enriched with interactive displays, smart devices and sensors of various kinds, able to identify multiple users and their activities. As a result, the need to design and implement applications for IEs has dramatically increased. Unlike traditional applications, which mainly focus on a single device, interactive applications targeting IEs are expected to be distributed across various devices in the surrounding space, while numerous factors influence their behavior, such as implicit or explicit user actions, and the state of the environment (which might also be affected by user actions). These peculiarities highlight the necessity for designers to be able to evaluate the overall User Experience early in the design process, so as to identify potential shortcomings of their application and re-design them accordingly.

Existing interactive prototyping tools assist designers in building their applications for a single device, offering limited support towards creating distributed interfaces which can be used simultaneously through multiple devices, while they do not take advantage of the Intelligent Environments and their facilities (e.g. smart devices). The handful of studies addressing this issue either require users to have advanced technical knowledge (e.g. programming skills) or focus mainly in the very early stages of the design process, which eventually limits their practicality.

The purpose of this Thesis is to fill this gap by introducing the 'Wizard of AmI', an online platform for desktops and mobile devices, which supports the design process of applications for IEs by enabling the creation of interactive, prototype-based, scenarios. The 'Wizard of AmI' targets designers of

interactive context-sensitive (e.g. sensors, smart services) applications for Intelligent Environments, which can be used via multiple displays at the same time. Initially, through the system, the designers create the desired user scenarios by mapping a series of mockups to the environment (i.e. select what is to be presented and where). Next, through a novel graph-inspired interface, they define the flow and interactivity of the scenario (i.e. when it is to be presented), while at the end, they can execute the scenario either virtually or on-site, in order to assess the overall User Experience and to conduct user-based evaluation. During on-site scenario enactment, following the established ‘Wizard of Oz’ technique, the mobile companion application allows the designers to control its execution flow on demand.

**Keywords:** Interactive Prototyping, Wizard of Oz, Intelligent Environments

# Περίληψη

Τα Ευφυή Περιβάλλοντα, έχοντας προσελκύσει έντονο ερευνητικό ενδιαφέρον τα τελευταία χρόνια, αρχίζουν πλέον να ενσωματώνονται στην καθημερινότητα των ανθρώπων, καθώς πλήθος χώρων όπως σχολικές τάξεις, αίθουσες συναντήσεων, σαλόνια και κουζίνες σπιτιών εμπλουτίζονται με πολλαπλές οθόνες, έξυπνες συσκευές και αισθητήρες που αντιλαμβάνονται έναν ή περισσότερους χρήστες, καθώς και τις ενέργειες αυτών. Κατά συνέπεια, η ανάγκη για σχεδίαση και υλοποίηση εφαρμογών για έξυπνα περιβάλλοντα έχει αυξηθεί δραματικά. Σε αντίθεση με τις παραδοσιακές εφαρμογές, οι οποίες συνήθως επικεντρώνονται σε μία συσκευή, οι εφαρμογές για ευφυή περιβάλλοντα βρίσκονται κατανεμημένες στον χώρο, ενώ επηρεάζονται από πλήθος παραγόντων, όπως η κατάσταση του χώρου, η θέση του χρήστη, καθώς και το ότι μία άμεση ή έμμεση ενέργεια του μπορεί να επηρεάσει συνολικά το περιβάλλον (π.χ. κατάσταση των έξυπνων αντικειμένων και συσκευών του). Αυτές οι ιδιαιτερότητες ενισχύουν την ανάγκη δοκιμής των σεναρίων νωρίς κατά τη σχεδίαση τους, ώστε ο σχεδιαστής να αντιληφθεί την εμπειρία χρήσης (User Experience), αλλά και τις ελλείψεις της εφαρμογής που σχεδιάζει.

Τα υφιστάμενα συστήματα σχεδίασης επικεντρώνονται σε πρωτότυπα που αφορούν μία συσκευή, δεν προσφέρουν υποστήριξη για πολλαπλές οθόνες, ούτε εκμεταλλεύονται την ύπαρξη έξυπνων συσκευών. Οι ελάχιστες ερευνητικές προσεγγίσεις προς αυτή την κατεύθυνση, είτε απαιτούν από τους χρήστες να έχουν προχωρημένες τεχνικές γνώσεις (π.χ. προγραμματισμό) ή επικεντρώνονται μόνο στα πολύ αρχικά στάδια της διαδικασίας σχεδίασης, γεγονός που περιορίζει τη χρησιμότητά τους.

Αναγνωρίζοντας το παραπάνω πρόβλημα, αυτή η εργασία παρουσιάζει το 'Wizard of AmI', μια διαδικτυακή πλατφόρμα, για υπολογιστές και κινητές

συσκευές, που επιτρέπει τη δημιουργία αλληλεπιδραστικών σεναρίων βασισμένων σε πρωτότυπα, με στόχο την υποστήριξη της διαδικασίας σχεδίασης εφαρμογών για Ευφυή Περιβάλλοντα.

Το 'Wizard of AmI' απευθύνεται σε σχεδιαστές αλληλεπιδραστικών εφαρμογών για Ευφυή Περιβάλλοντα, οι οποίες μπορούν να χρησιμοποιούν ταυτόχρονα πολλαπλές οθόνες και να αξιοποιούν πληροφορίες από το γενικότερο πλαίσιο χρήσης (π.χ. αισθητήρες, έξυπνες υπηρεσίες). Αρχικά, μέσω του 'Wizard of AmI', ο χρήστης μπορεί να δημιουργήσει σενάρια χρήσης, τα οποία αποτελούνται από ένα σύνολο μακετών και αντιστοιχίζονται σε ευφυή περιβάλλοντα. Μέσω μιας πρότυπης διεπαφής σε μορφή γράφου, ο σχεδιαστής μπορεί να ορίσει τη ροή και την αλληλεπιδραστικότητα (interactivity) του σεναρίου. Τέλος, ο χρήστης έχει τη δυνατότητα να εκτελέσει το σενάριο είτε εικονικά ή στο πραγματικό περιβάλλον για το οποίο σχεδιάστηκε οπότε και να αντιληφθεί αυτοπροσώπως την εμπειρία χρήσης του καθώς και να πραγματοποιήσει αξιολόγηση με χρήστες. Κατά την εκτέλεση ενός σεναρίου, η συνοδευτική εφαρμογή για κινητές συσκευές δίνει τη δυνατότητα στον χρήστη να ξεκινήσει ή να σταματήσει ένα σενάριο καθώς και να επέμβει οποτεδήποτε ώστε να αλλάξει την πορεία του (ακολουθώντας την γνωστή τεχνική 'Wizard of Oz').

**Λέξεις κλειδιά:** Αλληλεπιδραστικά Πρωτότυπα, Wizard of Oz, Ευφυή Περιβάλλοντα

# Contents

Acknowledgments .....	vii
Ευχαριστίες .....	ix
Abstract .....	xi
Περίληψη .....	xiii
Contents .....	xv
List of Figures .....	xix
List of Tables .....	xxiii
Introduction.....	1
1.1 Motivation .....	1
1.2 Objectives .....	4
Related Work .....	7
2.1 Interactive Prototyping tools for Traditional Applications.....	7
2.1.1 Summary .....	18
2.2 Interactive Prototyping tools for cross-device Applications.....	22
2.2.1 Summary .....	26
2.3 Wizard of OZ Prototyping tools.....	26
2.3.1 Summary .....	30
2.4 Intelligent Environment Simulation frameworks .....	31
2.4.1 Summary .....	35
2.5 Discussion .....	36
Design Process .....	39
3.1 Methodology.....	39
3.2 Motivating Scenarios .....	41
3.2.1 Interactive Scenarios for eAmIDelivery.....	42
3.2.2 Designing an Intelligent Classroom .....	43

3.3.3 The Future Kitchen.....	45
3.3 Co-Design Workshop.....	46
3.3.1 Preparation .....	47
3.3.2 Co-designing .....	47
3.3.3 Discussion.....	47
3.4 Requirements .....	48
3.4.1 High-Level Functional Requirements .....	49
3.4.2 Non-Functional Requirements .....	51
Wizard of AmI.....	53
4.1 Challenges .....	53
4.2 The Wizard of AmI Core Concepts.....	55
4.3 Key Features .....	56
4.3.1 Mockup management .....	56
4.3.2 Interactive prototyping through scenarios .....	57
4.3.3 Previewing the Scenario.....	58
4.3.4 Collaboration .....	58
4.3.5 Experiencing Prototypes in Intelligent Environments.....	59
4.3.6 Wizard of Oz .....	59
4.4 The Wizard of AmI Approach.....	61
4.4.1 Setting up a project.....	62
4.4.2. Importing and organizing mockups .....	63
4.4.3. Creating a prototype through scenario .....	63
4.4.4. Running a prototype .....	64
4.5 Interactive prototypes with and without Wizard of AmI.....	64
4.5.1 Building an Interactive Prototype without having programming skills.....	65
4.5.2 Building an Interactive Prototype with help from a developer .....	66
4.5.3 Building an Interactive Prototype with Wizard of AmI .....	67
4.5.4 Discussion.....	67
4.6 Summary .....	69



Wizard of AmI Implementation.....	71
5.1 User Interface .....	71
5.1.1 Homepage .....	71
5.1.2 New project page .....	72
Space configuration functionality.....	73
5.1.3 Mockup Workspace .....	75
5.1.4 Mockup preview.....	77
5.1.5 Scenario workspace .....	80
5.1.6 Interactivity page .....	81
5.1.7 Step page .....	82
5.1.8 Preview page .....	84
5.1.9 Mobile companion application.....	85
5.2 Wizard of AmI Architecture .....	86
5.3 Wizard of AmI Database .....	88
5.4 The infrastructure behind the execution of a prototype .....	96
5.4.1 Displays .....	97
5.4.2 Smart Devices .....	97
5.4.3 Contextual Services .....	97
Cognitive Walkthrough Evaluation .....	99
6.1 Process .....	99
6.2 Results.....	100
Conclusion and Future Work.....	103
Bibliography .....	107



# List of Figures

Figure 1:Creating hotspots for mockup transitions [6].....	8
Figure 2: (a) and (b) creating interactions, (c) and (d) Uset Test results for Indigo.Design [11] and Marvel [9] respectively. ....	9
Figure 3: Prototyping on (a) Sketch [7], (b) Figma [13], (c) Adobe XD [14] and (d) Lunacy [15] . ....	10
Figure 4: Creating interactions in ProtoPie and preview [16]. ....	11
Figure 5: Origami Studio Patch Editor [17].....	12
Figure 6: Setting up interactions on (a) UX-App [18], (b) UXPin [19] and (c) Axure RP [20]. ....	13
Figure 7: Setting up interactions on (a) Balsamiq [21], (b) HotGloo [22], (c) Pidoco [23], (d) MockFlow [24], (e) Moqups [25] and (f) Fluid UI [26]. ....	14
Figure 8: Wireflow Editor [27]. ....	15
Figure 9: Prototyping in (a) Presentator [28], (b) Symu [29], (c) Principle [30] and (d) Flinto [31]. ....	16
Figure 10: Inspect tool for developers in (a) Avocode [32] and (b) Mockplus [33] and (c) the interaction pane in Proto.io [34]. ....	17
Figure 11: Setting up interactions (a) in Justinmind [36] and (b) in MockingBot [37]. ....	17
Figure 12: Frame X micro interactions [38].....	18
Figure 13: Flux Flow Editor [39]. ....	22
Figure 14: An example of a miniature ubiquitous system in MiniStudio [40].	24
Figure 15: Creating a node graph on SketchStudio editor [41]. ....	25
Figure 16: Topiary's (a) Storyboard workspace and (b) Wizard UI [46]. ....	28
Figure 17: A snapshot of BrickRoad's wizard User Interface [47] .....	29
Figure 18: The controller while running a prototype [48].....	30

Figure 19: Smart home built and simulated using OPENSIM [50].	32
Figure 20: The User Interface of eHomeSimulator [52].	33
Figure 21: Designing an indoor space in UbikSim [54].	34
Figure 22: 3DSim created virtual environment features three display and an avatar [55].	35
Figure 23: Design Thinking Process [56].	39
Figure 24: Designing lo-fidelity prototypes on the Wizard of AmI on CognitOS Board [58].	41
Figure 25: Hi-fidelity prototypes of the Wizard of AmI.	41
Figure 26: (a), (b) Prototyping during the co-design workshop and (c), (d) two of the four prototypes presented.	48
Figure 27: Wizard of AmI approach	62
Figure 28: The process for creating an Interactive prototype for Intelligent Environments without the help from a developer	65
Figure 29: The process for creating an Interactive prototype for Intelligent Environments with help from a developer	66
Figure 30: The process for creating an Interactive prototype for Intelligent Environments with Wizard of AmI	67
Figure 31: Homepage	72
Figure 32: New project (First page)	74
Figure 33: Selecting a space	74
Figure 34: Searching for a user	75
Figure 35: Space configuration	75
Figure 36: Mockup workspace with no mockups	76
Figure 37: Mockup workspace with mockups uploaded	77
Figure 38: Adding tags to a mockup	77
Figure 39: Mockup preview	79
Figure 40: Adding an interaction area	79
Figure 41: Scenario workspace	81
Figure 42: Step selected in scenario workspace	81

Figure 43: Interactivity page.....	82
Figure 44: Step page.....	84
Figure 45: Adding a mockup on a display .....	84
Figure 46: Preview page .....	85
Figure 47: Selecting project in companion application .....	86
Figure 48: The Wizard of AmI Architecture .....	88



# List of Tables

Table 1: Comparison Matrix of Interactive Prototyping Software targeting one device .....	21
Table 2 Summarize of the way that each challenge was addressed .....	69





# Chapter 1

## Introduction

### 1.1 Motivation

Ambient Intelligence (AmI) has been an active area of research for almost three decades now and has expanded the Human-Computer Interaction field. "Ambient Intelligence emphasizes on greater user-friendliness, more efficient services support, user-empowerment, and support for human interactions. In this vision, people will be surrounded by intelligent and intuitive interfaces embedded in everyday objects around us and an environment recognizing and responding to the presence of individuals in an invisible way"[1]. The emergence of the AmI paradigm can significantly improve a plethora of domains, such as the classrooms, homes, agricultural spaces (e.g. greenhouses), offices, and hospitals.

Applications for AmI Environments present unique characteristics that differentiate them from traditional applications. Traditional applications focus on one device (desktop or mobile) while applications for AmI Environments are distributed across various devices in the surrounding space, which may include multiple displays that need to be synchronized alongside the user context, meaning that they should recognize and respond to user

presence and location. Moreover, an application may include devices that are not displays, such as smart lamps and smart blinds.

While there are many different processes to design an application, User Centered Design (UCD) is widely considered as one of the key processes for creating useful and usable products [2]. User Centered Design [3] focuses on users and their needs. In each phase of UCD, users are involved through a collection of different design and research techniques (such as user observation, questionnaires, interviews, prototyping and evaluation) in order to design usable and accessible applications that are useful for the end-users. In detail, UCD is an iterative process consisting of 4 general phases, namely Specify context of use, Specify requirements, Design and Prototype, and Evaluate. For the prototype stage, the design team evolves prototypes, from low-fidelity to hi-fidelity, through iterations that involve user testing. While building such prototypes, designers use different approaches (e.g. Storyboarding, Scenarios) for connecting the different mockups they design to add interactivity to communicate the idea of the whole application interaction without the need to implement them.

An abundance of tools is available that focus on helping designers to create interactive prototypes for traditional applications (i.e. mobile and desktop applications). Common features of interactive prototyping tools consist of (i) functionality which allows the designer to connect different mockups and create interactions, (ii) a way to import mockups and add interactivity between them, (iii) collaboration functionality among the stakeholders, (iv) previewing the prototype while building it, and (v) facilitates the execution of the interactive prototype on devices that the prototypes are designed for.

Interactive prototyping while designing an Ambient Intelligence application can help designers understand better the User Experience of their designs and capture usability issues more accurately through user testing. Moreover, interactive prototyping is particularly helpful for User Based Evaluation where end-users can interact with a working prototype and evaluate it without the

need to develop it first. As the literature suggests, interactive prototypes for Intelligent Environments with higher fidelity and automated flow can enhance the data available for evaluation [4]. In addition, the Wizard of Oz technique [5], a technique that introduces the use of a user, usually a designer, as a 'wizard' who simulates the behavior or intelligence of the application, where the end users interact with a system experiencing it as a functional system where in the reality the 'wizard' operates the system and chooses the outcome based on the user's input. This technique can be particularly helpful for such applications. As the interactive prototypes usually need the context of the environment, it may be difficult to add the information in the prototype, particularly in low-fidelity prototypes, or impossible in case that the underlying infrastructure needed to generate the data is not installed in the environment.

While there is a plethora of tools for interactive prototyping of traditional applications, they are not suitable for AmI applications. AmI applications may use multiple displays, smart devices and may rely on implicit interaction from the user which derives from data generated through sensors and actuators in an AmI Environment. The research community has been focusing on frameworks that can overcome these issues. A lot of the existing research work is focusing on applications that can simulate an Intelligent Environment in a 2.5-Dimensional way or use a physical setup which mainly requires high technical skills or programming experience, making it hard for designers to prototype, evaluate, and iterate designs. Some work has been done that focuses on the ease of use of the prototyping framework that can be friendlier for designers, but the focus of those is mainly on the very early stages of the prototyping, allowing only for use of sketches or wireframes and are limited in only simulating the Intelligent Environments.

This research focuses on supporting the process of creating interactive prototypes for AmI Environments, tailored to designers, that can be used throughout the entire design process, while designing low-fidelity or high-

fidelity prototypes, and while testing/evaluating these prototypes inside the actual environment.

## 1.2 Objectives

To achieve the goal of allowing designers to create interactive prototypes for AmI Environments, this thesis proposes Wizard of AmI, a prototyping tool that allows designers to effortlessly and effectively create and experience interactive prototypes for applications in AmI Environments. Wizard of AmI aims to allow designers to create interactive prototypes for applications that use multiple displays, smart devices and sensors installed in Intelligent Environments. It provides a browser-based interface that is accessible from any modern browser where designers can build interactive prototypes, design the interaction flow in the form of scenarios, and test them in the physical environments that their prototypes are designed for. Wizard of AmI encompasses multiple features that help the designers achieve their goal of creating an Interactive prototype. The key features are:

- A sophisticated mockup management system where designers can upload their mockups and organize them using tags that represent the displays of the environment that the mockups can be assigned to. Additionally, users can add custom tags to organize the mockups in the way that they prefer. Moreover, the mockup management system offers version control for the mockups, allowing users to view the previous versions of updated mockups. Finally, users can define interactive areas on the mockups to be used in the scenario building process.
- A graph-based environment that allows users to create interactive prototypes through scenario building, where they can create steps, representing the different states of the system, and set connections between the different steps to create the flow of the scenario. Additionally, Wizard of AmI offers a 2D representation of the space where users can

assign mockups to the different displays of the environment for each state of the scenario.

- Collaboration among users where designers can work together concurrently while creating a prototype. Additionally, prototypes can be shared to other users to comment.
- A preview mode where users can view their prototype while building it. The preview mode is a 2D representation of the environment where users can interact with the different displays to experience the flow of their scenario.
- A unique way to experience the created interactive prototypes. Users can enact their interactive prototypes within the actual physical environment that prototypes are designed for. In more detail, they can interact with the displays of the environment and experience the flow of the scenario while being in the actual environment.
- A Wizard of Oz application that allows designers, while running their prototype in the physical environment, to simulate different aspects of their creation. The application is available for tablets and smart phones and provides tools, that based on the state of the running scenario, allow for triggering state transitions without the need to be modeled while creating the prototype.

Wizard of AmI proposes a process that simplifies the overall complexity of creating interactive prototypes for Intelligent Environments. The process is separated into 4 sequential parts, namely a) Setup a project, where users first configure the Intelligent Environment that will be used for the prototype, b) Import mockups and categorize, where users upload their mockups and organize them based on the devices that mockups are designed for or based on user preferences such as the order of the scenario, c) Create a prototype through scenario, where users create steps that represent a state that the whole system will have (i.e. which mockup will each display show, in what state will the smart devices be) and d) Run the prototype, where users

experience their interactive prototype in the physical environment that the prototype is designed for.

The rest of this thesis is organized in the following way:

- Chapter 2 reviews related work regarding Interactive Prototyping Design tools.
- Chapter 3 describes the design methodology that was followed, introduces several scenarios that motivated this work, the process that was followed in a co-design workshop, and outlines the functional and non-functional requirements of the proposed framework.
- Chapter 4 describes the functionality of the system.
- Chapter 5 elaborates on the implementation details.
- Chapter 6 describes the evaluation process and its results.
- Chapter 7 concludes the thesis with a summary of the results and a discussion of possible future directions.

# Chapter 2

## Related Work

This chapter reviews Interactive Prototyping tools, Wizard of Oz frameworks, and Intelligent Environments Simulation software, and subsequently presents a summary of the features that those frameworks support. The importance of reviewing the aforementioned categories of tools and frameworks relies on the scope of the Wizard of AmI, as it is an interactive prototyping tool for Intelligent Environments which can simulate Intelligent Environments and uses the Wizard of Oz technique while enacting interactive prototypes in such environments.

### 2.1 Interactive Prototyping tools for Traditional Applications

**InVision** [6] is a well-known web-based prototyping tool with a main focus on collaboration. It features a plethora of tools that seamlessly work together to help a team design an application. Namely some tools are: Prototype, Preview, Craft, Inspect and Freehand. A user or team can create a project and upload their mockups to start creating prototypes. Using the prototype feature, a user can add hotspot areas by creating a rectangle on top of a mockup and establish a transition to another mockup. Preview mode allows the user to run the interactive prototype either on the device that they are building the prototype or to another device (for example a mobile phone) through a sharable link. Freehand is a handy collaboration tool that acts as a web-based whiteboard where users can view the mockups of their project and create notes on top of

them for feedback in real-time. Craft is an addon for design applications such as Sketch [7] that syncs Sketch projects to InVision with features such as uploading a mockup directly from inside the application and connecting the prototyping workflow from the application to the InVision Project.

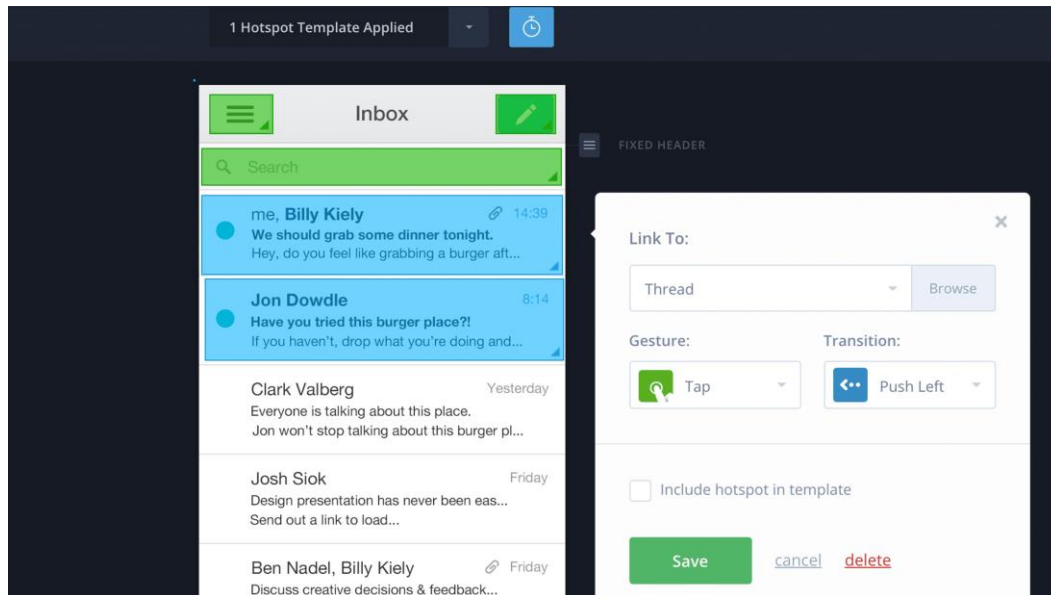


Figure 1: Creating hotspots for mockup transitions [6]

**Marvel** [9] and **Evrybo** [10] are web based prototyping platforms similar to InVision in the way that a user can upload design mockups and transform them into interactive prototypes. They follow the same idea of selecting areas on mockups and linking them as transitions to other mockups. They also feature collaboration in the sense that other users can comment on mockups for feedback. **Indigo.Design** [11] is a prototyping tool that also has a built in facility for creating mockups. It provides users with a unique design system that features prebuilt customizable UI kit components which can generate Angular [12] UI components ready to be used for development. Both Marvel and Indigo.Design have a User Testing feature built in. A session can be captured with event tracking (e.g. click and tap) and key insights such as test success rate and time spent on each task. Both tools also support video and voice capturing of the user while testing the prototype. Indigo.Design also



supports the creation of unmoderated user tests that can be sent directly to users to do on their own time.

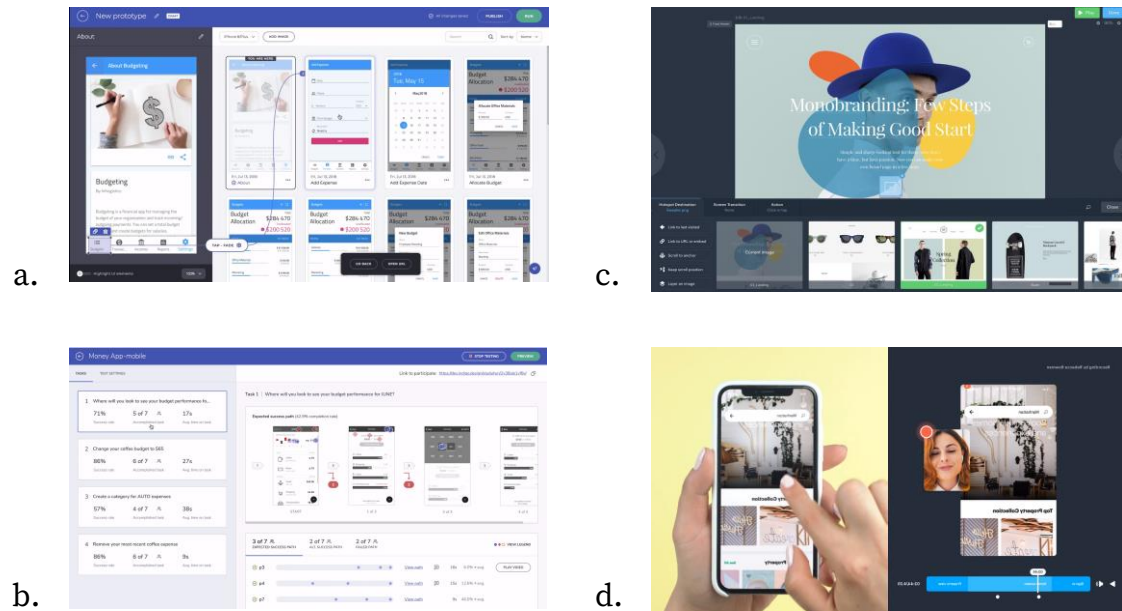


Figure 2: (a) and (b) creating interactions, (c) and (d) User Test results for Indigo.Design [11] and Marvel [9] respectively.

**Sketch** [7] , **Figma** [13], **Adobe XD** [14] and **Lunacy** [15] are all-in-one tools for vector-based design and prototyping. All of them feature macOS and Windows applications apart from Sketch, which is only available on macOS and Lunacy which supports only Windows platforms. Figma is also available as a browser-based application. Their main focus is the vector-based design with features such as a pen tool that allows users to create any type of vector elements, arc tool to easily build clock, watch and pie chart designs and auto layout which helps designers to create responsive user interfaces. Figma also supports live collaboration by design where multiple users can work in parallel on the same project, which differentiates it from other design tools. All of them support interactive prototyping where a user can create interactions to link different mockups and even support more advanced interactions such as opening a modal. Users can set transition effects to give their designs a more realistic feel. A preview mode is available, allowing the users to run their designs and

interact with mockups. The preview mode creates a sharable link which can be used to run the prototype on any device with internet connection.

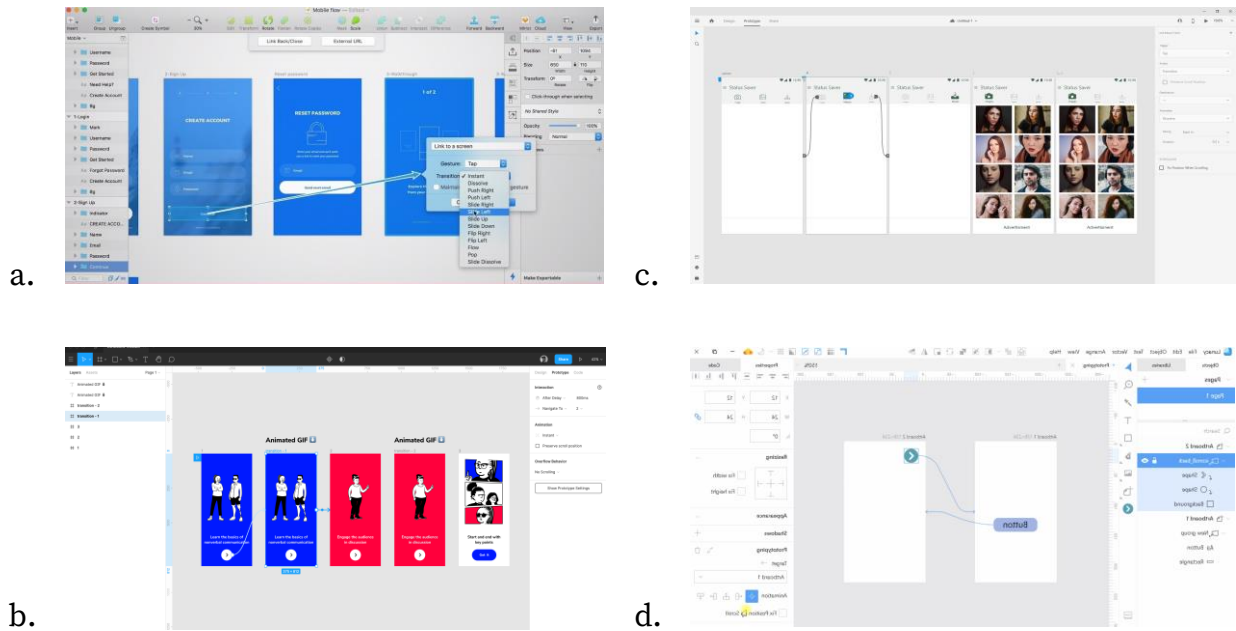


Figure 3: Prototyping on (a) Sketch [7], (b) Figma [13], (c) Adobe XD [14] and (d) Lunacy [15] .

**ProtoPie** [16] is a prototyping tool that presents a different approach on creating prototypes. Users can incorporate logic into their prototypes by creating variables, formulas, and conditions for their prototypes and they can take advantage of smartphone sensor data which allows for more detailed and advanced prototypes with micro interactions resulting to a prototype that looks and feels much closer to an actual product. Mockups can be imported directly from well-known design tools such as Adobe XD, Figma and Sketch. Each interaction consists of three fundamental components, an object, a trigger and a response. Triggers include touch and mouse interaction, input from keyboard, and smartphone sensor data; triggers can also be conditional such as a range of values. Objects can be any of the UI elements of a mockup. Finally, the responses are the result that an action will have which can be either a change of a property of an object (e.g. position on the mockup, background color etc.) which creates an animation, a link to another mockup, or even the vibration of a device or the playback of a video. It is also possible

to chain multiple responses to a trigger. In theory this makes it possible to create a wide range of complex animations and transitions. A unique characteristic of the aforementioned model is that multiple running prototypes can communicate as the designer can set a trigger that accepts messages from external sources and a response that broadcasts a message. Designers can connect a device and run their prototypes on a smartphone using a link (which is also accessible through an auto generated QR code).

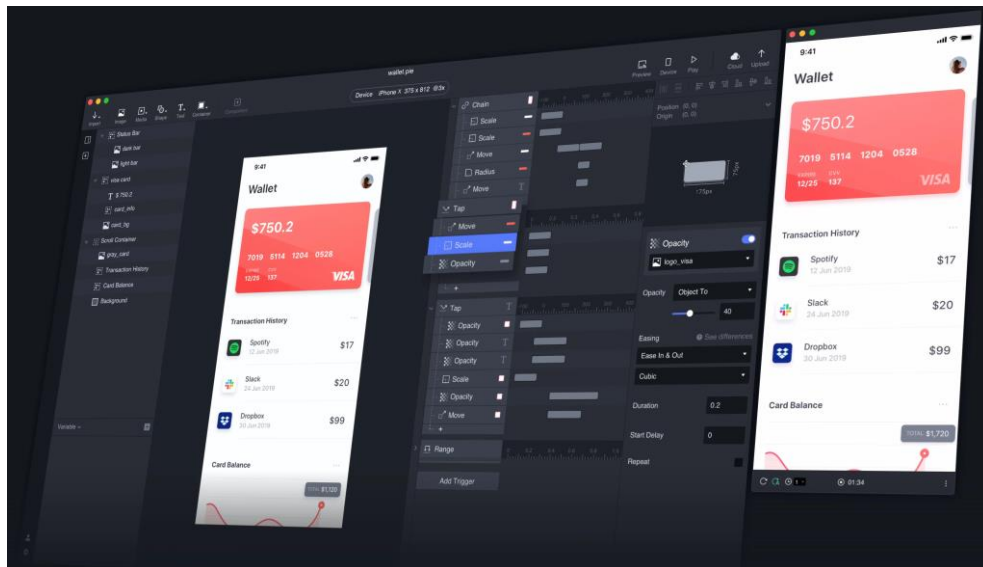


Figure 4: Creating interactions in ProtoPie and preview [16].

**Origami Studio** [17] from Facebook is another prototyping tool that follows a similar approach to ProtoPie. Interactions can be enriched with logic for complex prototypes with conditional transitions and data aware components. It supports input data from devices or sensors, such as a camera and apple's 3D touch. It also supports advanced features with network requests for real data, such as real weather status, which adds to the realism of the prototype. Every custom interaction is built using an editor called Patch Editor. Patch editor is a block-based editor where patches are the building blocks. Patches allow users to add interactions, animations, and behaviors to their prototypes.

Each patch can have inputs and outputs that can range from a simple number or Boolean to even a JSON file.

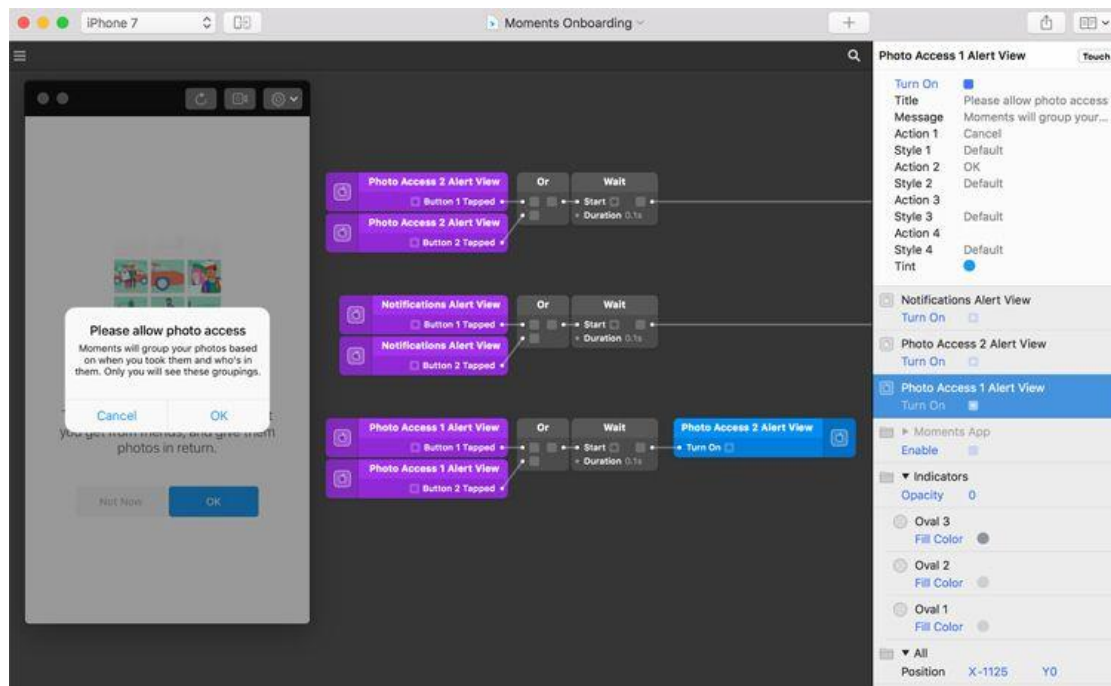


Figure 5: Origami Studio Patch Editor [17].

**UX-App** [18], **UXPin** [19] and **Axure RP** [20] are prototyping tools that allow for click-through prototyping and web sharing of the prototypes by generating a link that can be accessed from other devices. They also allow for collaboration in the sense that users can add comments on the mockups. They support advanced prototyping by allowing users to program interactions through simple scripts with variables and conditions. Another feature shared on all these tools is the ability to extract the styling of the components and the use of

interactive elements that are generated from code which can help bridge the gap between the design and the development of an application.

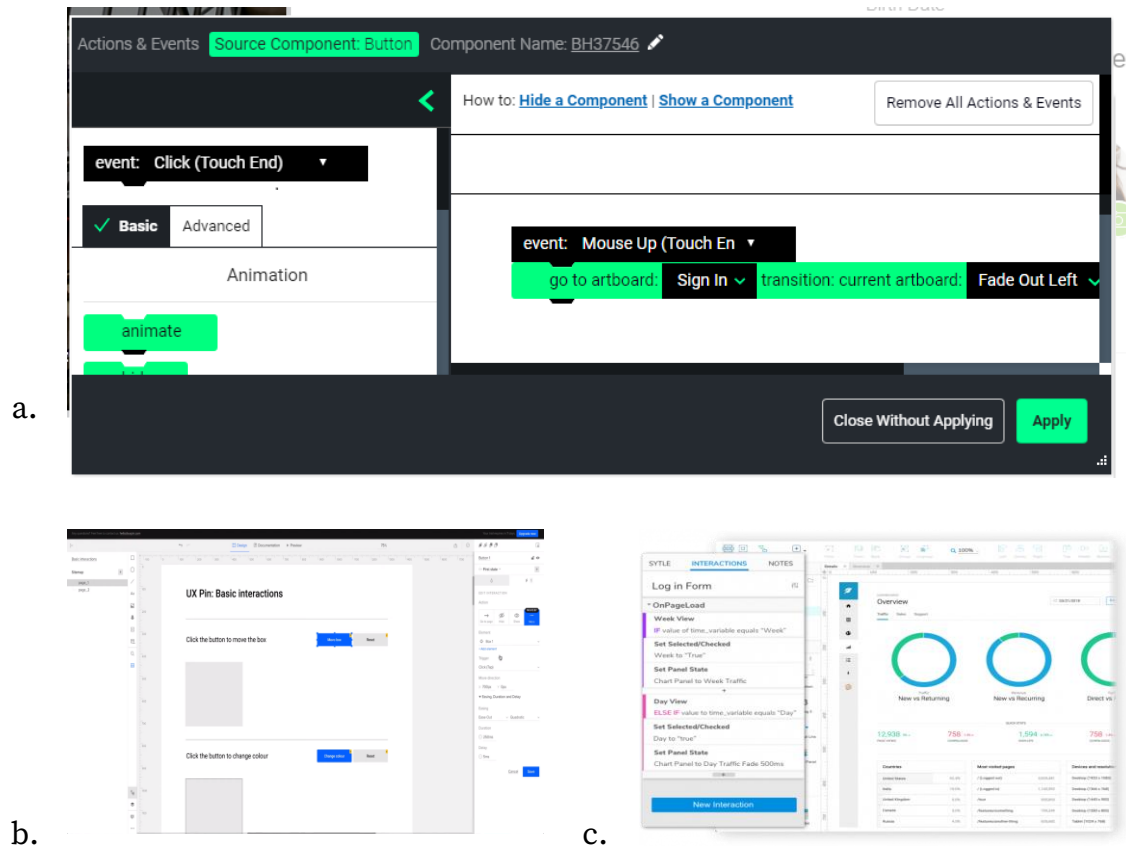


Figure 6: Setting up interactions on (a) UX-App [18], (b) UXPin [19] and (c) Axure RP [20].

**Balsamiq** [21], **HotGloo** [22], **Pidoco** [23], **MockFlow** [24], **Moqups** [25] and **Fluid UI** [26] are wireframing and prototyping tools. Most of them are web based while Balsamiq is only available for Windows and MacOS. They feature ready-to-use basic UI components (such as buttons, input fields, image places holders etc.) that can be used to create a visual guide which represents the skeletal framework of the application. The way that the wireframes can be built in all these applications is similar, the user can drag components from a component library and drop them into a canvas to create the desired wireframe. The components look like they were sketched on a paper (with small exception where some components are more visually pleasing with colors and round borders but nevertheless simple) allowing users to focus on

the layout and interaction patterns rather on other details that are not the objective of wireframes. Users can add simple interactions that create transitions from one wireframe to another. Interactions are simpler than those of other prototyping applications, as animations and advanced interactions, such as overlaying another smaller mockup on top of another, are unnecessary for wireframes.

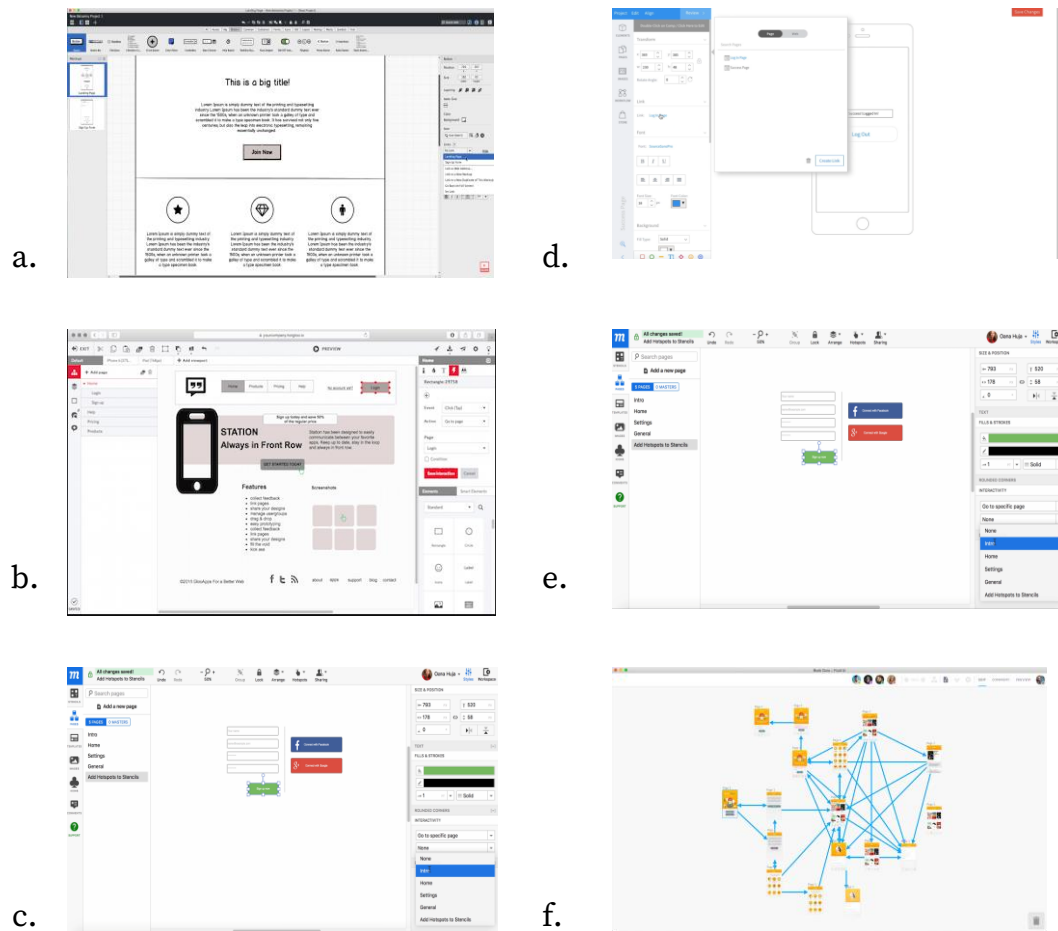


Figure 7: Setting up interactions on (a) Balsamiq [21], (b) HotGloo [22], (c) Pidoco [23], (d) MockFlow [24], (e) Moqups [25] and (f) Fluid UI [26].

**Wireflow** [27] is a simple web-based tool for creating user flows. It is intended to be used in the early stages of project planning and brainstorming. It provides a simple whiteboard User Interface where the user can drag-and-drop prebuilt UI sketches that are representations of common web and mobile pages. These elements can be stacked together to create a single page, grouped

together for semantical use and connected using arrows to represent the flow of the designed application. On the bottom right there is also a mini map that makes navigation for large scale flows easier. Users can also download their user flow prototypes as images.

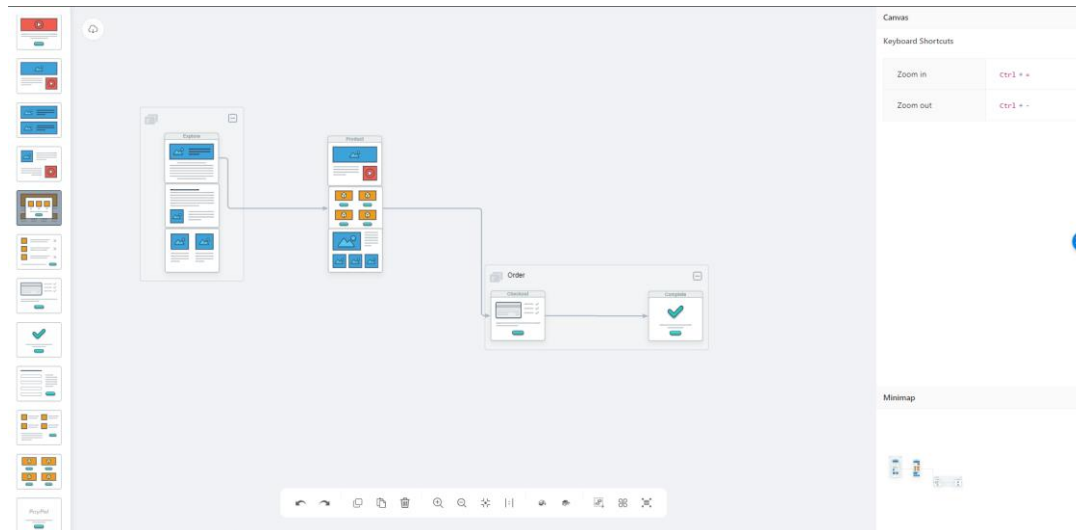


Figure 8: Wireflow Editor [27].

**Presentator** [28], **Symu** [29], **Principle** [30] and **Flinto** [31] are prototyping tools that allow designers to create interactive and animated prototypes. Principle and Flinto offer exclusively macOS applications, while Presentor and Symu offer a browser-based application. User can either directly upload their mockups or import them from well-known design tools using provided addons. They can then create interaction areas on the mockups to create transitions from one mockup to another. Each tool has some unique features that help designers to create dynamic prototypes that result to a feeling like a working application. All of them provide some type of collaboration features where users can leave comments directly on their mockups and collaborate efficiently to finalize their design. Flinto and Principle offer some tools to edit mockups and create more realistic animations. Flinto differentiates from the others by providing a behavior designer to create micro-interactions with screens, which maximizes the realistic feel of UI elements such as buttons and switches.





Figure 9: Prototyping in (a) Presentator [28], (b) Symu [29], (c) Principle [30] and (d) Flinto [31].

**Avocode** [32], **Mockplus** [33] and **Proto.io** [34] are prototyping tools with a focus on collaboration between designers and developers. Designers can drag and drop interactive and customizable UI components or import their designs directly from popular design tools such as Adobe XD or Photoshop [35]. All of them provide collaboration features where designers can get feedback from users. A preview mode is available to view prototypes in action. Users can extract specs, assets, interaction flows and CSS snippets to help the development of the designed application. Mockplus and Avocode provide a handy tool for developers where they can inspect mockups to extract any needed information for the development of the design.

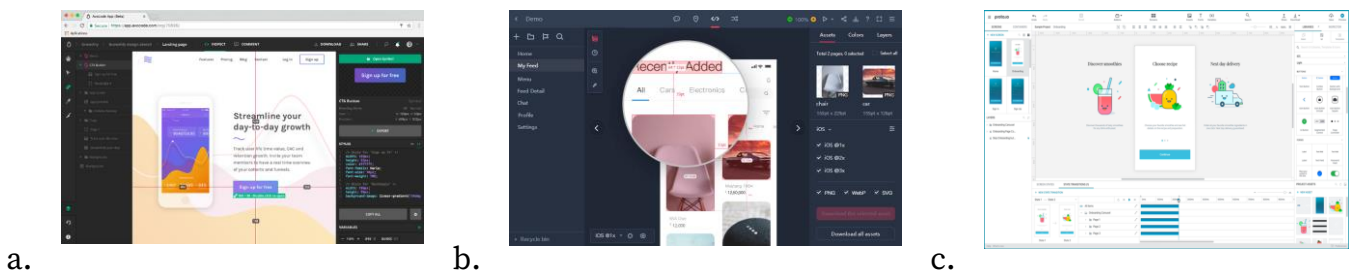




Figure 10: Inspect tool for developers in (a) Avocode [32] and (b) Mockplus [33] and (c) the interaction pane in Proto.io [34].

**Justinmind** [36] and **MockingBot** [37] are all-in-one design and prototyping tools for web and mobile applications. Justinmind is a desktop application for Windows and macOS, while MockingBot is web-based. Both include libraries with widgets or UI elements that can be dragged and dropped to create prototypes ranging from simple wireframes to high-fidelity mockups. They both support interactions from one mockup to another and basic animations. UI Kits and templates with rich UI assets are available to easily add to a project. As with the other prototyping tools, both support preview of the prototypes and ways to view the prototypes on the desired devices that they are built for (e.g. tablet, smartphone).

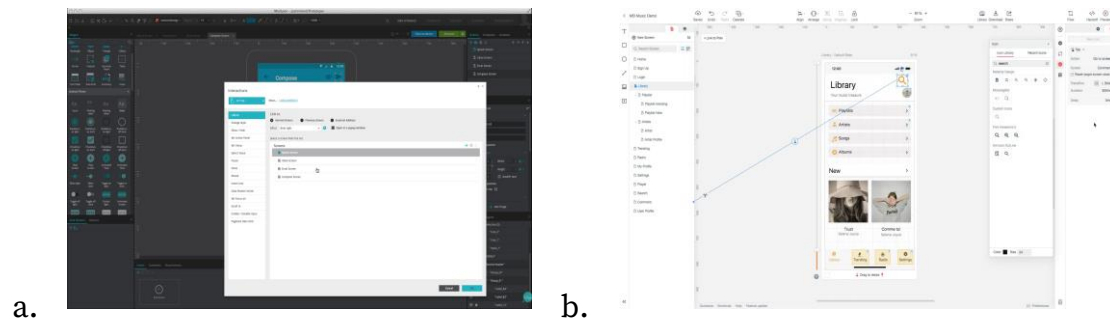


Figure 11: Setting up interactions (a) in Justinmind [36] and (b) in MockingBot [37].

**Framer X** [38] is a browser-based and macOS prototyping tool that tries to differentiate from other prototyping tools by providing professional grade animations and extensive interaction support. It supports collaboration for easy sharing of the project and inline commenting on mockups. Users can either import mockups from design tools like Sktech and Figma or create new mockups using the built-in editor. Teams can create libraries with interactive UI components to reuse on different design projects. The tool takes advantage

of a features called 'Magic Motion' where it detects the best possible transition animation from one mockup to another.

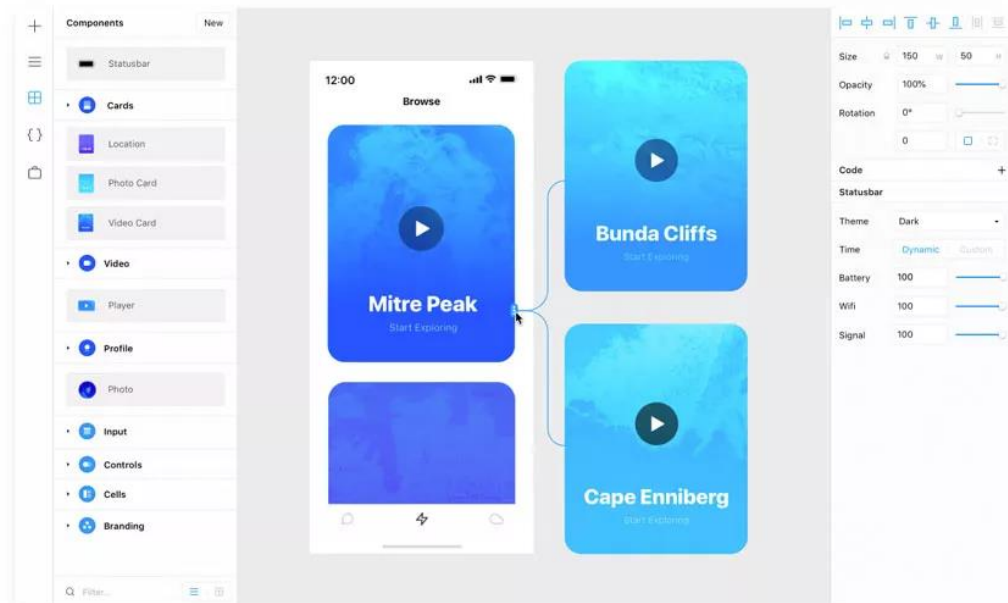


Figure 12: Frame X micro interactions [38].

### 2.1.1 Summary

The literature review revealed several features that are supported by frameworks targeting Interactive Prototyping for a single device, which are summarized below.

**A. Functionality to transform a set of mockups or wireframes to an interactive prototype** which is the main purpose of interactive prototyping tools. All the tools support this function by allowing the user to select an area as an interactive spot that when an action (e.g. click) happens, a transition to another mockup occurs.

**B. Functionality to create or import mockups.** As this is a basic feature that is needed to create a prototype, all tools support this feature. Most tools that are 'by design' prototyping tools feature addons for popular designing applications that allow importing mockups directly from within those applications while others even provide a tool for creating mockups directly from inside the tool either by using UI components or creating them from scratch using vector-based tools.

**C. Collaboration among stakeholders.** This feature is supported by all interactive tools but to a different extent. There are tools that allow live collaboration while creating a prototype where each user can contribute to the creation process. Other tools have a feature where users can add comments on connections and mockups of the prototype for feedback. Collaboration is also available on some tools between designers and developers to help with the process of developing the prototype.

**D. Preview while building a prototype** refers to the ability for a designer to preview their prototype while creating their prototype to view and interact with in order to validate the flow that they are building. All tools support previewing.

**E. Execution of the prototype** is the ability to run the prototypes on the devices they are designed for. The majority of the tools support that by using a sharable link that can be opened from any browser or by providing an application that can run a specific prototype (this case is more applicable for mobile prototypes where there is a companion application that a user can download on their device to run prototypes created with the prototyping tool). This feature is important as this allows end-users to test the prototype on the intended to use devices.

**F. Interaction Logic** is the ability to create more complex connections by adding logic into the prototypes. Multiple tools support this feature in different ways. The main idea is the definition of conditional transitions based on context that is generated during the execution of the prototype. This is a type of programming and depending on the complexity of the logic that a tool supports; users can use block-based programming or even write scripts on high-level programming languages created specifically for prototyping tools that are closer to natural language.

**G. User Testing** refers to the ability of the prototyping tools to run user tests without the need for an external application for this process. Only two of the reviewed tools contain this feature, with the ability to record events while

running a test, including video and voice capturing, and insights about the success rate for each task.

Table 1 presents the features, categorized previously, that each of the studied frameworks support. B is separated into three sub categories: **B1** is the ability to **add mockups in the tool** to use as mockups, **B2** is the ability to **import mockups directly from other design tools** using provided addons from the prototyping tool, and **B3** is the ability to **create mockups inside the tool** either through a vector-based system or using UI components to create composites. C is separated into three categories, **C1** is the collaboration in a manner that allows **multiple users to edit a prototype simultaneously**, **C2 is collaboration among stakeholders to comment on designs and flows** of the prototype, and **C3 is the collaboration between developers and designers** with tools that help developers to transform a design to a working application.

Table 1: Comparison Matrix of Interactive Prototyping Software targeting one device

	A	B			C			D	E	F	G
		1	2	3	1	2	3				
Invision [6]	✓	✓	✓		✓	✓		✓	✓		
Marvel [9]	✓	✓	✓	✓		✓	✓	✓	✓		✓
Evrybo [10]	✓	✓	✓	✓		✓		✓	✓		
Indigo.Design [11]	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Sketch [7]	✓	✓		✓		✓		✓	✓		
Figma [13]	✓	✓		✓	✓	✓		✓	✓		
Adobe XD [14]	✓	✓		✓	✓	✓	✓	✓	✓		
Lunacy [15]	✓	✓	✓	✓				✓	✓		
ProtoPie [16]	✓	✓	✓	✓	✓	✓		✓	✓	✓	
Origami Studio [17]	✓	✓	✓	✓				✓	✓	✓	
UX-App [18]	✓	✓		✓		✓	✓	✓	✓	✓	
UXPin [19]	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Axure RP [20]	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Balsamiq [21]	✓			✓	✓	✓		✓			
HotGloo [22]	✓			✓	✓	✓		✓			
Pidoco [23]	✓			✓	✓	✓		✓	✓		
MockFlow [24]	✓	✓		✓	✓	✓		✓	✓		
Moqups [25]	✓	✓		✓	✓	✓		✓	✓		
Fluid UI [26]	✓	✓		✓	✓	✓		✓	✓		
Wireflow [27]	✓			✓							
Presentator [28]	✓	✓	✓		✓	✓		✓	✓		
Symu [29]	✓	✓	✓		✓	✓		✓	✓		
Principle [30]	✓	✓	✓	✓				✓	✓		
Flinto [31]	✓	✓	✓	✓				✓	✓		
Avocode [32]	✓	✓		✓	✓	✓	✓	✓	✓		
Mockplus [33]	✓	✓	✓		✓	✓	✓	✓	✓		
Proto.io [34]	✓	✓	✓	✓		✓	✓	✓	✓	✓	
Justinmind [36]	✓	✓	✓	✓		✓	✓	✓	✓		
MockingBot [37]	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Framer X [38]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	

## 2.2 Interactive Prototyping tools for cross-device Applications

**Flux Studio** [39] is a Prototyping tool for Cross-Device Interactions. Flux is a macOS application with a flow editor and devices overview. The user can add devices from a library that support an abundance of different types such as displays and potentiometers. It is based around the idea that the user can visually link the device states together. Instead of writing code or describing linear flows, users can create dynamic prototypes in Flux that operate between multiple devices. For example, one can create a prototype for a switch and a prototype for a display where they can set states for each prototype, on/off states for the switch and two different mockups for the display, and visually link the states. There is a preview mode available where the user can test their prototypes and interact with the different devices.

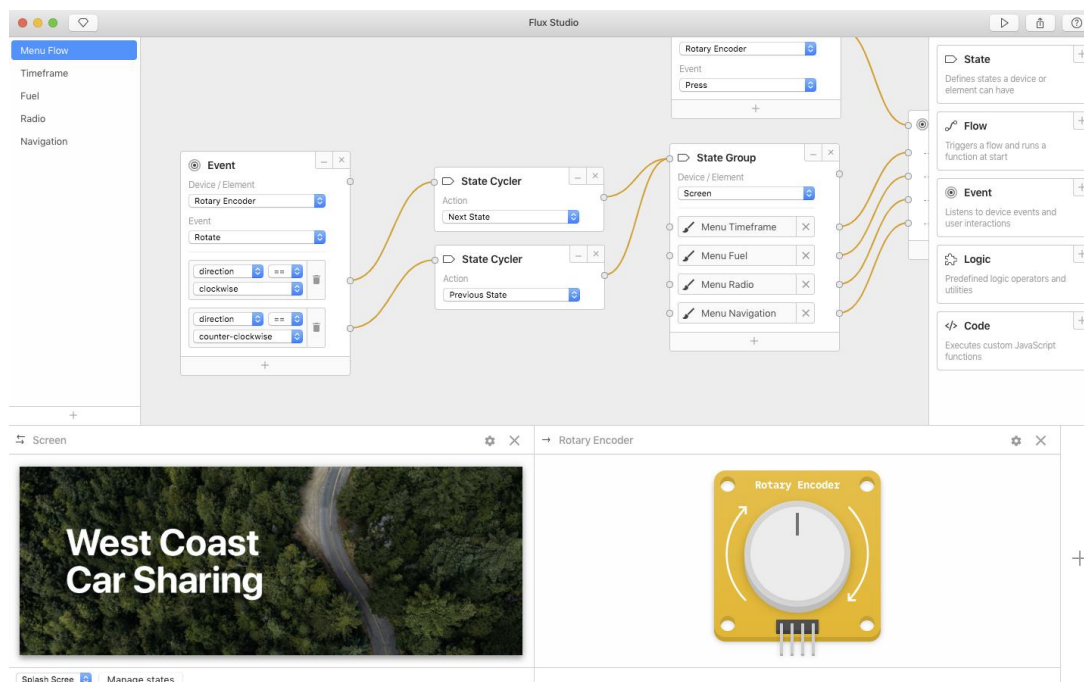


Figure 13: Flux Flow Editor [39].

The work in [40] introduces **miniStudio**, a Designers' tool for prototyping ubicomp spaces with interactive miniature. The tool intends to help designers with proxemic interactions in large-scale ubicomp spaces. The process they

follow to design a prototype is divided into three steps. The first step is the construction of the miniature models. miniStudio offers a plugin for photoshop. The plugin provides several plate-based model types as templates. It also contains instructions on assembling the prototypes. It can auto generate free-form models from an image. After creating the models, each model is assigned a unique identifier that is printed with an Augmented Reality (AR) Marker. The second step is the definition of events and states for the miniatures. Events can be set through a dialog panel in the photoshop addon. Users can set different event parameters such as distance or motion. Different layers of the Image play a role of the different states of the miniature. The third and final step is testing and modifying the miniatures. The setup of the miniature prototype is one camera-projector system that tracks the miniatures using the AR markers and superimposes each miniature from the created images according to the layers created on step 2 and event conditions. The authors conducted a user study with 12 participants, familiar with designing ubiomp applications. The result of the study showed that using a familiar tool to create a prototype was intuitive and a relatively fast process. Another finding is that the augmented surfaces were creating a more realistic feeling as the high-fidelity images were projected. They found that miniatures were limiting on providing actual experience due to imperfect scaling.



Figure 14: An example of a miniature ubiquitous system in MiniStudio [40].

The work in [41] presents **SketchStudio**, a prototyping tool for creating, sharing and evaluating animated design scenarios for spatial wide scenarios. The tool helps designers to define the user experiences by simply building a node graph of the user's journey through time and space. SketchStudio instantly generates unique 2.5D animated design scenarios by integrating 2D sketches and 3D characters. In addition, the tool supports the virtual reality mode for immersive viewing of the generated scenario prototype. Authors conducted a study of nine users with experience in the design of spatial interactions. They argue that from the results of the study they found that it supports rapid creation and execution of design scenarios effectively. The authors also shared insights from the user study that can benefit future interactive system design tools. One outcome of the study is that 3D modeling for spatial design can help to better visualize dynamic scenarios rather than static models. Another outcome was that users would like text input support, which can be generalized as an important function for any application that supports scenario building in the early stages of experience design. Finally, an interesting idea that arose from the user study was the ability to capture and



export static scenes to create conventional storyboards while a video recording can help to visualize and communicate a prototype.

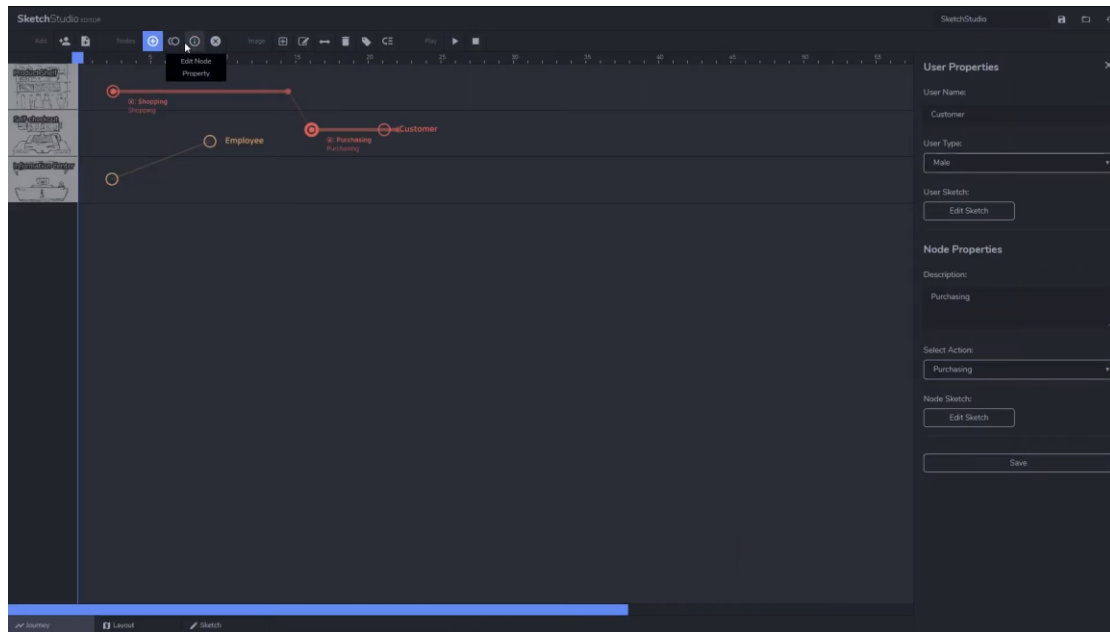


Figure 15: Creating a node graph on SketchStudio editor [41].

The work in [42] introduces a framework that allows designers and developers to create Interaction Designs in Intelligent Environments. In more detail, the authors contribute with a framework to make interaction easier and a tool that supports interaction design for Intelligent Environments. The framework tries to fuse concepts coming from Natural User Interaction theories with concepts from Intelligent Environments [43], [44]. The authors propose an architecture that abstracts the behavior of Intelligent Environments resulting to an overall behavior that is not related to a single entity. The Interaction Design tool is a block-based interface where each block represents either an input event, a device, an environment or a goal and users can connect those blocks to create the interaction design. As the tool was in early stages of development when the authors published this work, there is no specific plan for the interaction design tool and will depend on further research and user studies.

### **2.2.1 Summary**

The literature review of prototyping tools for Intelligent Environments with support for multiple displays and interactions reveals that there is a limited number of research and tools related to this area. This area presents obstacles that are difficult to overcome. Creating a flow for a single device when you need to model multiple interactions is difficult and time consuming; by adding more devices and services, the complexity of the flow grows exponentially. Due to this problem, work in this field is usually focused on one specific area of Intelligent Environments and even more specifically to one type of application (e.g. an application that is centered around a smart table or a location-enhanced application [45]), which can be more effective for prototyping applications targeting specific branches of Intelligent Environments. Another finding is that the research done for prototyping tools for Intelligent Environments, focuses on early stages of the design process and therefore it is based on sketches or wireframes.

## **2.3 Wizard of OZ Prototyping tools**

The work in [46] presents **Topiary**, the first prototyping tool for location-enhanced applications with support for modeling both indoors and outdoors location prototyping. Designers can create storyboards for location-enhanced applications and 'fake' location information through wizard of Oz techniques or even use real sensors and incorporate the actual location of the user. The tool contains three parts. The Active Map, the Storyboard and the Test workspaces. In the Active Map workspace, designers can sketch or upload an image as a map and add entities (i.e. people, places, and things), model spatial relationships between those entities and then build scenarios describing location contexts. There are dedicated tools inside the Active Map for creating each entity and scenarios. A relationship is based on basic location contexts that Topiary supports (for example near/far from a location or enters/exits a

location). Scenarios are a collection of location contexts that can be used for specifying behaviors. Topiary provides a tool that allows designer to build scenarios visually on the drawn map by selecting areas and visually represent a location context, for example if the designer selects an area and a person and then they place the person inside an area, the system identifies that the context is 'user enters area'. The storyboard workspace allows users to create pages that represent screens and links that are transitions between pages. Pages are drawn directly inside the tool using a pen tool and links are drawn lines from one page to another. Links can be either explicit or implicit. Explicit links are actions that a user has to do on the pages (i.e. click on a specific component) while implicit link are transitions that are triggered from sensed data. Scenarios can be attached on a link. This way, designers add the logic to the prototype because the transitions from a page to another page are made based on the state of the location context. Finally, the Test workspace allows designers to run their prototypes. The test workspace has two components, the End-User User Interface which is what the end user will interact with (i.e. the pages created in the Storyboard) and the Wizard User Interface. These can either run on the same device to preview the prototype or in two separate devices. The wizard UI contains a map that allows the wizard to update the current position of the user and buttons to start, end, or replay a session. Additionally, the wizard UI has a window that can show the current pages and the last transition which help designer figure out the interaction flows and a viewer that mirrors the end user UI where the wizard can also trigger the end-user explicit interactions.

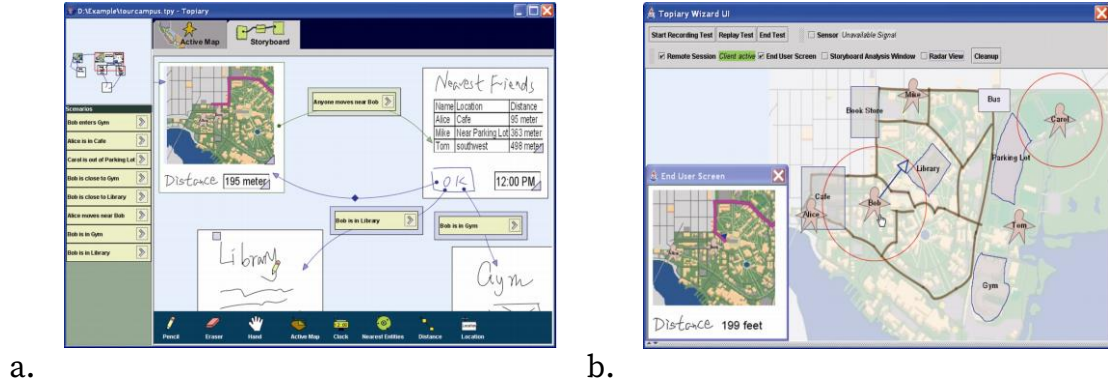


Figure 16: Topiary's (a) Storyboard workspace and (b) Wizard UI [46].

The work in [47] introduces **BrickRoad**, a Wizard of Oz tool for early stages of designing location-enriched applications. It does not require any interaction logic to be specified before running any test and designers can experiment with different designs on the fly and perform changes in their prototype while executing it. The wizard, who is usually a designer, can simulate interaction logic and location tracking while running an experiment from the BrickRoad wizard interface. The wizard is divided in three panels. The first panel is the canvas that shows a map of the simulating area. Canvas also contains a representation of the UI that the end-user currently observes which is presented on the current position of the user, and trail with user's previous locations that also shows the previous UIs the end-user had. The second panel is the Resource Area, a component library with different components that can be sent to the end user UIs. The wizard can also combine multiple components to create a composite and then send it to the end-user. The third panel is a chronological timeline that shows the past end-user UIs. The process that a designer has to follow in order to use BrickRoad is: a) they have to load a map representing the space where the user will be b) they have to prepare the material for the prototype, such as images, text, and audio clips for the Resource Area. After the preparation they can run the test. While running the test, a wizard application for tablets allows the wizard to spontaneously create the composites and to select the positioning of the user. When a user inputs an

action on a device, the device freezes and waits for the wizard to send a new composite. The authors claim that they used BrickRoad for several location-enhanced application and believe that it is a valuable addition for the early stage of designing location-enhanced applications.

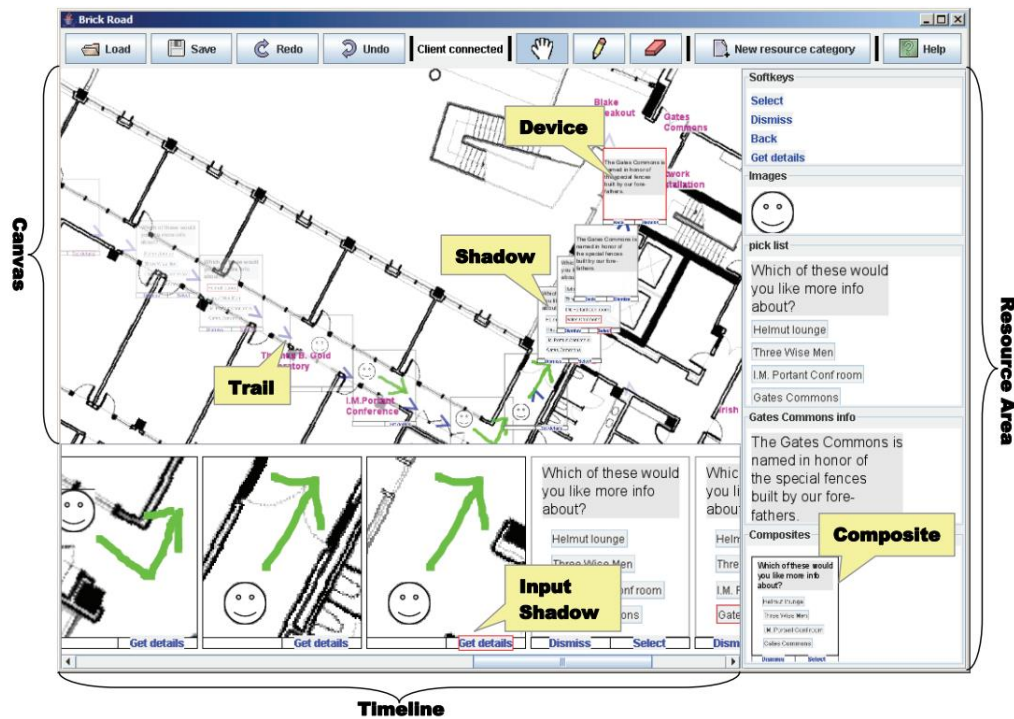


Figure 17: A snapshot of BrickRoad's wizard User Interface [47]

The work in [48] presents a contextual Wizard of Oz prototyping framework. The goal of this work is to allow for easier and faster prototyping of contextual applications in specific contexts in order to conduct user studies in flexible Ambient Intelligence setups. The framework provides a tool that allows users to create prototypes that combine real sensor data and simulated contextual data. The tool offers a visual programming environment that allows users to visualize the data flow of the prototype through wires (i.e. lines that connect the different states of the prototype) while users can add contextual data from real sensors and actuators alongside simulated ones. Simulating contextual data is important in case that the underlying infrastructure that generates the data is not installed, or special cases that either are dangerous to produce with real data or difficult in general to reproduce. The framework offers a Wizard

of Oz tool that can configure the context of the prototype while running. This allows for the creation of semi-functional prototypes where a user can simulate the non-functional part of the prototypes. The authors conducted an evaluation that revealed the importance of a tool that can simulate context in Ambient Intelligence setups. The results of the evaluation indicated that the visual programming environment made the learnability of the tool easier, and that the Wizard of Oz approach is necessary and useful while prototyping for such environments.

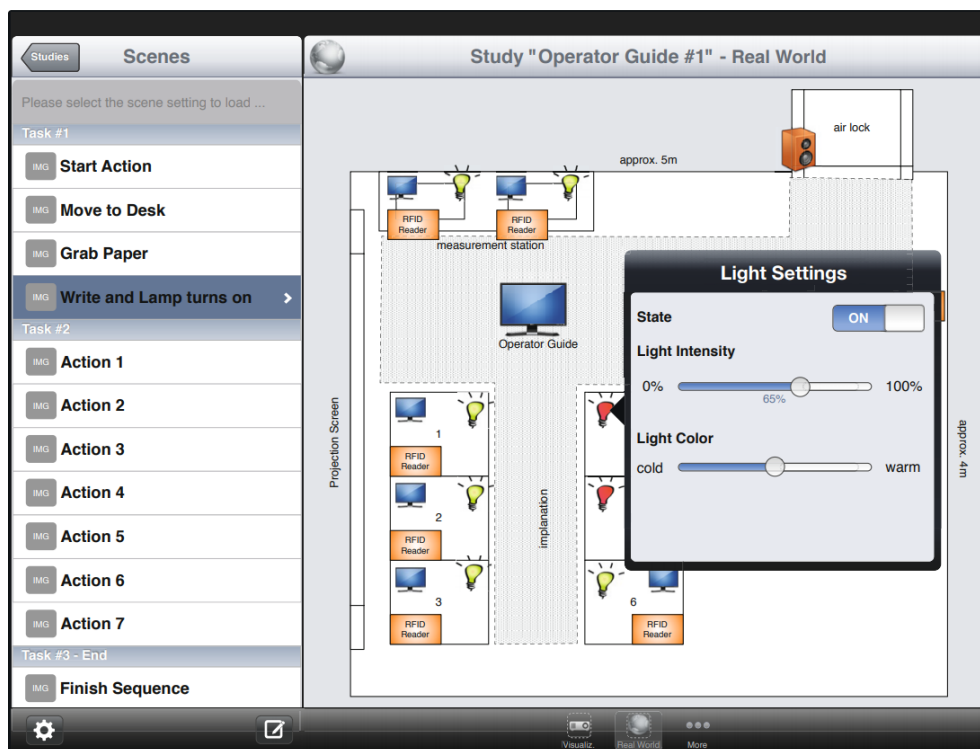


Figure 18: The controller while running a prototype [48]

### 2.3.1 Summary

Wizard of Oz is a technique which was first introduced in early 90s for natural-language dialog systems [5]. The Wizard of Oz technique is used in the design process and refers to the use of a 'Wizard', which can be one person or more that simulate missing functionalities and system intelligence of a system. The Wizard of Oz technique is particularly helpful for Intelligent Environments [49] as it allows for simulating sensing data, such as user location, and complex

intelligence that an Intelligent Environment requires, removing the need for software implementations or hardware requirements that would be necessary. The literature review of prototyping tools that use Wizard of Oz techniques to simulate states of an environment or an application, reveals that the tool that the wizard will use while running an experiment should have an interface curated towards the application that it simulates is needed in order to help the wizard in simulating events and state changes effectively, based on the context of the prototype's state while running. Moreover, most of the reviewed frameworks support a way to add real data from sensors to minimize the work of the wizard.

## **2.4 Intelligent Environment Simulation frameworks**

**OpenSHS** [50], is a hybrid, cross-platform 3D smart home simulator. In more detail, it is a research tool for generating datasets for smart homes. This is particularly helpful for applications that need smart home generated data (e.g. a machine learning algorithm) from activities done in a home without the need to capture data from an actual smart home. The framework works using Blender [51], a free and open source 3D creation suite. Users can use blender to design a floor plan and import smart devices from a library to place them into the virtual environments. Then, user can create multiple contexts. Each context is a file containing initial data for the state of each smart device. Users can also position an actor, a 3D model human. They can then start a simulation with a specific context and control the actor to select activities and interact with smart device in order to generate data.



Figure 19: Smart home built and simulated using OPENSIM [50].

The work in [52] introduces **eHomeSimulator**, a software tool for research on software engineering aspects of Intelligent Houses (which they call eHomes), such as service development, specification, composition, deployment, and personalization. It serves two main purposes, a) the testing and evaluation of other tools and services for Intelligent Houses and b) the demonstration of such tools and services. eHomes contain three main elements that can be emulated in the eHomeSimulator, the rooms of the home, the persons, and the smart devices. The tool provides an editor for simulated environments in form of tiles placed next to each other. Tiles are two-layered, one representing the background image of the environment and one containing a code that represents a smart device if a device is placed in the specific tile. Users can add devices, persons and set areas that are inaccessible. For the smart devices, the authors use a service gateway for communication between the simulator and the device services. To connect different services, an action can be mapped to call a function of another service, for example, for a lamp and a switch a user can select that when a switch turns on the call the function of the lamp that turns it on. The simulator allows the user to move the active person inside the simulated environment and interact with the available smart devices. A



control panel shows the current state of the room, allows to select other persons or rooms to view and for the room that is selected it displays the state of the available smart devices and sensors.

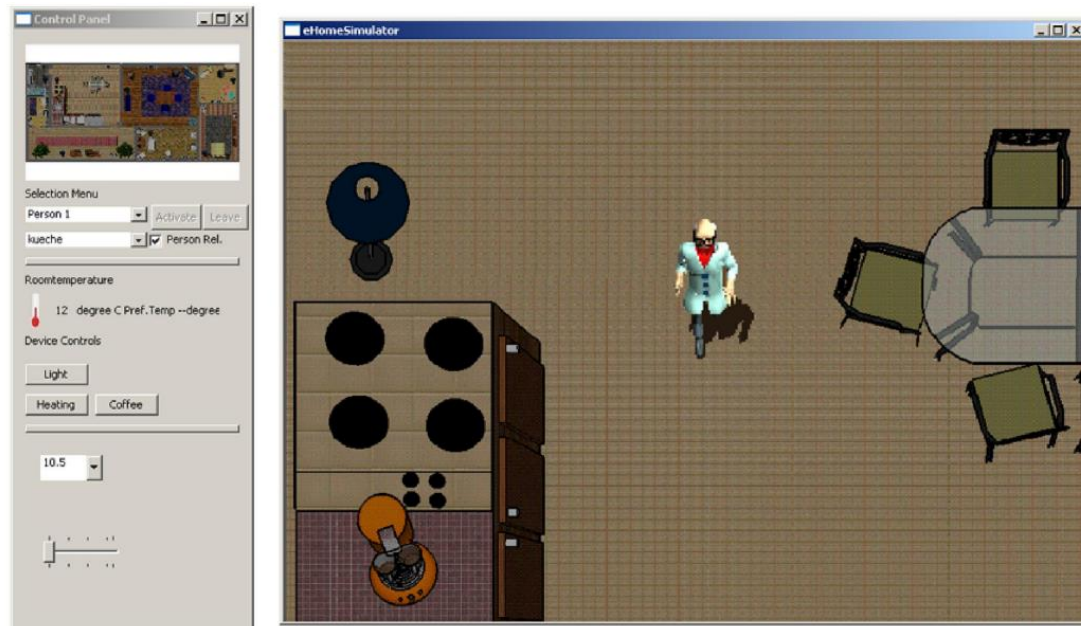


Figure 20: The User Interface of eHomeSimulator [52]

The work in [53] presents a tool for generic simulation suitable for the many scenarios encompassed by ubiquitous computing, such as simulation of sensors, actuators, and the environment. In addition, an emulation framework for middleware and software under development is provided which interfaces with the simulation tool. Authors provide a layered, flexible and modular approach to supporting the simulation of ubiquitous computing environments without constraining the simulator to one aspect of the many possible ubiquitous computing deployment scenarios.

The work in [54] introduces **UbikSim**, a ubiquitous computing environment simulator that aims to alleviate problems when testing services and applications that rely on both physical environments and user behavior. Each user is defined as an agent which acts independently, and the behaviors of the simulated agents are modelled probabilistically. Behaviors can be set to agents

which are situation that the agent should play in each moment. After running a simulation, devices, like for example sensors that respond to a presence of a user, produce information that can be used by other ubiquitous computing systems to be able adapt to changes based on the context of the environment.

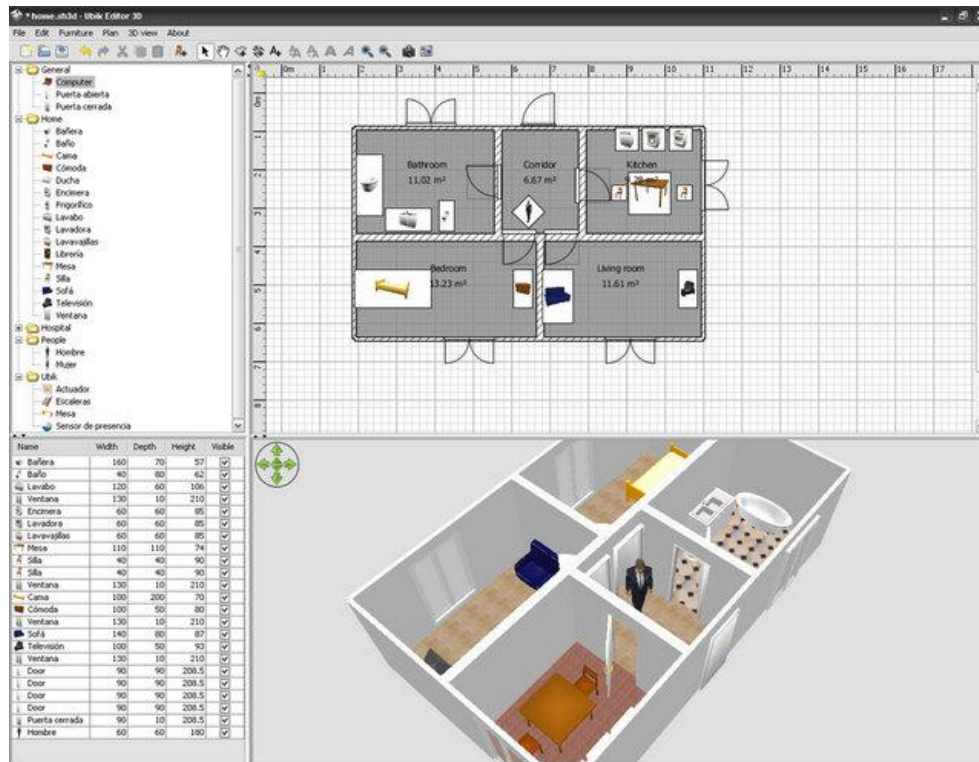


Figure 21: Designing an indoor space in UbikSim [54].

The work in [55] presents **3DSim**, a tool that uses a 3D virtual environment to represent an intelligent space for photo-realistic rapid prototyping of Ambient Intelligence Applications. The tool allows users to integrate a wide range of devices from sensors, SMART Boards, lights, blinds, large display walls, projects to human activity. Based on the usage of an open standardized interface they support the insertion of any UPnP control point can invoke action on virtual devices resulting on state changes such as turning off the lights. This feature allows users of 3DSim to fuse data from real sensors and virtual sensors in order to decide the state changes of the prototype virtual environment. The tool allows to realize an Ambient Intelligent Environment in the early development phases without the need to invest on creating a

physically existing environment. When this work was published, 3DSim had only a meeting room modeled as a virtual environment, but the authors intended to integrate several environments such as classrooms, home environments, and office spaces.

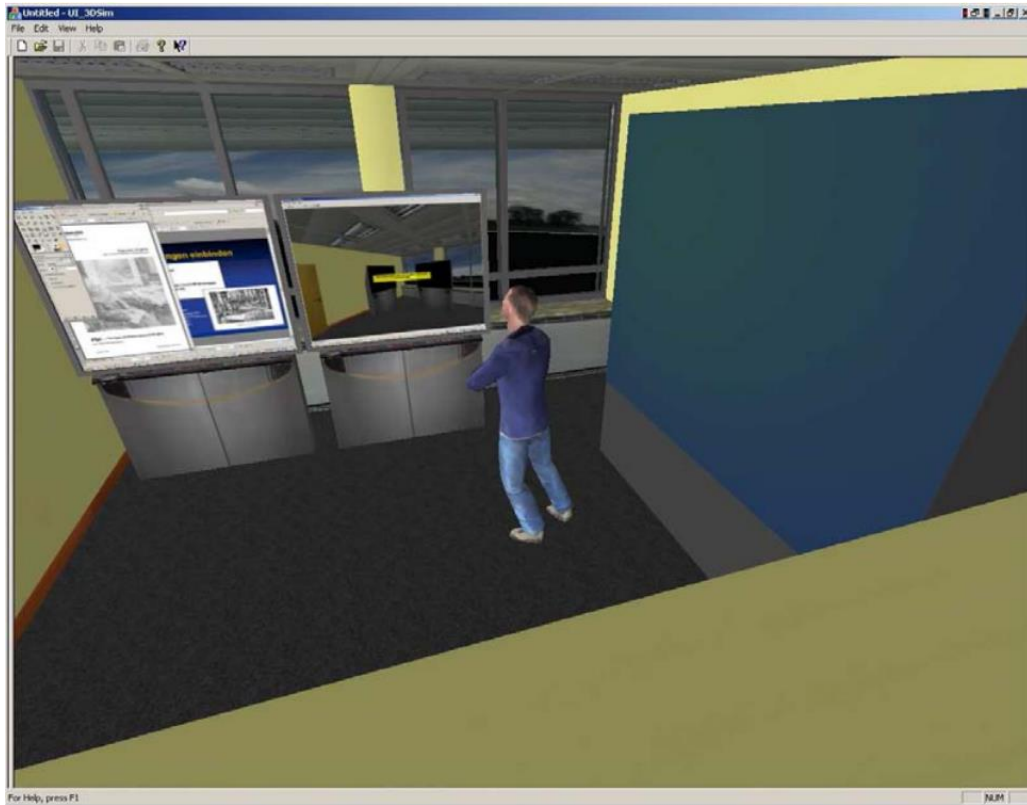


Figure 22: 3DSim created virtual environment features three display and an avatar [55].

### **2.4.1 Summary**

The simulation of environments is an area that has captured the interest of researchers for at least two decades now. The purpose of simulation frameworks is broad. There is an extensive list of tools targeted to simulate ubiquitous environments in order to generate data that otherwise would be impossible to generate, as they can produce data for different situations, for one or more users, in order to use the information generated for context-aware applications or machine learning algorithms that need a big amount of data. Another area of interest that the literature review of simulating frameworks for Intelligent Environments revealed, is to help users to immerse themselves into the environment and understand concepts with ease without the need for

designers and developers to create fully working actual environments which is usually a costly process. An observation emerging from the literature review is that tools which allow the simulation of applications for Intelligent Environments are usually targeted towards developers or technically skilled users.

## **2.5 Discussion**

This literature review helped on identifying some of the requirements that Wizard of AmI should fulfill (described in detail in section 3.4) and provided insights on shortcomings of different approaches (e.g. running a prototype in a simulated Intelligent Environment may be a limiting factor for User Experience assessment).

The literature review reveals a gap in tools for interactive prototyping for Intelligent Environments. While the list of interactive prototyping tools for traditional applications targeted to designers is extensive and well matured, the frameworks targeting Intelligent Environments have several drawbacks. There is no tool that succeeds in providing at the same time (a) ease of use for designers, (b) management of multiple screens and smart devices, (c) support for a wide range of different types of Intelligent Environments, (d) support for all the phases of the design process and (e) a well-rounded way to allow designers and end-users to assess the overall User Experience of the prototypes.

Wizard of AmI's purpose is to fill this gap by taking into consideration features from different interactive prototyping tools for traditional applications as well as the advantages that the different approaches of interactive prototyping for Intelligent Environments have.

Wizard of AmI utilizes features from traditional prototyping tools, such as the collaboration, support for importing mockups in multiple ways as well as the overall look and feel of those tools to achieve a similar experience for a user that is already familiar with them. It utilizes the idea of creating interactive prototype-based storyboards that tools for prototyping for Intelligent

Environments use but in the form of an interactive prototype-based scenario to help visualize better the overall flow of the prototype. It employs the Wizard of Oz technique as it is particularly helpful for Intelligent Environments where a prototype may need to incorporate logic that needs the fusion of data from different sources which is difficult to be modeled. Wizard of Oz can be used to simulate complex logic to overcome the need for either trying to model it, which can be extremely time consuming, or writing code to incorporate logic into a prototype. It also uses the idea of simulating an Intelligent Environment by allowing designers to run their interactive prototype-based scenarios either on a computer through a 2D representation or enact them in the physical Intelligent Environment.



# Chapter 3

## Design Process

This Chapter initially introduces the methodology that was followed, presents several scenarios that motivated this work, describes the co-design workshop that revealed several requirements, and afterwards outlines the final functional and non-functional requirements of the system.

### 3.1 Methodology

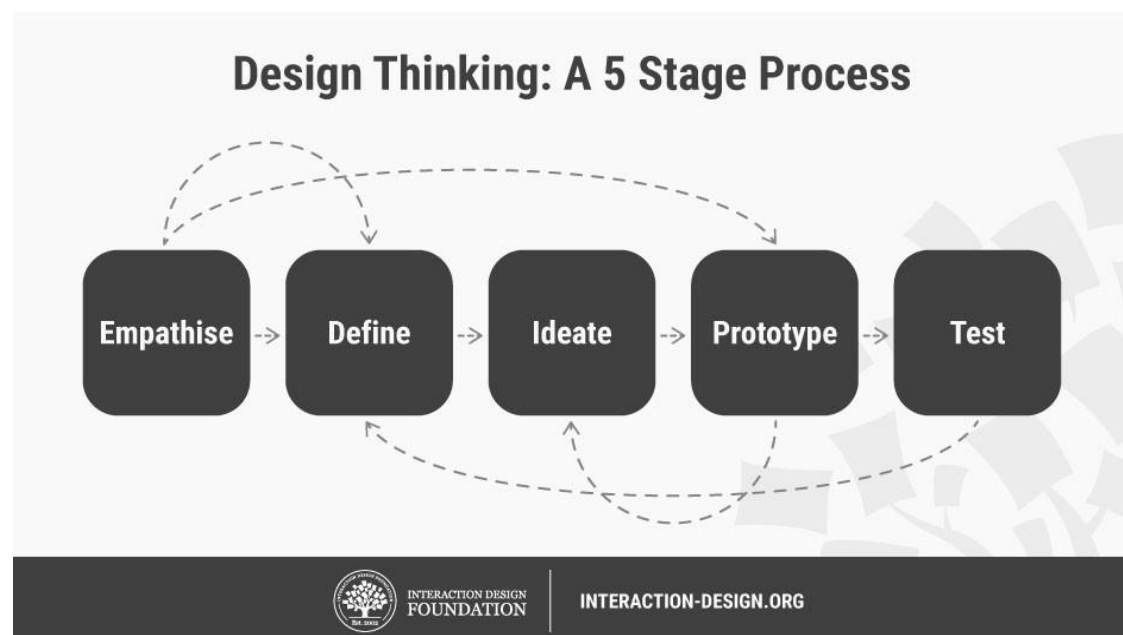


Figure 23: Design Thinking Process [56].

The process followed while designing the system, was the Design Thinking methodology [57]. In a series of meetings with several potential end-users who have previous experience of designing applications for Intelligent Environments, scenarios and personas were created for the ‘Empathize’ and ‘Define’ steps of Design Thinking (Figure 23). Next, the ‘Ideate’ step was

separated into three rounds. For the first round, multiple brainstorming sessions were organized, and produced multiple ideas. For the second round, and after successfully structuring a basic idea about the requirements and important features that such application should fulfil, a co-design workshop was organized, with 12 participants, male and female experienced designers and developers in the field of Ambient Intelligence. After the workshop, the findings of the discussion were analyzed and categorized. For the third round, a series of meetings were organized to filter the ideas that were generated from the two previous rounds. After considerable work and analysis on the ideas that were generated in combination with the insights from the literature review, a preliminary set of requirements were generated. After this, the design team continued with the prototyping phase. At the beginning, low fidelity (Figure 24) prototypes were created in order to visualize the core features of Wizard of AmI. After iterating on low-fidelity prototypes and with the help of experienced interaction designers the requirements were finalized. The team then designed hi-fidelity (Figure 25) interactive prototypes of the system. To test the usability of the Wizard of AmI, a cognitive walkthrough evaluation was conducted with four (4) User Experience (UX) experts. In detail, a scenario was given regarding the creation of an interactive prototype of an application for an intelligent living room that was complex enough to allow the designers to experience multiple features of the Wizard of AmI and uncover any issues that the system may have. After the first cognitive walkthrough, the first version of the system was developed, and the team conducted a second cognitive walkthrough evaluation with five (5) UX experts. Section 6 describes the evaluation process in detail.



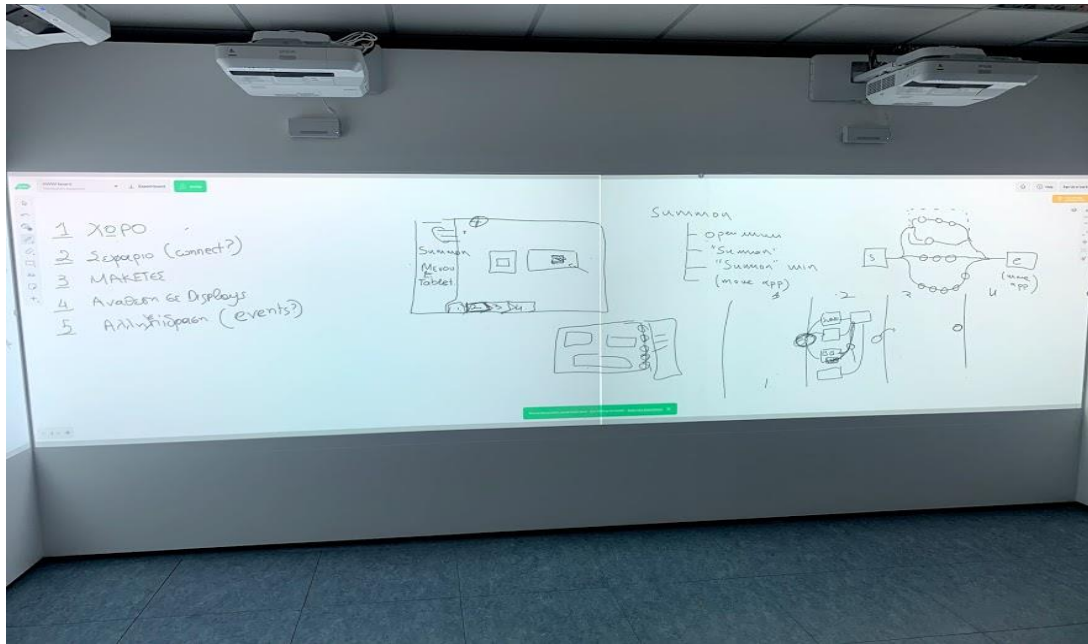


Figure 24: Designing lo-fidelity prototypes on the Wizard of AmI on CogniOS Board [58]

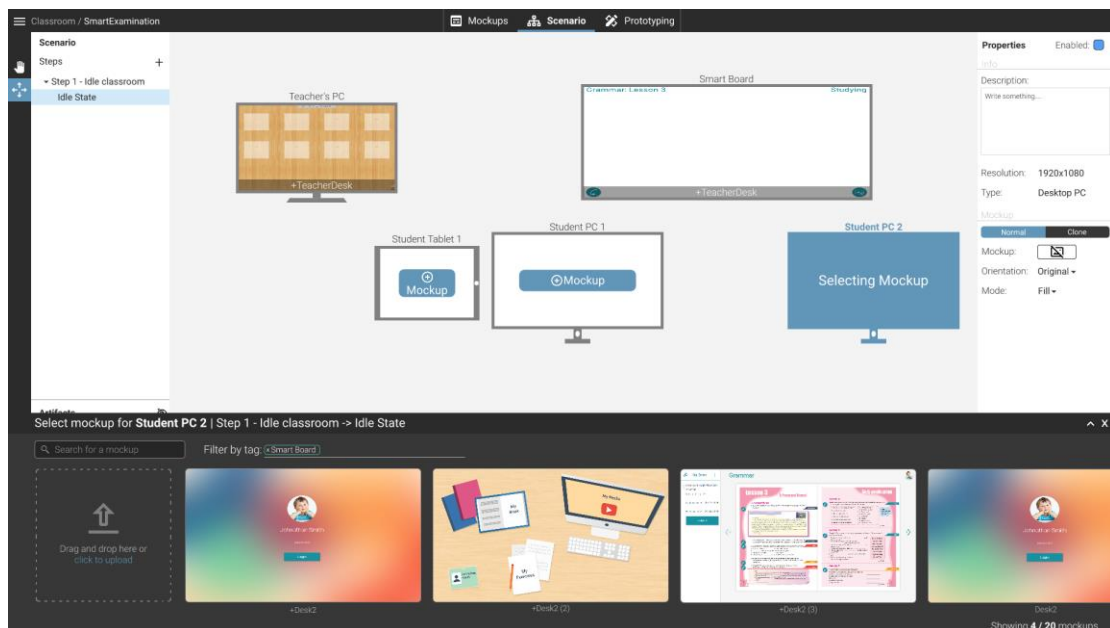


Figure 25: Hi-fidelity prototypes of the Wizard of AmI.

## 3.2 Motivating Scenarios

Scenario building is a widely-used requirements elicitation method [59] that can systematically contribute to the process of developing requirements.

Scenarios are characterizations of users and their tasks in a specified context, which offer concrete representations of a user working with a computer system in order to achieve a particular goal. Their primary objective in the early phases of design is to generate end user requirements and usability aims. The following sections present a collection of envisioned scenarios where Intelligent Environment User Experience Designers use the Wizard of AmI in order to create interactive experiences for their applications.

### ***3.2.1 Interactive Scenarios for eAmIDelivery***

Scotty is a designer at the Technology Company IntelligentX. He is the lead designer of a project called eAmIDelivery, a multi-screen application for ordering food in an Ambient Intelligence Livingroom. The hardware requirements for eAmIDelivery are a set of smart devices (i.e. one smartTable, one large smart tv and multiple mobile phones). For the purpose of this project, IntelligentX has built a testbed intelligent living room containing the devices needed for this project. The design team has already designed the basic mockups for each device, and they want to test them by creating an interactive prototype-based scenario. IntelligentX uses Wizard of AmI as a tool for interactive prototypes in Ambient Intelligent Environments. Scotty, as the lead designer and experienced user of the platform, is responsible for creating the interactive prototype.

Scotty launches the web-based app, from his laptop, and creates a new project. He selects the living room as the space of the project; he adds his colleagues as collaborators and finally configures the space in order to use only the set of available devices of the space that are applicable for the project. He then proceeds to upload the mockups that the team has designed. He drags n drops the folders with the images that are already categorized by device. The system recognizes the folder names and suggests tagging the images to pair them with the appropriate devices, Scotty confirms the suggestion and now the mockups are uploaded and categorized for each device. He then starts creating the scenario. He creates the steps of the scenario based on the mockups already

designed. After that he creates the appropriate connections from one step to another to create the flow of the scenario. In order to validate that this scenario adheres to what the team envisioned, he asks Fiona, the project manager, to check the scenario. As she already is a collaborator to the project, she can view the scenario through a shared link that Scotty sent her. Fiona adds some comments, pointing out that some important steps are missing. Scotty now reads the feedback and proceeds to add the missing steps. After finally having the complete flow of the scenario, he asks Fiona again to verify it, in order to proceed with adding the states (i.e., the different snapshots of the intelligent space for the prototype) that the prototype will have and the transitions between the states. Fiona verifies that indeed the flow is ok, and Scotty now starts creating the states.

Scotty creates the first set of states and sets the needed transitions. He uses the preview mode to verify that everything works as expected and then continues creating the other states by simply copying and pasting the previous states and changing only the mockups for the devices that are affected. After completing the states and transitions that are required, he asks other members of the design team to check the prototype by sharing a link that opens a preview of the prototype which can be run through a browser. Now, he proceeds to run the prototype in the testbed intelligent living room to communicate the idea of the eAmIDelivery application to the whole team.

### ***3.2.2 Designing an Intelligent Classroom***

The company NextEducation is developing a system for the intelligent classroom. Inside their facilities, NextEducation has a testbed classroom, equipped with high-end desks (with displays, speakers, sensors etc.), a large interactive board, and multiple sensors that can measure student attendance and other metrics regarding the teaching process.

Their latest idea is about an application for exams that will take place when the student attendance is low. A small design team of four people, John, Nicki, Gerard and Sofia, are responsible for the whole design process of the project.

They already designed the basic mockups of the system, as each member of the team was responsible for a different device design. They have created personas and they would like to generate a scenario in order to communicate their design ideas to the company. NextEducation recently started using Wizard of AmI to streamline and simplify the design process of their products. The team decides to create a new project in Wizard of AmI to help them visualize the scenario and after that to be able to use it to create an interactive prototype. John creates a new project for the smart classroom, invites his teammates as collaborators and configures the space appropriately to accommodate their project. Each member receives a notification inviting them to collaborate for the project, and they join. Now they are ready to collaborate in order to create a scenario. Each one uploads their mockups for the project and uses the tag system to indicate the devices that each mockup is designed for.

They are ready to create a scenario. John, as the owner of the project starts creating a scenario by adding a generic description. After that, the team brainstorms and creates the steps of the scenario. Then, each member is assigned a set of steps. Each member creates states and select mockups for each device. Now they run the scenario logic step by step to check if it is reasonable and after that, they start connecting the states. For each state, they select interactive areas for the mockups or sensor-based triggers to change from a state to another state.

Now they are ready to run the scenario live at the testbed intelligent classroom in order to show their interactive designs to the company.

Three of the team members play as actors for the scenario. More specifically, Nicki is the Teacher, Gerard and Sofia are two students and John does not play a role in the scenario, but he oversees the scenario to check if everything works as expected. John can also intervene if needed in order to change the progress of the scenario through a companion application through his mobile phone.

### **3.3.3 The Future Kitchen**

EveryDayKitchen is a company located in Athens. Their expertise is full kitchen installations. After years of excellence, they decided to build a futuristic kitchen with a smart kitchen countertop containing a display, interactive walls, and smart appliances such as smart fridge and smart stove. EveryDayKitchen is using Wizard of AmI for their design process, as it is important for them to have a clear and complete idea about their designs before implementing them.

One of their goal is to equip this kitchen with a software that can assist the cooking process. They assigned this project to Annie and Qamar, who are the User Experience designers of the company, to get an idea about how such an application could look like.

Annie and Qamar decided to start their task by creating an appealing scenario for home cooks that will highlight the key features that a system like this could have. Annie starts by creating a new project, using Wizard of AmI, and invites Qamar. She selects the smart kitchen environment and configures it in order to use only the devices they will need for this project (the fridge display, the kitchen countertop screen and the interactive wall in front of the counter). They then proceed to create a scenario. After that, they think about steps of the scenario through a brainstorming session, and they proceed to create them.

As it is the first time that they both design for this kind of application, they first need to create some medium-fidelity designs to view them on site to check if their designs (text size, image size etc.) are appropriate. They create four different designs, and they add four steps in Wizard of AmI in order to view them in the environment. They just added the steps without connecting them. They are now in the actual future kitchen and they run their prototype through the mobile Wizard of AmI application. They choose to view the first step and then browse through the different steps they created to decide which is the

most appropriate one. After viewing the different designs and discussing about them, they decide what design fits best.

They now start designing on top of their selected mockups using their favorite design tool. In parallel, they use Wizard of AmI to create the steps and states that they will use. This helps them to have a better overview of the flow in case that they need to design more mockups to show transitions in order to have a compelling interactive scenario. After creating the mockups, they upload them and add them to each state they created. They then proceed to connect the different states and choose how they will move from one state to another. As the prototype requires actions that they cannot model into Wizard of AmI, as the future kitchen does not contain all the needed equipment yet, they create multiple triggers that change states, such as ‘the user collected all the products for the recipe’ in order to use them when they are running the scenario to provide a more realistic feeling. After finalizing the scenario, they are ready to enact it in the intelligent kitchen.

Annie takes the role of the cook for the scenario and Qamar is the ‘Wizard’ who simulates the needed logic through the Wizard of AmI mobile. Qamar starts the scenario for the kitchen. The environment shows the first state and Annie starts the cooking process. Qamar oversees the process in order to intervene when necessary to trigger the actions that are not possible to be automated such as activity recognition.

### **3.3 Co-Design Workshop**

Co-designing is a relatively new approach while designing applications, where the roles of designers and users get mixed up and stakeholders work together to achieve a design that is closer to what the end-users need [60]. The emphasis is on design with the end-users rather than only designing for end-users. Leveraging ideas that are generated directly from end-users combined with the application designer’s expertise can produce results that can directly impact the usefulness of the final design. To that end, a co-design workshop was conducted for Wizard of AmI, aiming to allow end-users to participate in

the design process while generating helpful ideas for the Wizard of AmI tool. The process of the co-design workshop is presented below.

### ***3.3.1 Preparation***

The co-design workshop was conducted with 12 participants (male and female experienced designers and developers in the field of Ambient Intelligence) and 2 facilitators. The facilitators separated participants into 4 teams of 3 members each, where each team had at least one developer and one designer. Each team was equipped with all the necessary tools (i.e. whiteboard, different sizes of sticky notes, a set of whiteboard pens containing different sizes and colors) to create a journey map.

### ***3.3.2 Co-designing***

After the preparation, the co-design process started. An introduction was made about creating experience prototypes for Intelligent Environments. Each team was assigned to prototype a prototyping tool for Intelligent Environments given some basic requirements (such as that the application should support multiple displays and services). Additionally, the focus was on prototyping the process of creating a prototype for a given Intelligent Environment in the form of a journey map and creating some basic low fidelity prototypes for the important features of the tool. Finally, the spatial configuration and the devices and monitors of the environment has, were given as data. Each team was given an hour and a half to create their prototypes while two (2) observers, which are members of the design team for Wizard of AmI, were present for clarifications in case any team needed help.

### ***3.3.3 Discussion***

Afterwards, each team presented their prototype in front of all the teams in order to discuss about their ideas. Each presentation was about fifteen (15) minutes long and another ten (10) minutes were allocated for discussion. Alongside the presentation, the two observers documented each idea as well as comments that generated from the discussion. The discussion process

turned out to be useful. It generated a plethora of fruitful discussions about certain ideas that were particularly helpful for understanding the needs of end-users and how they imagine such a tool should be. The results of this discussion verified the ideas that have been already discussed during the brainstorming sessions and meetings.



Figure 26: (a), (b) Prototyping during the co-design workshop and (c), (d) two of the four prototypes presented.

## 3.4 Requirements

This section presents the high-level functional and non-functional requirements that Wizard of AmI and the related front-end helping tools (i.e. desktop, tablet, mobile applications) need to satisfy. Such requirements have been collected through an extensive literature review and the described iterative elicitation process based on multiple collection methods, including brainstorming, co-design workshop, and scenario building.



### **3.4.1 High-Level Functional Requirements**

**FR-1:** Users should be able to log in and log out from the platform at any time

**FR-2:** Users should be able to create interactive prototypes for a specific space

**FR-3:** Users should be able to configure the space for their project

- a. Select a subset of devices to use for their project
- b. Rename devices
- c. Group devices together
- d. Change the orientation of a display

**FR-4:** Users should be able to invite other users to collaborate in a project

- a. Multiple users should be able to work in one project concurrently

**FR-5:** Users should be able to explore their projects

**FR-6:** Users should be able to filter and search for a space or a project

**FR-7:** Users should be able to view information about the project

- a. Project name
- b. Space
- c. Collaborators
- d. Total number of mockups
- e. Total number of used devices

**FR-8:** Users should be able to upload mockups to the platform

**FR-9:** Users should be able to view details of the uploaded mockups, edit, and delete them

**FR-10:** Users should be able to add tags on the mockups. Tags can be either a specific device or custom tags named by the user. They should also be able to delete the provided tags

**FR-11:** Users should be able filter and search through mockups based on their name and tags

**FR-12:** Users should be able to either view mockups all together or categorized by tag

**FR-13:** Users should be able to create a scenario based on a selected Intelligent Space

**FR-14:** Scenarios may be for one or more final users

**FR-15:** Users should be able to add a description to the scenario

**FR-16:** Users should be able to decompose a scenario into Steps (i.e. states of the system based on the user's perceptive)

**FR-17:** Users should be able to create steps and connect them -by simply drawing arrows- in any way they like

**FR-18:** Users should be able to group steps together to form Tasks (i.e. A set of steps that can be managed one separate scenario that has a starting step, a flow with steps and an ending step)

**FR-19:** User should be able to name each step and write a short description

**FR-20:** For each step they should be able to create states (i.e. snapshots of the application where each display contains a specific mockup, and each smart device has a given state)

**FR-21:** For each state users should be able to set specific mockups for each display

**FR-22:** When adding a mockup, users should be able to select a mockup based on the tag that they may added for the mockups

**FR-23:** Users should be able to copy, cut and paste states

**FR-24:** Users should be able to set the transitions between steps using events

**FR-25:** Users should be able to select interaction areas on mockups to act as triggers in order to change states

**FR-26:** Information about missing connections and steps which are not connected with other steps should be available to the user

**FR-27:** Users should be informed for incomplete state (i.e. missing mockups for devices)

**FR-28:** Users should be able to preview their scenario while creating it

**FR-29:** The preview mode should allow users to interact with devices through a virtual representation of the space

**FR-30:** The preview mode should allow users to easily browse through the scenario

**FR-31:** Users should be able to execute a created scenario, in the physical environment they selected as a space for the project

**FR-32:** Users should be able to control the scenario through a companion application running on mobile phones or tablets

**FR-33:** In the companion app users should be able to start and stop the execution and select specific task, step, or state to execute.

### ***3.4.2 Non-Functional Requirements***

#### **NFR-1: Acceptance Testing Requirements**

A full-scale user-based evaluation should be carried out to ensure designers' acceptance. Participants should have varying levels of experience using web or mobile applications. The overall score of a Standard Usability Scale (SUS) based post-evaluation questionnaire should be above 75%.

#### **NFR-2: Documentation Requirements**

A Quick Start Guide should be provided.

#### **NFR-3: Learning Tools Requirements**

Wizard of AmI should provide means to help a user to learn how to use the platform.

#### **NFR-4: Platform Compatibility Requirements**

From a user perspective, the main application should be accessible through a web page from a personal computer regardless the running OS and the execution companion app should be accessible through a mobile device.

#### **NFR-5: Deployment Requirements**

Wizard of AmI should minimize any deployment requirements for its core components. During system's deployment the installed applications should not be affected, and vice versa.



# Chapter 4

## Wizard of AmI

Wizard of AmI is an interactive prototyping tool for Intelligent Environments and more specifically a tool for prototyping applications that use multiple displays and sensors installed in an Intelligent Environment. It aims to simplify the design process of applications for such environments by allowing users to create interactive prototypes in the form of scenarios and test them in the physical environment. Wizard of AmI (i) offers users a digital representation of the space, which can be configured based on the requirements of the system that the user designs, (ii) equips users with a sophisticated mockup directory with version control, where users can upload and categorize their mockups, (iii) provides a practical User Interface to create scenarios, in a graph form, where they can define steps, and add mockups to create an interactive prototype-based scenario, (iv) features a preview mode where users can preview their interactive prototypes on a 2D representation of the space, and (v) offers users the ability to view their prototypes on the respective physical Intelligent Environment, and supports interactivity through a Wizard of Oz companion application for tablets and Smartphones.

### 4.1 Challenges

An important question to ask here is, why is prototyping Intelligent Environment applications different from prototyping for traditional applications? Why not use an existing prototyping tool? Existing prototyping tools cannot be used for Intelligent Environments because a prototype for such environments has unique characteristics which generate challenges that

a tool should handle and are not addressed by traditional prototyping tools. Such challenges, which were encountered while designing Wizard of AmI, are described below.

**Challenge A.** When designing for Intelligent Environments, each display of a space is a potential candidate to communicate with the user (i.e. present information, ask for input). Designers should be aware about the characteristics of each display and its location in the environment so as to make various decisions. To this end how can this information be communicated to designers? What is the most appropriate way to present a physical space, giving a spatial sense to users?

**Challenge B.** Considering that when designing for an Intelligent Environment, designers should be able to define the behavior of devices other than displays (e.g. smart lamps), what is the most appropriate way (intuitive and practical) to incorporate this functionality, without complicating the design process?

**Challenge C.** Considering that when prototyping for multiple devices, the number of transitions and states of the prototype can be enormous and complex, how can all these transitions and states be modeled to make it easy for users to create prototypes?

**Challenge D.** Considering that interactive prototyping tools are already widely used in the design process for traditional applications, how can the process of prototyping for Intelligent Environments can be simplified without significantly altering the practices that designers already follow?

**Challenge E.** Considering that interactive prototypes for traditional applications can be interactive paper prototypes with proven processes for creating and testing them, as they are cost effective and successful for the early design stages. How can a tool support interactive paper prototypes for Intelligent Environments, in an uncomplicated manner?

## 4.2 The Wizard of AmI Core Concepts

Given the aforementioned challenges, the design team focused on creating a sophisticated prototyping tool - appropriate for Intelligent Environments - that supports the design process of applications for such environments. In order to better describe the system and the solutions for the challenges, below are described the core concepts that Wizard of AmI utilizes.

1. **Space.** The term space refers to any Intelligent Environment that a user can select to prototype for, through Wizard of AmI. A space can contain devices, ranging from displays to other intelligent artifacts such as smart lamps, or user location service.
2. **Device.** A device is any smart artifact or service of an Intelligent Environment. In more details, a device can be any type of Display that is part of such environment and can be used in a prototype to show a mockup and even allow user to interact with it (e.g. touch screen). Additionally, a device can be a programmable smart artifact like a smart lamp or smart blinds. Finally, contextual services, such as service that reports the location of the user are also referred as devices, as they are used in the system in the same way that smart devices are used. Devices can have states (for example a smart lamp can have the states: lamp is turned on and lamp is turned off) and triggers (for example a smart blind can have: open a blind and close a blind).
3. **Mockup.** A mockup is an image that represents a state of a User Interface for a specific device in a prototype. Mockups can range from simple layouts to hi-fidelity images that look like a screenshot from a running application.
4. **Interactive Area.** A mockup in Wizard of AmI can contain interactive areas. Interactive areas are hotspots that can be used, while creating a prototype, to add interactivity. In more detail, interactive areas can be set to act as trigger to transit to another state of the prototype.
5. **Step.** A scenario can be split into steps. Steps are the different statuses that a prototype can have. Step is the status of the system from the point of view

of an end-user. More specifically, it is either an initial status that the prototype may have or a status that may be the result of an implicit or explicit action of the user. A step may contain alternative representations. For example, in a prototype that supports multiple users, one step is 'when one user is logged in'. This step can have multiple representations depending on the user who is logged in. A scenario is divided into steps. The way that the scenario will be divided is at the designer's discretion.

6. **State.** The state is a specific representation of the prototype in the space at a given time where each screen shows a specific mockup, and each device has a certain status. States are all the different snapshots a prototype can have.
7. **Connection.** Connections define the flow of the prototype. More specifically, a connection connects steps. The connections can be unidirectional or bidirectional. A connection can contains events. Events are defined as implicit or explicit interactions from a user that result to a state change. An event defines how the transition from one state to another step can be made. That is, when an action occurs in the system, while it is in a specific state, then in which state the prototype will change. An action can be either defined for a specific interactive area in which the user will interact, or an implicit interaction from a service, or generally defined as a transition that can be activated but has not specifically modeled how it will be triggered.

## 4.3 Key Features

### 4.3.1 Mockup management

To create an interactive prototype, a designer has to first create mockups. Wizard of AmI provides users with a mockup management system where users can upload their mockups or import them directly from other design tools



(such as Figma and Adobe XD). The mockups can be organized based on the device that they are designed for or based on user defined categories (for example one category for each step of the scenario). Regarding the fidelity of the mockups, designers can use mockups ranging for low-fidelity or even scanned paper prototypes to high-fidelity mockups. The whole process of creating an interactive prototype is not affected from the fidelity of the prototypes as Wizard of AmI offers the same simplified yet effective process for creating and running interactive prototypes for Intelligent Environments. Challenge E is addressed by providing the same uncomplicated process for creating interactive prototypes regardless if they are composed from paper prototypes or high-fidelity prototypes.

#### ***4.3.2 Interactive prototyping through scenarios***

Users can create interactive prototypes in a form of a scenario. Creating interactive prototypes for Intelligent Environments is challenging, as multiple displays and contextual data must be orchestrated to work together. In order to address Challenge C, Wizard of AmI introduces the idea of interactive prototype-based scenarios as the way of creating Interactive prototypes. It provides a practical User Interface to create scenarios, in a graph form, where they can define steps. This is done in high level, without the need to set specific mockups and deal with the overall management of each device of a space. For each step the user can then set the specific states (i.e. set the status of the different services and mockups for each device). This separation of the flow of the prototype from the setup of the states allows to reduce the workload of the designer. Separating the process of creating the flow and the specific states also addresses the Challenge B, as for each state users can set the status of the devices in each state and deal with the behavior (i.e. state changes) while setting up the flow of the prototype.

### ***4.3.3 Previewing the Scenario***

Designers can preview their scenarios through a preview functionality. While creating a scenario, designers can open a preview window that simulates the Intelligent Environment in a 2D representation of the space where they can interact with the different devices and experience their scenario. This feature provides designers with an easy way to test interactions and validate that the flow of their scenario is correct while creating it which saves time as otherwise, they would have to enact it in the physical environment.

### ***4.3.4 Collaboration***

The design process is a collaborative process. It includes designers, project managers, testers, users, and the majority of the stakeholders of a project. Collaboration is a key feature for prototyping tools which drives teams to a more refined result. As the literature review strongly suggests, collaboration helps teams achieve their goals faster and create better prototypes through valuable feedback and iterations. Therefore, Wizard of AmI supports collaboration extensively. There are two ways for teams to collaborate in real time.

- A. Users can collaborate concurrently on the same project. In detail, all users that are collaborators in a project can upload mockups which become instantly available to the other members and they can add interactive areas on mockups together. Additionally, they can use the scenario workspace and work together on creating their prototype. A chat functionality is also available for each project where users can communicate.
- B. Users can also collaborate with other members by sharing the preview of their prototypes which allows other users for immediate input on the prototype by adding comments.

#### ***4.3.5 Experiencing Prototypes in Intelligent Environments***

Designing a User Interface of an Application for Intelligent Environments can be challenging. The design process differs from a mobile or desktop application because designers have to deal with multiple displays and implicit interactions that occur in such environments. Designers have to focus more on the experience based on the spatial configuration of the space and is important to be able to visualize the entire application through prototypes while designing it. Existing prototyping tools usually only offer a way to view a prototype for applications targeting one device. While there are some efforts for prototyping tools for Intelligent Environments, they are only focusing on the early stage of prototyping or they offer solutions that need programming skills. To that end, Wizard of AmI offers a way for designers to test their creations, at any stage of the design process, through prototyping on already developed ambient facilities without the need to write any code. This allows designers to perceive how their design uses the ambient facilities and how it is impacted by them. A user can import their mockups, which can even be wireframes, and effortlessly create a prototype that can be tested in the physical environment.

#### ***4.3.6 Wizard of Oz***

An application in an Intelligent Environment is usually heavily based on contextual data. Explicit or implicit actions of the user can result on state changes which can be unexpected and are difficult to model when prototyping. To overcome this, prototypes should incorporate logic that would require users to write code, which can be either scripts or visual programming. This approach would require from users to have some programming knowledge, which from a designer perspective, is not desirable and would make the process of creating prototypes more difficult and tedious. In this respect, Wizard of AmI uses the Wizard of Oz technique, which allows users to play the role of a 'wizard' and simulate the application's intelligence.

The extent to which the Wizard of Oz technique is used, depends on the level of modeling that the user achieves while building the prototype. Based on the extensive work of [49], Wizard of AmI follows the same categorization regarding the level of responsibility of the 'wizard', altered to fit into the concept of the design process, and namely the categories are (A) Wizard as controller, (B) Wizard as supervisor, and (C) Wizard as moderator.

**A. Wizard as controller.** Wizard plays the role of a controller and is responsible completely for the flow of the prototype while running. In other words, the 'wizard' simulates the system's response and decides the outcome of user inputs. This type of role is particularly helpful in the early stages of design.

**B. Wizard as supervisor.** The supervisor role is responsible for overseeing the execution of the application and intervene and change the flow of the prototype if needed. This is usually the case when a prototype is modeled thoroughly, and the role of the wizard will be probably minimal.

**C. Wizard as moderator.** The moderator role is between the controller and the supervisor. In detail, this role is responsible for overseeing the execution of the application and control the flow of the prototype in cases that an interaction is not modeled. This is particularly helpful, when the application relies on context that is not currently available in the Intelligent Environment. For example, a prototype that relies on user location while the current setup of the space does not support it, the wizard can simulate the location of the user through modeled events while the rest of the application that relies on explicit user input or other contextual data that are available in the space run without the need of the 'wizard'.

These roles are conceptual and show the spectrum of the possibilities of the 'wizard'. Wizard of AmI provides users with a mobile application that contains a sophisticated Wizard of Oz User Interface, that depending on the modeling level of the prototype can be used from users playing any of the described roles.

## 4.4 The Wizard of AmI Approach

Wizard of AmI is designed in a way that decomposes the problem into 4 different sequential processes that a user should follow and introduces a flow that allows designers to successfully create prototypes in an organized yet agile manner. After the requirement analysis, and brainstorming sessions with potential end-users while iterating on prototypes, the process of creating a prototype in Wizard of AmI was split into the following processes: setting up a project, importing and organizing mockups, creating a prototype based on a scenario, and running a prototype. It is worth noting here that Wizard of AmI allows designers to follow their own process while creating a prototype. In detail, they do not need to follow the aforementioned process exactly as described. Wizard of AmI allows users to perform actions that refer to a specific process at different points of creating a prototype. For example, a user may not use the mockup directory at all and start creating a scenario while adding mockups directly to the scenario from their own file system. This approach is addressing the Challenge D, as the proposed approach is similar to what the designers are already familiar with and is not limiting to a specific flow for creating interactive prototypes, thus allowing designers to follow their preferred sequence.

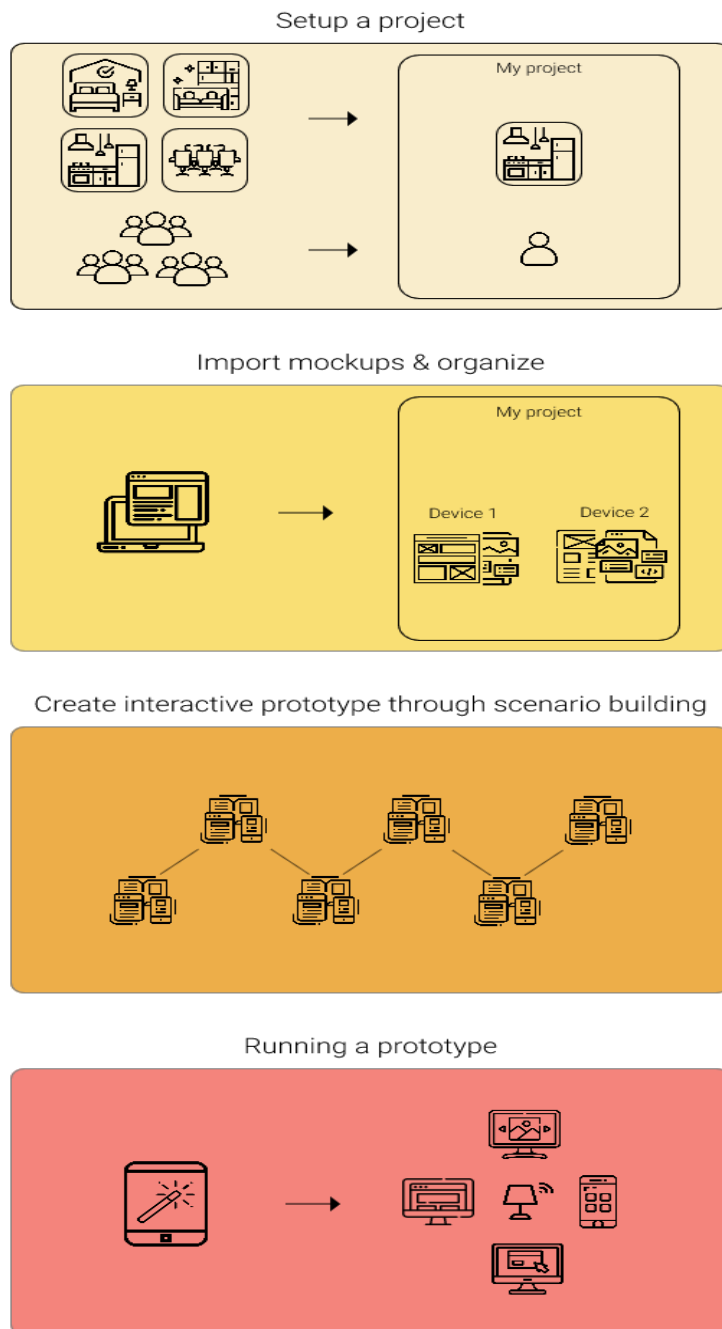


Figure 27: Wizard of AmI approach

#### 4.4.1 Setting up a project

Wizard of AmI allows users to create projects. A project is essentially an entity that refers to a prototype that a user would like to create for a specific system that is expected to run on a given space. While setting up a project, users select

the name, the space, and can also invite other users to collaborate on the project. Users can then view a 2D representation of the space, and configure it based on the requirements of the prototype they wish to create. In more detail, users can enable or disable the devices from a space, which is particularly helpful when a prototype requires less devices than the space offers, they can rename the devices, and change their orientation in case that their prototype uses a device in an orientation different from what currently is configured to be (e.g. a tablet that is configured in landscape for a space, may be used in portrait). Moreover, users can group devices together in order to create groups. Groups are important as a feature for prototypes which use multiple devices that usually change their states simultaneously and can help users when prototyping.

#### ***4.4.2. Importing and organizing mockups***

Users can directly upload or import their mockups from popular design tools, like Figma or Adobe XD, in their prototype in the mockup workspace and organize them. Wizard of AmI uses a tag system that helps with the organization process. In more detail, users can add tags that correspond to devices of the space or create their own tags. Moreover, the mockup directory offers a version control system that can keep track of a mockup version (i.e. history of all the versions of a mockup). Additionally, users can edit metadata of their uploaded mockups, such as name, and description. Finally, users can add interactive areas to mockups.

#### ***4.4.3. Creating a prototype through scenario***

Scenario workspace is the core component of Wizard of AmI. It provides an intuitive and innovative way for users to build their prototypes for Intelligent Environments. Wizard of AmI utilizes the concepts presented previously through a graph based, point and click UI in the form of a never-ending whiteboard where users can add steps, create states for the steps, and add connections between the steps to create a scenario. In order to address the

challenge A, when configuring a state, user can view a 2D representation of the space where they can assign mockups on the devices of the space. While creating a prototype, users can use the preview mode to run their prototypes in order to verify that everything works as expected. Preview mode simulates the Intelligent Environment in a 2D view where users can interact with the devices.

#### ***4.4.4. Running a prototype***

Wizard of AmI allows users to run their prototypes in the physical space that a prototype is designed for. Wizard of AmI is entirely responsible for managing the infrastructure behind the Intelligent Environments, allowing designers to focus only on the process of running the prototype without having to deal with any technical concerns. Users can use the Wizard of AmI companion app, available for smartphones and tablets, to simulate interaction and to advance the scenario manually. This way, end-users can have a realistic experience.

## **4.5 Interactive prototypes with and without Wizard of AmI**

Interactive prototyping for Intelligent Environments combines multiple devices and mockups. Wizard of AmI's approach simplifies the overall process and allows designers to spend less time on the process of setting up an interactive prototype and focus more on improving the overall user experience that their system provide. Without Wizard of AmI, designers should either manually setup each device with the appropriate mockups and navigate through them for each device manually, without creating any interactivity or by programming the Interaction which requires the designer to have the appropriate programming skills or collaborate with a developer to implement the needed logic. Below are described the different processes that a designer would have to follow with and without Wizard of AmI.



### 4.5.1 Building an Interactive Prototype without having programming skills

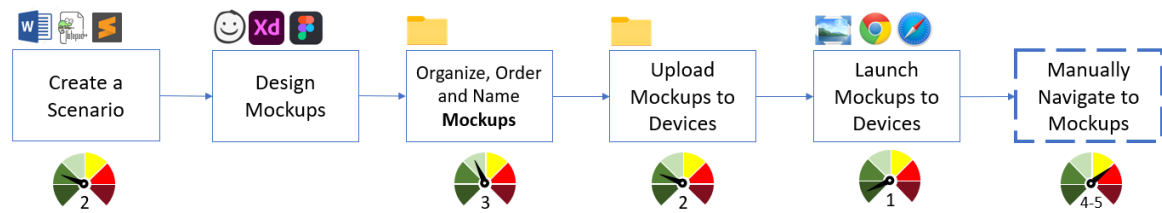


Figure 28: The process for creating an Interactive prototype for Intelligent Environments without the help from a developer

The process that a designer would typically follow for the creation of an interactive prototype for an Intelligent Environment (Figure 28) if they do not have any programming skills and without the help of a developer is as follows:

- A. The designer would create a scenario to understand the ideate and understand the user needs on either a text document or using any available scenario creation tool
- B. They would design the different mockups for their prototype using a design tool such as Figma and Adobe XD
- C. They would organize their mockups in folders, rename them and order them to create the appropriate flow
- D. They would either upload their mockups to an online image viewer to be accessible on other devices or transfer them on the devices that they would like to display them
- E. They would setup each device by either opening on a browser the appropriate online image viewer, containing the mockups for each device or in case that the mockups are stored in each device, by opening an image viewer
- F. Finally, in order to simulate the flow of their prototypes they would have to manually change the mockups displayed on each device.

### 4.5.2 Building an Interactive Prototype with help from a developer

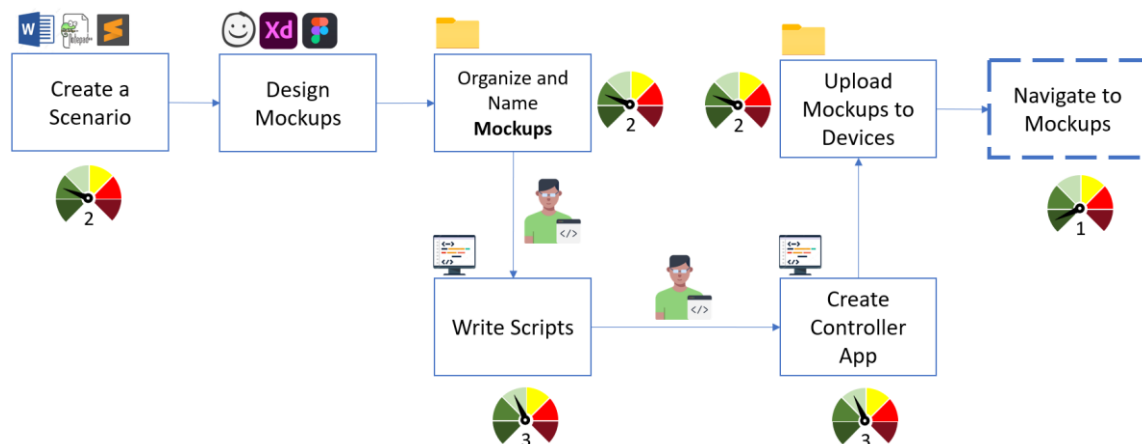


Figure 29: The process for creating an Interactive prototype for Intelligent Environments with help from a developer

The process that a designer would typically follow to create an interactive prototype for an Intelligent Environment (Figure 29) by including a developer, in detail the process is as follows:

- A. The designer would create a scenario to understand the ideate and understand the user needs on either a text document or using any available scenario creation tool
- B. They would design the different mockups for their prototype using a design tool such as Figma and Adobe XD
- C. They would organize their mockups in folders and name them appropriately
- D. A developer would write scripts that incorporate the logic of the interactive prototype to create the appropriate flow
- E. A developer would build a controller application to allow the designer to control the flow of the prototype
- F. The mockups would either be uploaded to each device or they would be managed through the application that the developer built and they would have to setup the devices of the space appropriately to connect to the controller application

G. Finally, to experience their prototypes by using the controller application to manage the flow of the prototype

### 4.5.3 Building an Interactive Prototype with Wizard of AmI

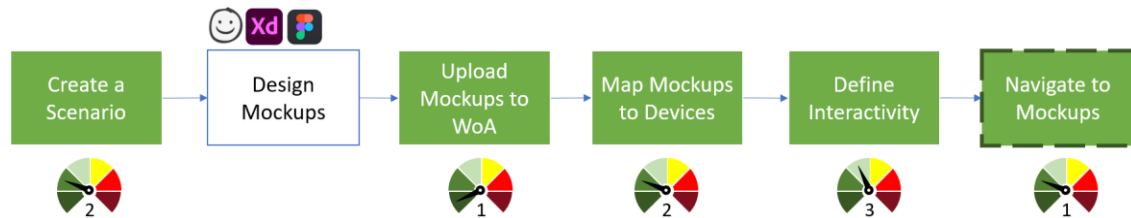


Figure 30: The process for creating an Interactive prototype for Intelligent Environments with Wizard of AmI

The process that a designer would typically follow to create an interactive prototype for an Intelligent Environment (Figure 30) by using Wizard of AmI as follows:

- A. The designer would create a scenario to understand the ideate and understand the user needs using the Wizard of AmI
- B. They would design the different mockups for their prototype using a design tool such as Figma and Adobe XD
- C. They would upload their mockups to Wizard of AmI
- D. They would map the mockups to the appropriate devices of the space
- E. They would define the interactivity of their prototype
- F. They would use Wizard of AmI companion app and navigate through the created scenario.

### 4.5.4 Discussion

The first two steps that a designer would have to follow are the same for any of the above workflows for creating an interactive prototype. The process of creating a scenario is the same either while using Wizard of AmI or any other tool, but the benefit from using Wizard of AmI for the creation of the different steps of a scenario is that it can be used later in the process and simplify the overall process. The process of designing mockup is done through design tools

that are widely used by designers and each design can use their favorite design tools to create mockups. From the third step onwards, the process is simplified when using the Wizard of AmI. In the case that a designer is not using Wizard of AmI, they would have to organize their mockup in their file system either by the devices that the mockups are intended to be used for or based on the scenario steps. They then would have to find a way to add those mockups to the actual devices that they would use, or they would have to include a developer in the process who would write scripts and create an application for managing the flow of the prototype. In contrast to that, when a designer uses wizard of AmI, they can just upload their mockups on the platform and organize them using the tag system which maps each mockup to a device. The process that is needed to do these two steps in Wizard of AmI would be less time consuming compared to the other processes without Wizard of AmI. After that, they would have to add interactivity. Without programming skills or the help from a developer, they would not be able to add interactivity. Comparing the Wizard of AmI approach on adding interactivity and a developer writing scripts and creating a controller application is clear that there are multiple advantages when using Wizard of AmI. There is no need to include a developer in the process or have prior programming knowledge and the time needed is less. Rather than spending time on writing code to manage the flow of the prototype and developing a controller application, the users focus on setting up the flow in a non-technical manner while they can use the provided companion application to control their prototype. The final step is to run and navigate through the prototypes to assess the User Experience of their application. With the first approach, they would have to manually change mockups on each device as there is no interactivity implemented for their prototype, which would be extremely time consuming. In contrast to that, navigating through interactive prototype, by following the second or third approach, is as simple as interacting with the devices or by using the controller application or the companion application respectively to control the flow.

## 4.6 Summary

This chapter described the challenges that differentiate interactive prototyping tools for Intelligent Environments from traditional interactive prototyping tools, and introduced the Wizard of AmI, its key features, and the approach that the system follows as well as a comparison between current design processes and the process using Wizard of AmI that a designer would follow to create an interactive prototype for Intelligent Environments. Below, is a table that summarizes the way that the system addresses the challenges described in 4.1.

Table 2 Summarize of the way that each challenge was addressed

Challenge	Solution
A	When configuring a state in space manager, users can view a 2D representation of the space where they can assign mockups on the devices of the space (Section 4.4.3)
B	For each state, users can set the status of the devices and deal with the behavior (i.e. state changes) while setting up the flow of the prototype (Section 4.4.2)
C	Interactive prototype-based scenarios are introduced as the way of creating Interactive prototypes (Section 4.4.2)
D	Wizard of AmI approach is similar to what the designers are already familiar with and is not limiting to a specific flow for creating interactive prototypes, thus allowing designers to follow their preferred sequence (Section 4.4)
E	The whole process of creating an interactive prototype is not affected from the fidelity of the prototypes as Wizard of AmI offers the same simplified yet effective process for creating and running interactive prototypes for Intelligent Environments. (Section 4.3.1)



# Chapter 5

## Wizard of AmI

## Implementation

Wizard of AmI is implemented using the latest web-based technologies and techniques. It offers a browser-based application for desktops that utilizes the possibilities of a browser extensively to provide an experience similar to a native desktop application. Wizard of AmI uses a microservice architecture, where the different components work autonomously, encapsulating the complexity of the functionality of each component, but together form the functionality of Wizard of AmI. The User Interface provides a plethora of features described in detail in the previous section, that allow users to achieve their goal of creating prototypes efficiently. To support the User Interface functionality, Wizard of AmI uses multiple components that facilitate the features that the system offers.

### 5.1 User Interface

Wizard of AmI is a browser-based application. It is available for use through any modern web browser. The following section presents the User Interface (UI) of Wizard of AmI and describes the functionality that the UI provides.

#### *5.1.1 Homepage*

After logging in, the user is welcomed to the home page (Figure 31). The home page contains the projects that either the user has created or is a collaborator for. The list of the projects can be filtered and sorted. A search functionality is

available to search for a specific project. From the top right plus button, a user can start the process of creating a new project.

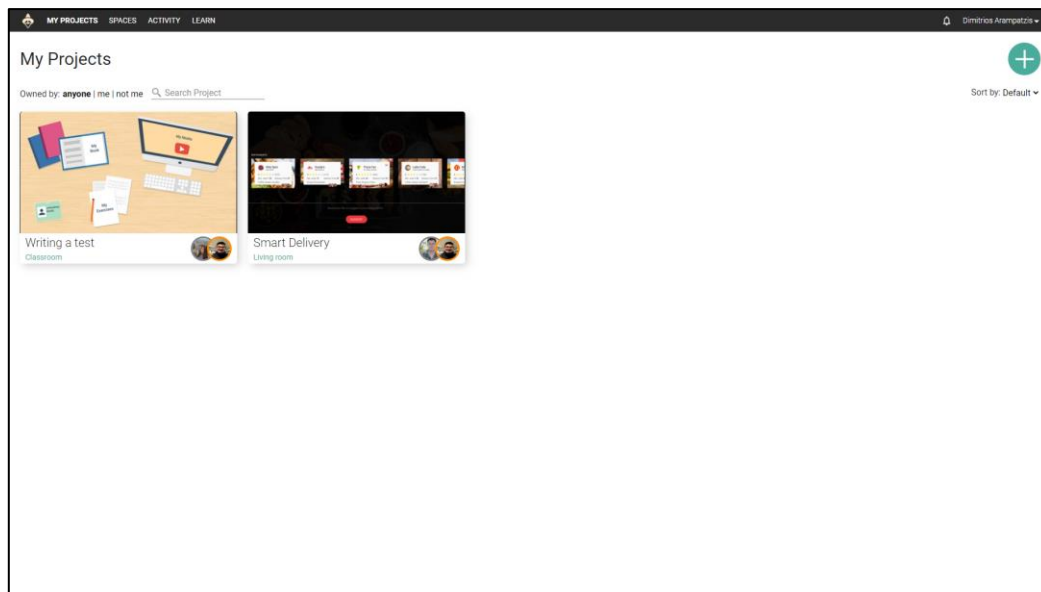


Figure 31: Homepage

### 5.1.2 New project page

The creation of a new project is divided into two pages. In the first page (Figure 32), the user can set the name of the project, select the space that the project will use and add collaborators. When the user chooses to select a space, a window slides from the bottom (Figure 33), showing the available spaces. Each space contains an image and basic information (such as the number of devices that a space provides). The list of spaces can be filter in order to show only the spaces that the user has already used in other projects. Moreover, a search functionality is available to search a space by name. The space selection window can be maximized, in case that the user prefers to browse the spaces in full screen. Users can add collaborators through search either by name or email. When a user writes in the collaborator input box, a dropdown window appears, showing the users that the input matches (Figure 34). The second page is the configuration of the space (Figure 35). This page is divided into three panels. On the left is a panel showing a list of the devices of the space, on the center is the space map, a 2D representation of the space, and on the right is the inspector, a panel that shows information for a selected device. The



user can either select a device from the list on the left panel or from the space map. When a device is selected, the inspector panel shows the orientation of the device and information about it (i.e. resolution, type, and actual size). From the inspector window, a user can change the orientation of the device and write a description.

### ***Space configuration functionality***

The 2D space map is interactive. In detail, users can pan around the map, zoom in and out and select devices. This functionality is available through dedicated buttons in the bottom right corner of the map. Additionally, user can use a combination of their keyboards and mouse to do the same actions. Users can use the mouse middle click or the SPACE key to enable pan mode. Scrolling up and down moves the map accordingly. If the SHIFT key is pressed down, the scrolling wheel works as left and right scroll and if the CTRL key is pressed down the scroll wheel works as zoom in and out.

Users can rename the devices of the space. This can be done with multiple ways. They can right click with their mouse on the device from the list or the space map and select rename from a menu that opens. An alternative way is to double click the name of the device from the list or select the device and press the shortcut CTRL + R from their keyboard.

Users can enable or disable a device. They can right click on the device either in the device list in the left or the device on the map and select the Enable/Disable option from the menu. Another way is from the inspector, through a toggle button that enables and disables a device. Finally, the shortcut CTRL + E can be used when a device is selected to enable or disable it. When a device is disabled, an indicator appears next to the device's name in the device list showing that it is disabled and the device in the space map fades. Users can group devices together. To group devices, users have to select two or more devices. To select more than one devices, users can first select one device and by pressing the CTRL key select another device. After selecting multiple devices, users can group them either from the inspector by clicking

the 'group devices' button or by right clicking one of the selected devices and select the option 'group devices' from the menu or by using the shortcut CTRL+G. The same process is used to ungroup devices.

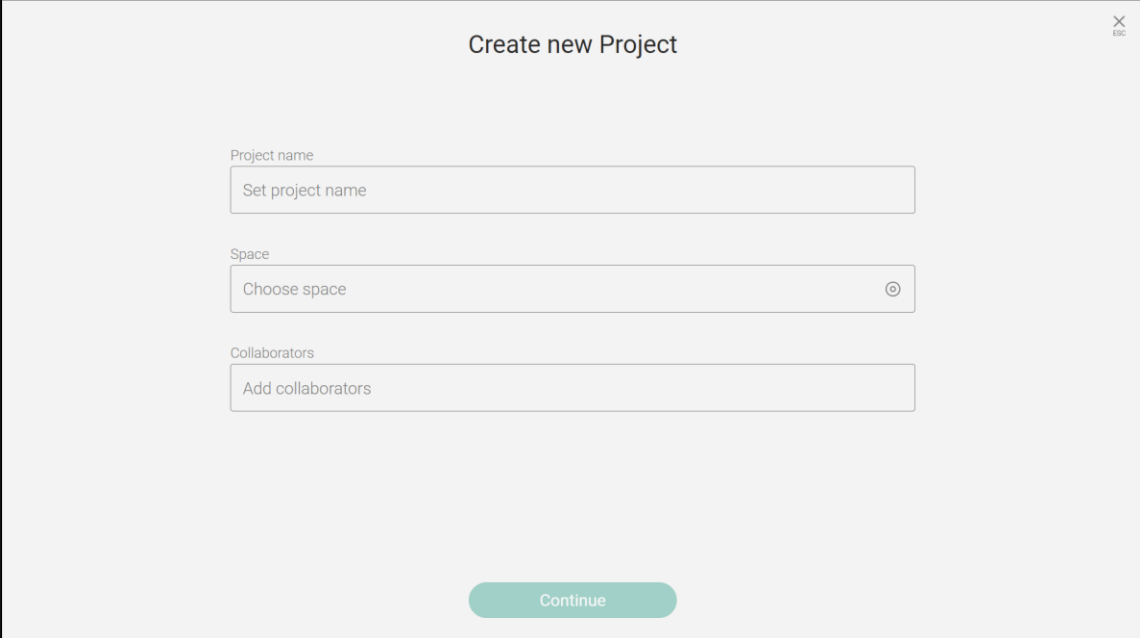


Figure 32: New project (First page)

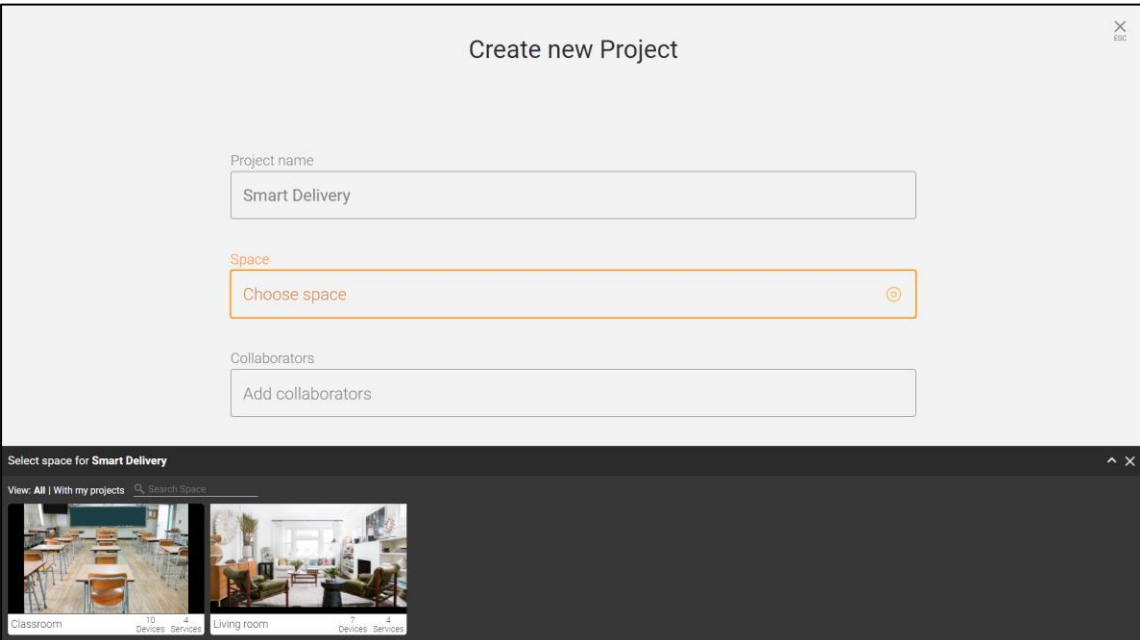


Figure 33: Selecting a space

Create new Project

Project name  
Smart Delivery

Space  
Living room

Collaborators  
cm  
cmanoli

Continue

Figure 34: Searching for a user

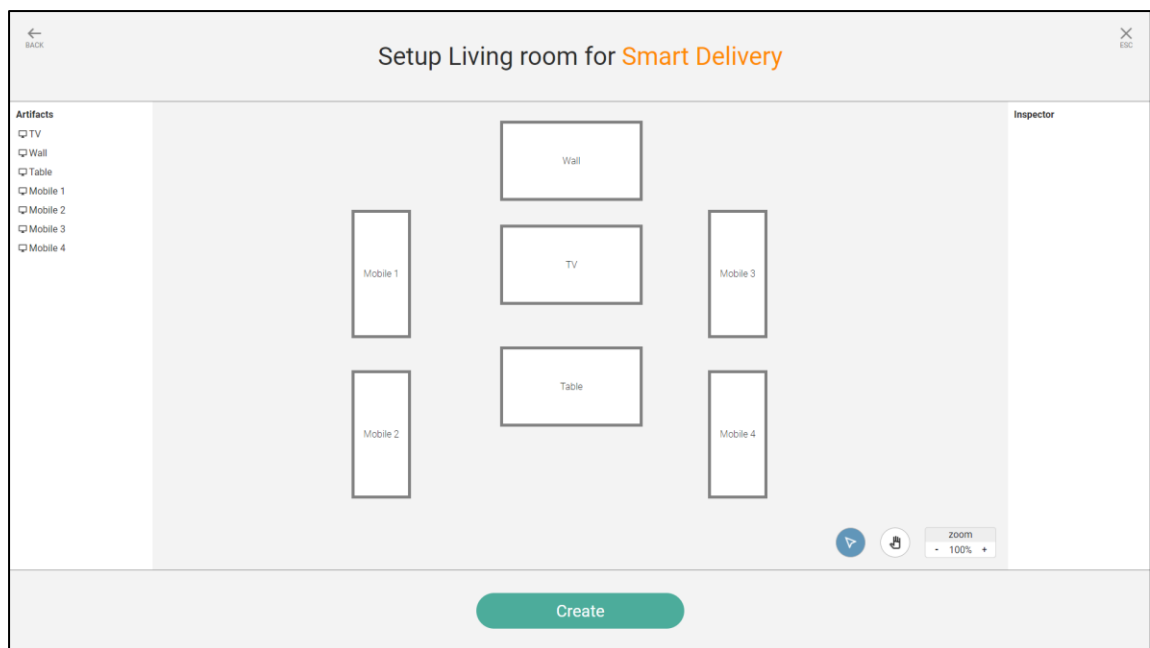


Figure 35: Space configuration

### 5.1.3 Mockup Workspace

The mockup workspace contains the list of the mockups uploaded for the project. Users can upload their mockups either by pressing the upload button and browse through their file system to select the images or by drag and drop

(Figure 36). The mockups are presented in a list showing a thumbnail version of the mockup alongside the name (Figure 37). Under each mockup a list with the tags assigned for the mockup are shown. Users can view the mockups as a unified list or by tag, grouping mockups based on their tags. The mockups can be sorted by name or date of upload. When the mouse hovers over a mockup, information about the mockup appear on top of the thumbnail (i.e. the original name of the image and the date that the mockup was uploaded or updated). On the top right of the mockup, a menu button is available with options to rename, delete, or add tags. Users can select a mockup by pressing the round button on the top left of the mockup. When a mockup is selected, a menu bar appears on the bottom. The menu bar offers the same functionality as the menu that opens from the top right button of the thumbnail. Users can also select multiple devices and use the bottom menu bar to add tags on multiple devices simultaneously or delete them. When selecting to add a tag, a window slides from bottom that allows users to select tags (Figure 38). User can select device tags or add their own from a button on the top right of the window. Search and filtering by device or user tags is also available while selecting tags.

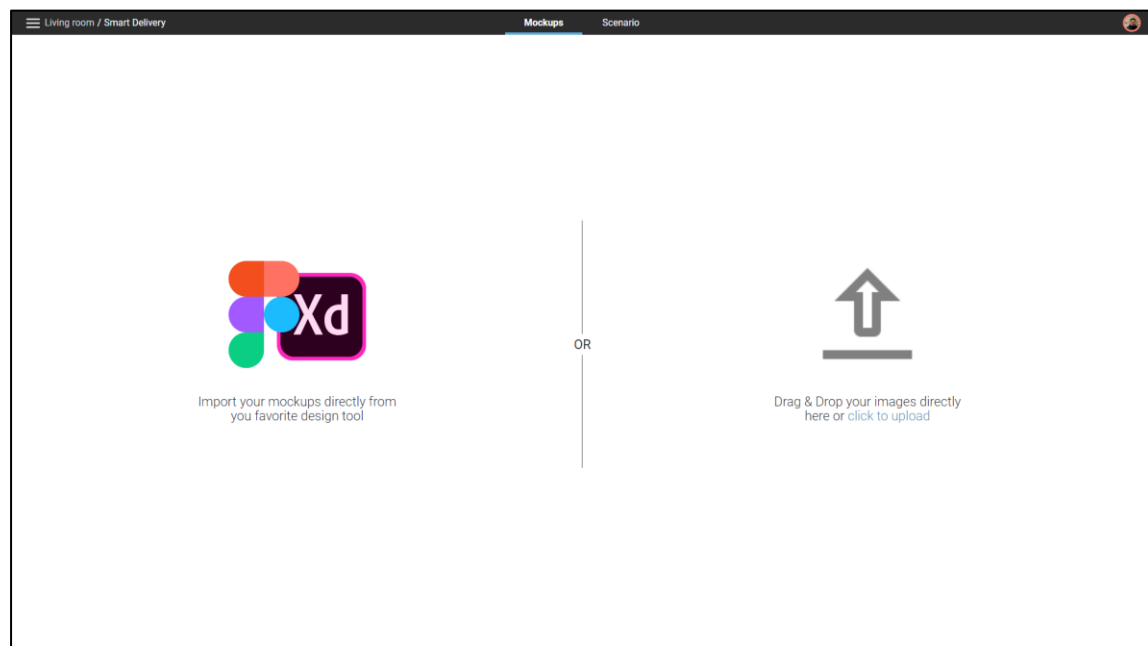


Figure 36: Mockup workspace with no mockups

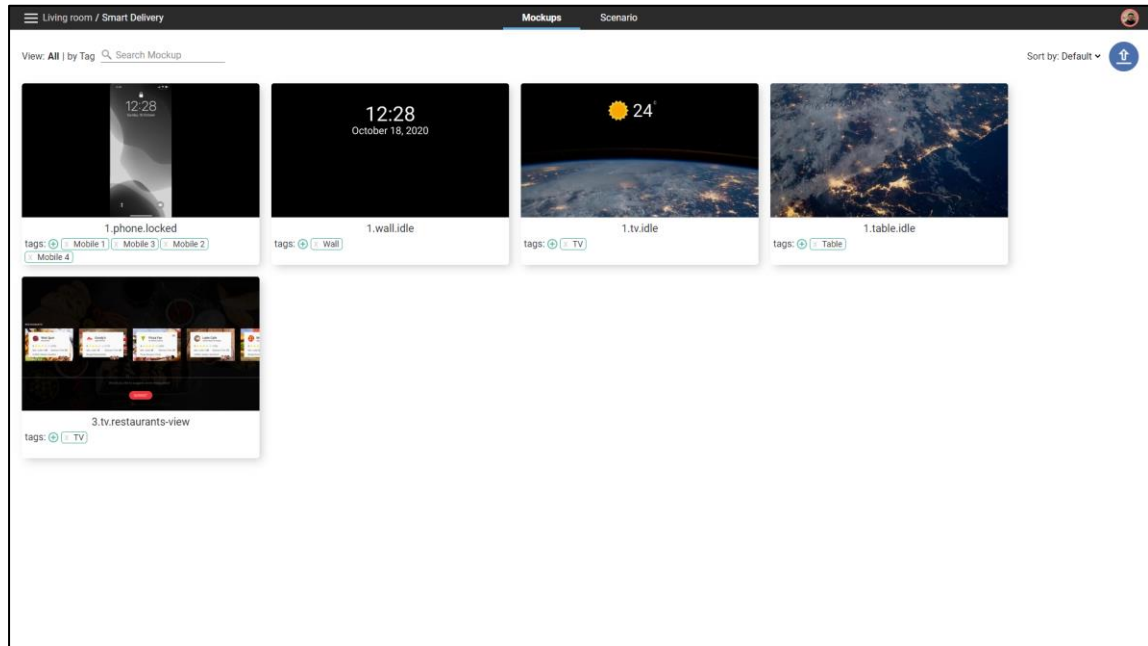


Figure 37: Mockup workspace with mockups uploaded

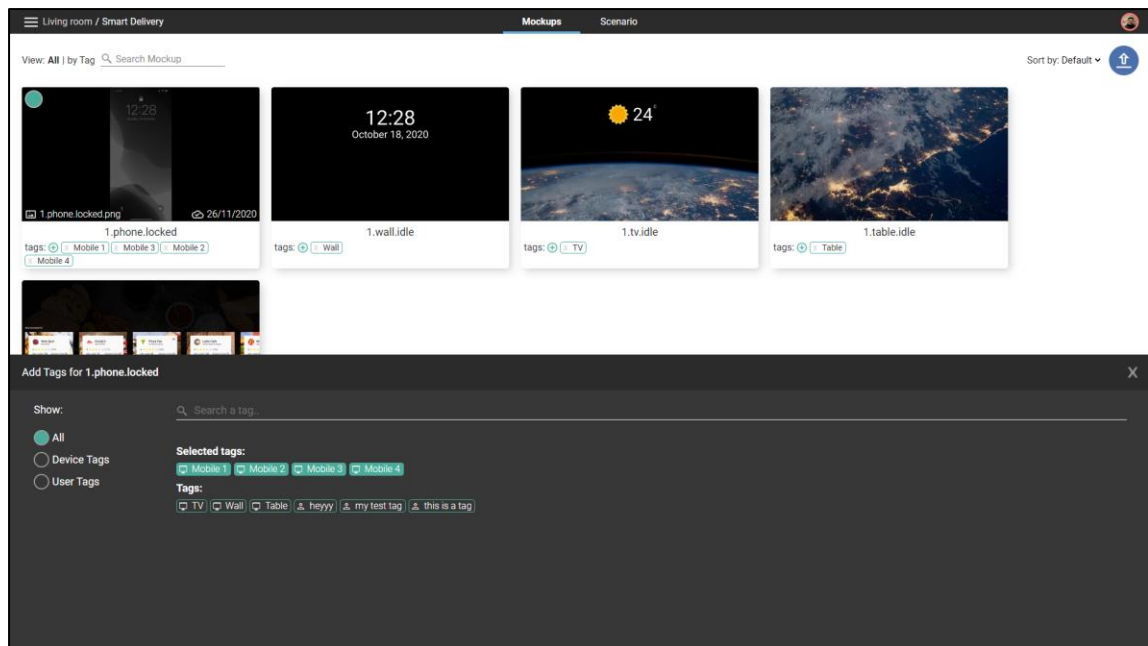


Figure 38: Adding tags to a mockup

### 5.1.4 Mockup preview

When a user presses on a mockup in the mockup workspace, the mockup preview windows opens. The mockup window offers two tabs, the preview tab and the interactivity tab. The preview tab (Figure 39) shows in the left side the

mockup in a large view and on the right side is a panel with details of the mockup. From the right panel, users can rename the mockup, add description and tags. On the top right side, they can set the mockup as the cover of the project or delete it. The interactivity tab (Figure 40) allows users to add interactive areas on the mockup. Users can drag their mouse and create rectangles to indicate interactive areas. When a user creates an interactive area, a window appears that allows users to name the interactive area. User can move, resize and delete the interactive areas.

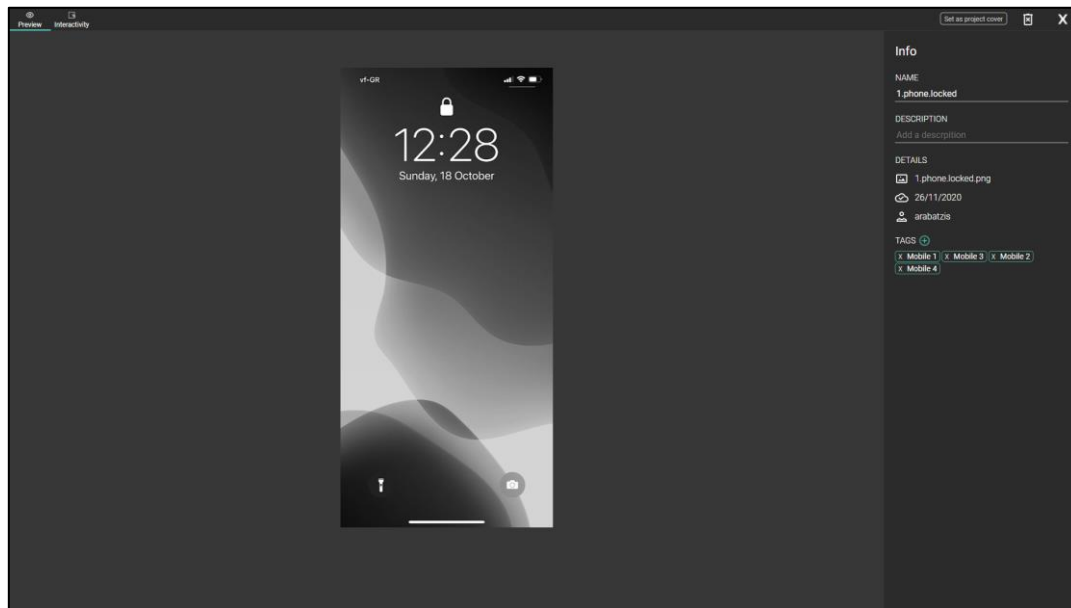


Figure 39: Mockup preview

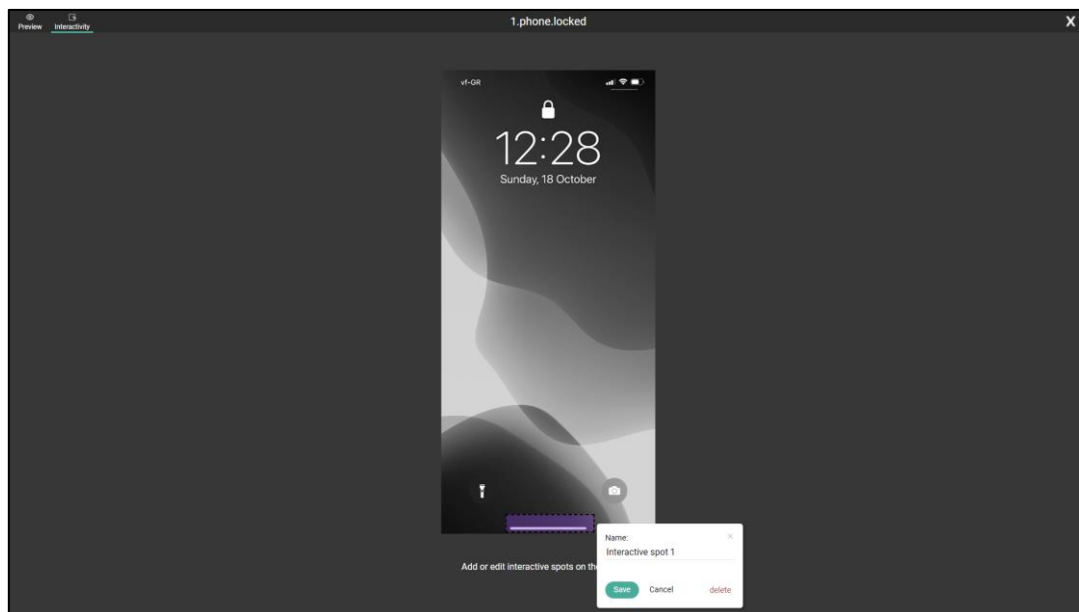


Figure 40: Adding an interaction area

### ***5.1.5 Scenario workspace***

The scenario workspace is a free flow whiteboard where users can add steps and connect them (Figure 41). A set of tools are available on the bottom right. More specifically, a select tool that can be used to select any of the steps or connections, a pan mode tool which can be used to pan around the whiteboard, an add step tool which when selected, users can click on any area of the map to add a step, an add connection tool that can be used to add connections between steps by selecting a step and then the step that is to be connected to, and a delete tool that can be used to delete steps or connections. These tools are also available through keyboard shortcuts and namely, the N key enables the new step tool, the C key enables the new connection tool, the D key enables the delete tool, and finally the S key enables the select tool. A zoom functionality is also available through the bottom right side of the screen or by pressing down the CTRL key and using the scroll wheel to zoom in and out. Moreover, pan mode can be activated by pressing down the SPACE key or the mouse middle click. Moving around the whiteboard can be achieved either with the pan mode or using the scroll wheel to move up and down or by pressing down the SHIFT key and using the scroll wheel to move left and right. A user can move around a step by pressing down and holding the left click while hovering on the step.

When selecting a step (left click on a step), options regarding the step appear around it (Figure 42). In detail, the options are: delete the step, rename the step, set the step as a starting or ending step, remove a connection that starts from the selected step, add a connection to another step, starting from the selected step or add a new step that will be connected with the selected step. Shortcuts are also available for some of these options such as the add new step which can be triggered by pressing down the TAB key.



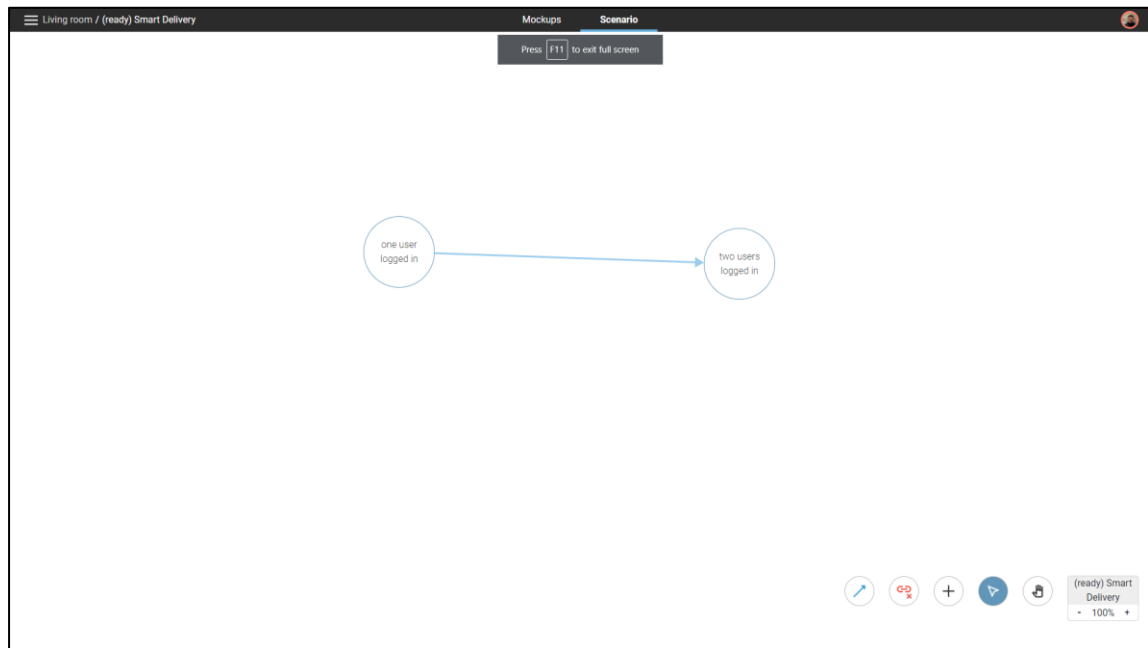


Figure 41: Scenario workspace

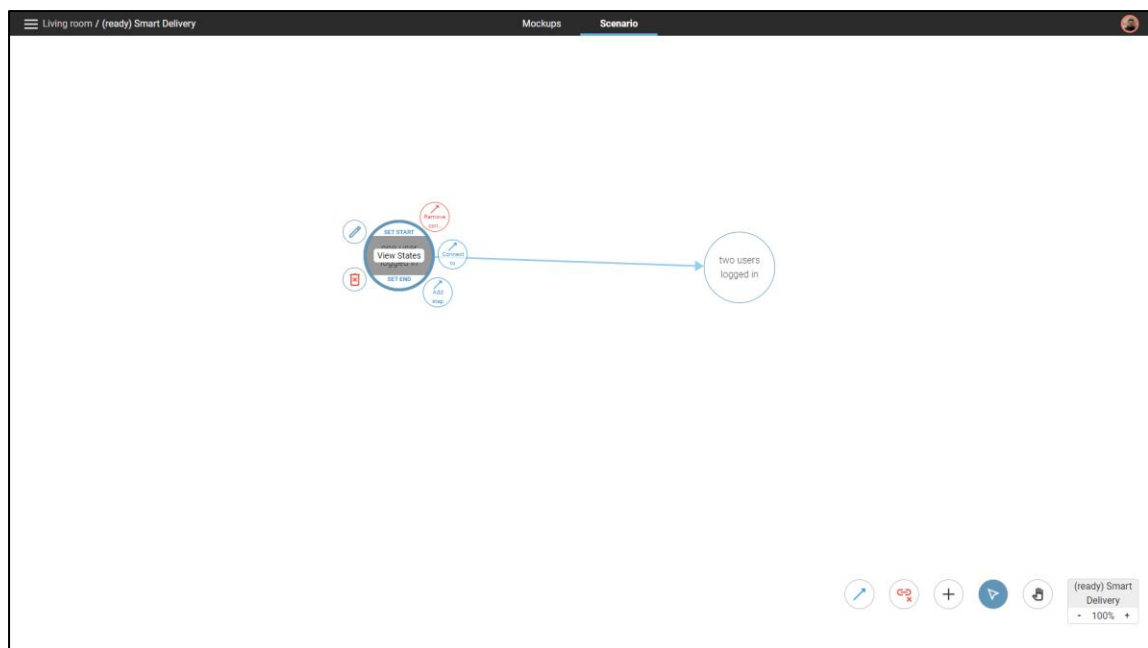


Figure 42: Step selected in scenario workspace

### 5.1.6 Interactivity page

When a user selects a connection in the scenario workspace, the interactivity page opens (Figure 43). The interactivity page allows users to create events and set interaction points that trigger a transition from one state to another. More

specifically a transition can be made from one state of a step to a state of another step. The interactivity page shows, in the left, a list of the states that a step where the connection starts contains and on the right another list of the states that a connecting step contains. Users can select an interactive area from a device of a state and then click on the state that they would like this action on the interactive area to transit. Users can also select an entire state and then select a state to transit in order to create an event. The states on the right can be minimized to display only their names instead of a mini view of the state. Users can click on a line that represents an event or interaction to delete it. Users can expand one step and set mockups or create interactive areas directly from this page without the need to go to the mockup workspace.

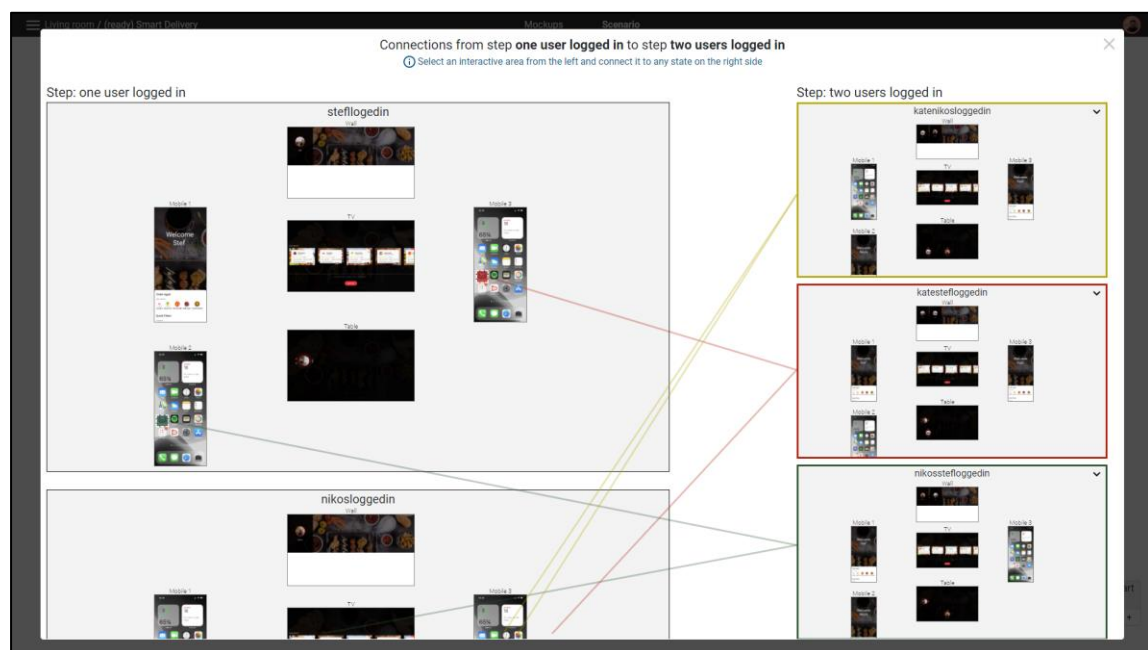


Figure 43: Interactivity page

### 5.1.7 Step page

A user can open a specific step from the scenario workspace to view, create, and edit states for the selected step. The step page is separated into three panels (Figure 44). On the left side is a list of the steps, on the center is the 2D map representing a space and on the right side is the inspector panel which based on the selected item (i.e. a state or a device) shows information and

options. The 2D map in the center is similar to the 2D map used in the space configuration page. Users can create a new empty state or create a state based on an already created one. More specifically, they can press the plus button on the left panel to create an empty state or copy through the shortcut CTRL+C and press the shortcut CTRL+V to paste it as new one containing the same mockups as the copied one. Another option is to click the duplicate button next to the name of the state to create a copy of the selected state. The functionality of copying or duplicating is particularly helpful as the difference from state to state can be a small number of mockups (of specific devices) and users are able to change only mockups on devices that are different. Otherwise, they would have to create an empty state and add mockups for all the devices. When a state is selected the map in the center shows the status of each device (i.e. the mockup that each display contains). Users can double click on a device to open the mockup selection window (Figure 45), a window that contains the mockups associated with the given device where the user can select which one to map to the device for the given state. When a device is selected from the map, information about the device appears on the right side as well as options regarding the presentation of the mockup in the screen. Users can change the orientation of the mockup, the background color in case that a mockup size is smaller than the resolution of the device and finally they can change the mockup mode (i.e. stretch to fill the whole screen, original size of the mockup, and fill the max width or max height of the screen). The 2D map is interactive, users can select devices, pan around and zoom in and out.

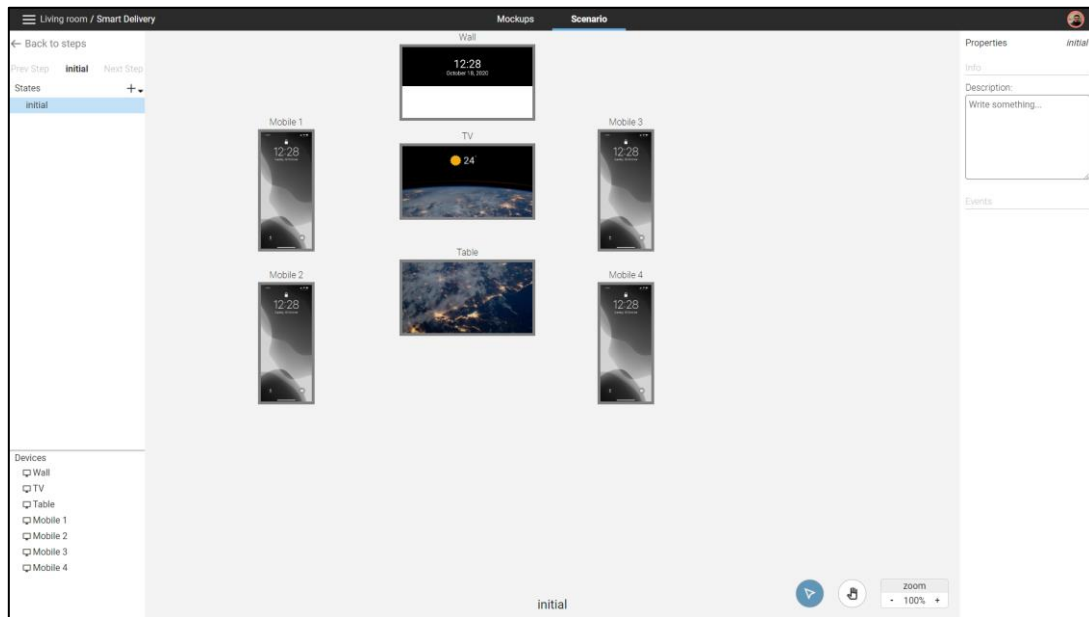


Figure 44: Step page

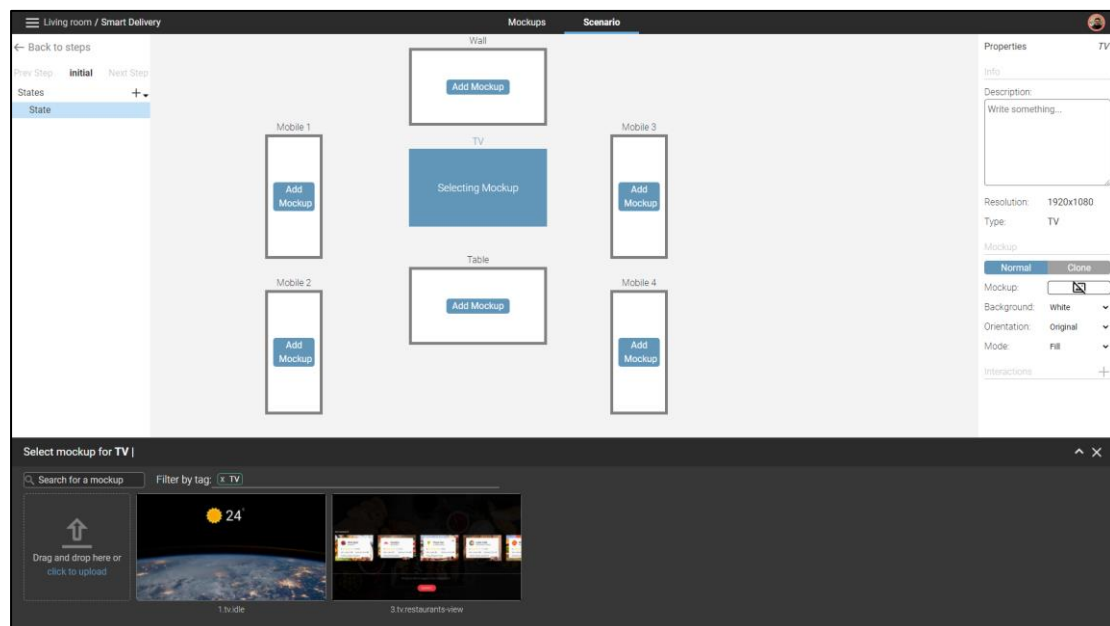


Figure 45: Adding a mockup on a display

### 5.1.8 Preview page

The preview page (Figure 46) allows designers to experience their scenario from their computer in a 2D representation of the space where they can interact with devices and verify the flow that they created. More specifically, they space is depicted in 2D in the same way that that the space is presented in the state creation process. On the bottom, there is a bar indicating the current

state as well as the step. This page is available through a generated link that can also be shared to other users to view the flow of the scenario.

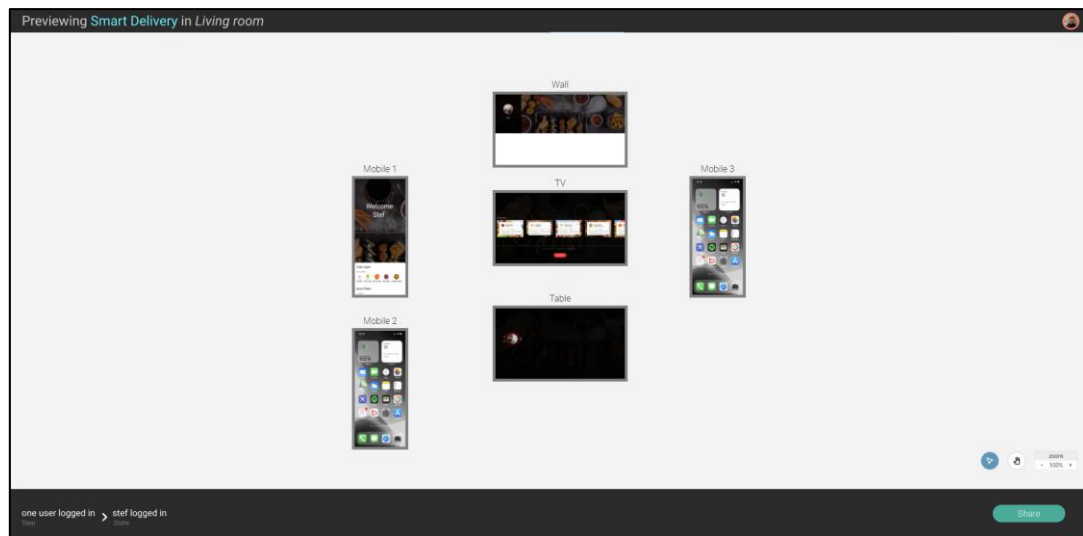


Figure 46: Preview page

### **5.1.9 Mobile companion application**

Users can use the mobile, browser-based, companion app to run their prototypes in the physical Intelligent Environments that they are designed for. After logging in a user is welcomed with a list of all the projects that they can run (Figure 47). After selecting a project, the wizard page opens. The wizard page contains specific controls regarding the Intelligent Environment and the scenario of the project. User can control the environment and set a scenario to start or end. For the scenario, in the tablet version of the companion app, they have a view of the flow of the prototype and are able to select a specific step to run on the environment. Moreover, while the prototype is running, based on the step that the scenario is currently in, a list of the available events is present where users can select to trigger in order to manually make a transition to another state.

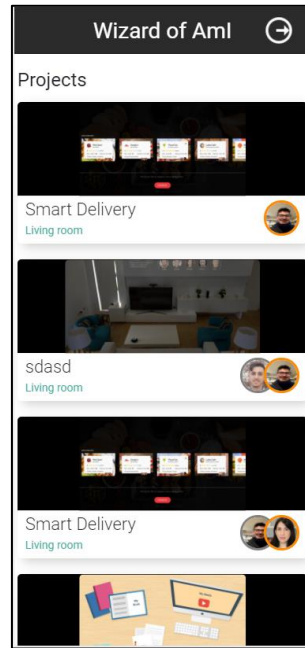


Figure 47: Selecting project in companion application

## 5.2 Wizard of AmI Architecture

Wizard of AmI uses multiple services that communicate with each other and are exposed to the front-end application through a proxy. Moreover, they store and retrieve data from the same database. The services that Wizard of AmI uses are described below.

**User service** is responsible for storing user data such as their username and basic personal info. Moreover, it handles connection status of each user (i.e. online or offline).

**Collaboration service** manages data that need to be transmitted between users while they are working in the same project. More specifically is responsible to record the actions of each user and inform other users who are working together.

**Project service** is responsible for keeping track of all the projects in Wizard of AmI. It stores data about each project, such as the users, the list of mockups that they use and metadata.

**Space service** stores all the data about the Intelligent Environments that Wizard of AmI needs in order to a) visualize the space in the User Interface and b) run the prototypes on the physical environments.

**Mockup service** is used to upload and download mockups, that are automatically optimized for use in the web which offers better overall performance. The mockup service creates thumbnail versions for each uploaded image in order to be used in pages that don't need the full version of the mockups. Moreover, mockup service also stores the interaction areas that users set on mockups.

**Scenario service** is responsible for storing all the data for the scenarios that users create. More specifically, it relies on the Database to manage the connections between the different entities that a scenario uses and is responsible for verifying that a scenario is complete (i.e. there are no steps that are not connected with other steps).

**Wizard service** is responsible for the functionality that Wizard of AmI provides regarding the Wizard of Oz paradigm. It provides the companion app with the actions that a user can simulate and allows users to configure an execution of their scenario in the physical environment.

**IE Handler** is the service that handles the infrastructure that is needed to run prototypes in Intelligent Environments. It uses the data from the space service and communicates with the different devices in the environment to set their status.

**File storage** service is used to store all the uploaded mockup from users and version control them. More specifically in case that a user uploads a new version of an image, it adds the new image and keeps a record of the previous image.

**Database** is used to store all the data that Wizard of AmI needs for users, projects, mockups, scenarios, and spaces.

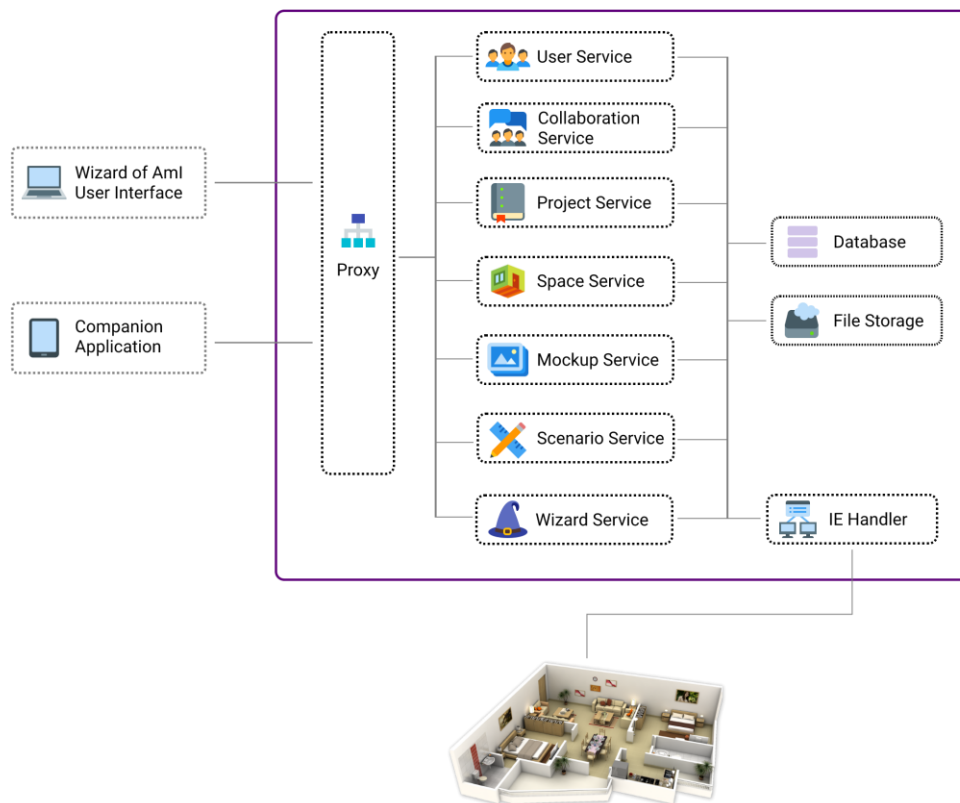


Figure 48: The Wizard of AmI Architecture

## 5.3 Wizard of AmI Database

Wizard of AmI utilizes a database to store all the data that Wizard of AmI uses to provide its functionality. A scenario uses a plethora of different entities that are linked together to be able to successfully model an interactive prototype-based scenario. An extensive modeling of the entities was required to achieve that. The different entities are described below.

User		
The entity that represents a user		
Name	Description	Type
Id	The unique id of the user	String
Username	The user's unique username	String



Fullname	The user's full name	String
password	The user's encrypted password	String
Email	The user's email	String
AvatarPath	User's avatar path	String

## Display

The entity that represents a display in an Intelligent Environment

Name	Description	Type
Id	The unique id of the display	String
Name	The display's name	String
Resolution	The display's resolution	{x: Number, y: Number}
Type	The display type (e.g. mobile, TV, tablet)	String
PhysicalDimentions	The display's physical dimensions in cm	{height: Number, width: Number}

## Device

The entity that represents a device (smart device or service) in an Intelligent Environment

Name	Description	Type
Id	The unique id of the device	String
Name	The device's name	String
Endpoint	The uri for the display	String
Type	The device type (e.g. smart light, smart blinds, other)	String

## Space

The entity that represents an Intelligent Environment that can be used in Wizard of AmI

Name	Description	Type
Id	The unique id of the space	String
Name	The space's name	String
Image	A path for an image of the space	String
Devices	A list with the devices that the space contains	Array<Device Reference>
Displays	A list with the displays that the space contains and their position	Array<{display: Display Reference, posX: Number, posY: Number}>

## SpaceConfig

The entity that represents a configuration for a space regarding a specific project

Name	Description	Type
Id	The unique id of the space configuration	String
Space	The configured space	Space reference
Devices	A list of selected devices	Array<Device reference>
Displays	A list of selected displays	Array<Display>

## Project

The entity that represents a Wizard of AmI project

Name	Description	Type
Id	The unique id of the project	String

Name	The name of the project	String
Space	The space that the project uses	Space Reference
Collaborators	A list with the users in the project	Array<User reference>
Owner	The user who created the project	User reference
Cover	A path to a cover image for the project	String
Mockups	The mockups that the project contains	MockupList reference
SpaceConfig	The configuration of the space for this project	SpaceConfig reference
Scenario	The scenario of the project	Scenario reference

## MockupList

The entity that represents the mockups that a project contains

Name	Description	Type
Id	The unique id of the mockup list	String
UserTags	A list with user defined tags	Array<String>
Mockups	The list of the mockups	Mockup reference

## Mockup

The entity that represents a mockup

Name	Description	Type
Id	The unique id of the mockup	String
Name	The name of the mockup	String
Src	The uri of the image	String
Filename	The original name of the image	String

User	The user who uploaded the mockup	User reference
Description	A description of the mockup	String
Date	The date of when the mockup was uploaded	String
DeviceTags	Tags assigned for the mockup regarding devices	Array<String>
UserTags	Tags generated from users that are assigned to the mockup	Array<String>
PreviousVersion	Reference to a previous version, in case that this mockup updated a previous mockup.	Mockup reference

## InteractionArea

The entity that represents an interaction area of a mockup

Name	Description	Type
Id	The unique id of the interaction area	String
Name	The name of the interaction area	String
Mockup	The mockup of the interaction area	Mockup reference
StartPos	The top left corner position of the interaction area (Percentage)	{x: Number, y: Number}
Width	Interaction area width (percentage) relative to the mockup	Number
Height	Interaction area height (percentage) relative to the mockup	Number

## Scenario

The entity that represents a scenario

Name	Description	Type
Id	The unique id of the scenario	String
Description	The description of the scenario	String
Settings	Setting to use or not the steps	Boolean
Steps	A list of the scenario's steps	Array<Step reference>
Tasks	The scenario tasks	Array<Task reference>

## Step

The entity that represents a step

Name	Description	Type
Id	The unique id of the step	String
Description	The description of the step	String
Name	The step name	String
States	A list of the step's states	Array<State reference>
Position	The step position in the whiteboard	{x:Number, y:Number}

## Task

The entity that represents a task

Name	Description	Type
Id	The unique id of the task	String
Description	The description of the task	String
Name	The task name	String

Step	A list of the steps that a task contains	Array<State reference>
Order	A number representing the ordering of task	Number

## State

The entity that represents a state

Name	Description	Type
Id	The unique id of the state	String
Description	The description of the state	String
Name	The state name	String
Groups	A list of the devices and their status	Array<Group reference>
Order	A number representing the ordering of the state in the step	Number

## Group

The entity that represents a state of a device in a state

Name	Description	Type
Id	The unique id of the group	String
Device	The device of the group and its position	{ device: Device reference, posX: Number, posY: Number}
Status	The device status	String
Mockup	In case that the device is a display, the info about the mockup set for the state	{ mockup: Mockup reference, background:

		string, orientation: string, mode: string }
Clone	Info about the other device In case that the device clones info from another device	Group reference

### StartStep

An entity representing an ending step

Name	Description	Type
Id	The unique id of the start step	String
Step	The start step	Step reference

### EndStep

An entity representing an ending step

Name	Description	Type
Id	The unique id of the end step	String
Step	The ending step	Step reference

### StepConnection

An entity representing a connection between two steps

Name	Description	Type
Id	The unique id of the connection	String
FromStep	The step that the connection starts	Step reference
ToStep	The step that the connection ends	Step reference

<b>StateEvent</b>		
An entity representing a state event		
<b>Name</b>	<b>Description</b>	<b>Type</b>
Id	The unique id of the state event	String
Step	The event's step	Step reference
FromState	The state of the step that contains this event (it can also be empty to indicate that it is a step wide event)	State reference
ToState	The state of the step that the trigger of the event will result	State reference

<b>GroupInteraction</b>		
An entity representing a connection of an Interaction area to a StateEvent		
<b>Name</b>	<b>Description</b>	<b>Type</b>
Id	The unique id of the group interaction	String
State	The state where the Group Interaction is	State reference
Group	The Group of the state where the interaction is	Group reference
Interaction Area	The interaction area that triggers this interaction	InteractionArea reference

## 5.4 The infrastructure behind the execution of a prototype

Wizard of AmI can run prototypes in physical Intelligent Environments. More specifically it can use, displays, ranging from TVs, PC Monitors to mobile phones, smart devices such as smart lamps, smart door locks, etc. and



contextual services, such as user location service and humidity monitoring service. To achieve that, Wizard of AmI provides different clients based on some requirements that a device should meet. The work that was done to achieve that for each type of devices alongside the requirements are described below.

#### ***5.4.1 Displays***

Any display that can present a website can be used in Wizard of AmI. This means that any display that can open a browser which is capable of loading webpages with JavaScript support works in Wizard of AmI. Smart TVs are all capable of this while any other displays that can be connected on a computer (a full desktop PC or even a Raspberry PI) can be potentially used in Wizard of AmI. To be able to manage such displays, a client application was developed that can run on Windows, Linux and MacOS. The client can automatically open a web browser on the Wizard of AmI page and pair the display. For smart phones, a wizard of AmI application is available on Android and IOS that can pair the smart phones to the system to work as displays for the environment.

#### ***5.4.2 Smart Devices***

Any smart device that provides a RESTFUL API can be imported as a device for a prototype. Given that a RESTFUL API is described with an OpenAPI Specification [61] Wizard of AmI can take advantage of these data and model the smart devices triggers and actions to allow users to access them while prototyping.

#### ***5.4.3 Contextual Services***

Given that smart devices expose their functionality through APIs, the process and requirements for contextual data is the same as to the smart devices. In details, a contextual service should contain a RESTFUL API that is described with the OpenAPI Specification in order for Wizard of AmI to extract the data needed to model the service in the tool.



# **Chapter 6**

## **Cognitive Walkthrough Evaluation**

A cognitive walkthrough evaluation experiment of the Wizard of AmI was conducted with the participation of five (5) User Experience (UX) experts with prior experience in designing systems for Intelligent Environment. The evaluation process was conducted at the Human-Computer Interaction Laboratory (HCI) of the Institute of Computer Science of the Foundation for Research and Technology - Hellas (ICS-FORTH). Cognitive walkthrough [62] is a technique for evaluating the overall user interface of a system and focuses on how a system is perceived by users the first time they use it without formal training. The goal of the evaluation was to identify any potential issues regarding the concept, missing features, and usability errors through comments from the experts before planning a large-scale user-based evaluation.

### **6.1 Process**

During the evaluation, a conductor was present to help the UX experts during the experiment. Users were given five tasks, one by one, and after each task, the conductor asked four questions. The four questions used, are based on the work of [63]: (i) Will the user try and achieve the right outcome, (ii) Will the user notice that the correct action is available to them, (iii) Will the user associate the correct action with the outcome they expect to achieve, and (iv)

If the correct action is performed, will the user see that progress is being made towards their intended outcome. The conductor was keeping notes for each task containing any comments or suggestions from the experts and recorded the answers for the four questions.

## 6.2 Results

The most notable issues that were identified from the evaluation process, along with the proposed solutions, are listed below:

**Issue 1:** While creating an Interactive Prototype for an Intelligent Environment, a designer may decide to add another device or remove an already selected one. Therefore, they should be able to configure the space while building then scenario rather than having to decide definitively the configuration from the creation of a project.

**Solution:** The wizard of AmI will integrate the feature of configuring the space while creating the scenario. For each project, a config page will be always available in case a designer decides to remove or add a device in the space.

**Issue 2:** The page that allows users to connect the different states is complicated and the mental model that a user creates after using the rest of the system does not apply in this page. Moreover, the interaction points on the mockups are not easily visible to the user.

**Solution:** The connection page will be redesigned to match the overall aesthetic of Wizard of AmI and the interaction with it will be similar to the page where a user creates and connects steps. This will help with the continuity while using the overall system. For the interactive areas of a mockup, a more vibrant color will be used to be easier for a user to observe over a mockup.

**Issue 3:** In parts of the process where a user creates an interactive prototype-based scenario, they can right click to view and select actions, while in other parts they cannot. This is an example of an action; this issue appears for multiple actions that in one page can be achieved with a certain way and in another page with a different way.

**Solution:** The way that users can perform actions in Wizard of AmI will be unified across the system to be more intuitive. Users expect that if an action can be performed in a certain way for one page, that they will be able to perform the action in the same way for another page that offers the same functionality.

**Issue 4:** Adding interactive areas for mockups is a fundamental action for creating an interactive prototype. The process to add interaction areas on a mockup contains two steps. First the user selects a mockup, which opens a page containing the information of the mockup. Then, a button is available to switch to another page that allows the user to add interactive areas. As the process of adding interactive areas is a repetitive process, the users would like for this process to be less time consuming.

**Solution:** The two tabs that the mockup page contains, the preview tab and the interactivity tab will be combined. The usability will not change as the User Interface is similar and those two tabs can be combined without losing any of the functionality and ease of use, while the process of adding interactive area will need less steps (i.e. no need to press the tab interactivity to open).



# Chapter 7

## Conclusion and Future Work

The main goal was to design and develop Wizard of AmI, an interactive prototyping tool that supports the design process of applications for Intelligent Environments. Wizard of AmI introduces a holistic approach on the process of creating and enacting interactive prototypes for such environments. Designers can collaborate through Wizard of AmI to build interactive prototype-based scenarios and assess the overall User Experience of their designs through on-site enactment. Wizard of AmI provides users with a novel graph-based User Interface where they can build their scenario by adding steps and connecting them. Moreover, a 2D representation of the space is available, providing spatial sense, where users can assign mockups to the displays of the environment to create states (i.e. the different snapshots of the application in the space). A companion application is available on mobile devices, which can be used to enact the scenarios in the physical Intelligent Environment. Through this application, designers can simulate aspects of their application and control the scenario flow.

While the Wizard of AmI is already in use from designers in Human-Computer Interaction Laboratory of the Institute of Computer Science, at Foundation for Research and Technology - Hellas (ICS - FORTH), potential changes and updates arise from the cognitive walkthrough evaluation as well as the actual

use of the tool by end-users in the laboratory. A detailed plan for future work is described below.

- A. One important step that is needed to verify the tool's usability is to conduct a formal user-based evaluation. The design team plans to conduct an evaluation with at least 15 participants, who have prior experience on designing applications for Intelligent Environments. The results of the evaluation will help to identify issues that were not possible to find through the cognitive walkthrough evaluation and determine how to improve the overall design.
- B. An integration of a space editor that will allow designers to create spaces from scratch directly within the tool.
- C. An extensive configuration support for spaces. In detail, the space configurator will allow users to add virtual devices (i.e. devices that are not already installed in the physical space). Additionally, users will be able to configure the space at any time after creating a project rather than performing the configuration only before the scenario creation process.
- D. A 2.5D preview of the space. Currently the space viewer in Wizard of Aml is a 2D representation of the physical environment. Based on the literature review and preliminary results from the use of the tool in the laboratory, users prefer to be able to view the environment in a more realistic way, hence the need to transform the space viewer to a 2.5D version.
- E. Support for previewing the created scenarios in Virtual Reality (VR). Alongside the 2.5D preview mode that will be implemented, a VR preview should be available as it allows users to assess the User Experience in a similar way as they would have by being in the physical environment but without the need be there.
- F. Support for Augmented Reality (AR) when experiencing the prototype. AR can be leveraged in order to add virtual displays or virtual smart devices in a physical space. In more detail, designers will be able to place virtual displays or smart devices in the space and use them in their prototypes.



This feature will be particularly helpful for new ideas requiring devices that are not available in a space in order to be able to test their idea without the need to actually buy equipment.



# Bibliography

- [1] “Ambient Intelligence.”  
[https://www.ercim.eu/publication/Ercim\\_News/enw47/intro.html](https://www.ercim.eu/publication/Ercim_News/enw47/intro.html)  
(accessed Nov. 29, 2020).
- [2] J.-Y. Mao, K. Vredenburg, P. W. Smith, and T. Carey, “The state of user-centered design practice,” *Commun. ACM*, vol. 48, no. 3, pp. 105–109, Mar. 2005, doi: 10.1145/1047671.1047677.
- [3] C. Abras, D. Maloney-Krichmar, J. Preece, and others, “User-centered design,” *Bainbridge W Encycl. Hum.-Comput. Interact. Thousand Oaks Sage Publ.*, vol. 37, no. 4, pp. 445–456, 2004.
- [4] L. Liu and P. Khooshabeh, “Paper or interactive? a study of prototyping techniques for ubiquitous computing environments,” in *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, New York, NY, USA, Apr. 2003, pp. 1030–1031, doi: 10.1145/765891.766132.
- [5] N. Dahlbäck, A. Jönsson, and L. Ahrenberg, “Wizard of Oz studies — why and how,” *Knowl.-Based Syst.*, vol. 6, no. 4, pp. 258–266, Dec. 1993, doi: 10.1016/0950-7051(93)90017-N.
- [6] “InVision Prototyping,” *InVision*.  
<https://www.invisionapp.com/cloud/prototype> (accessed Oct. 28, 2020).
- [7] “The digital design toolkit,” *Sketch*. <https://www.sketch.com/> (accessed Oct. 29, 2020).
- [8] “MagicIWB | Software Solutions,” *Samsung Display Solutions*.  
<https://displaysolutions.samsung.com/solutions/signage-solution/magiciwb> (accessed Mar. 19, 2020).
- [9] “Marvel - The design platform for digital products. Get started for free.”  
<https://marvelapp.com/> (accessed Oct. 29, 2020).

- [10] “Evrybo | Free Collaboration and Prototyping tool for Designers.”  
<https://evrybo.com/> (accessed Oct. 29, 2020).
- [11] “Prototyping and Usability Testing with Indigo Design | Design to Code  
| UI & UX Designs Infragistics.”  
<https://www.infragistics.com/products/indigo-design> (accessed Oct. 29,  
2020).
- [12] “Angular.” <https://angular.io/> (accessed Oct. 30, 2020).
- [13] “Free Prototyping Tool to Create Clickable Prototypes,” *Figma*.  
<https://www.figma.com/prototyping/> (accessed Oct. 28, 2020).
- [14] “Features overview | Adobe XD.”  
[https://www.adobe.com/gr\\_en/products/xd/features.html](https://www.adobe.com/gr_en/products/xd/features.html) (accessed Oct.  
28, 2020).
- [15] “Lunacy by icons8.” <https://icons8.com> (accessed Oct. 29, 2020).
- [16] “ProtoPie | The interactive digital prototyping tool.”  
<https://www.protopie.io/> (accessed Oct. 28, 2020).
- [17] F. Inc, “Origami Studio — Origami Studio 3,” *Origami Studio — Origami  
Studio 3.* / (accessed Oct. 29, 2020).
- [18] “UX-App: Interaction and User Interface Design Software for UX  
Designers.” <https://www.ux-app.com/> (accessed Oct. 29, 2020).
- [19] “UXPin - UI Design and Prototyping Tool.” <https://www.uxpin.com/>  
(accessed Oct. 29, 2020).
- [20] “Axure RP 9 - Prototypes, Specifications, and Diagrams in One Tool,”  
*Axure*. <https://www.axure.com/> (accessed Oct. 29, 2020).
- [21] “Balsamiq. Rapid, Effective and Fun Wireframing Software |  
Balsamiq.” <https://balsamiq.com/> (accessed Oct. 29, 2020).
- [22] H. hotgloo.com, “Web & Mobile Wireframe, Prototyping (Web, iOS,  
Android) and UX Tool,” *HotGloo*. <https://www.hotgloo.com> (accessed Oct.  
29, 2020).
- [23] “Pidoco - The Rapid Prototyping Tool.” <https://pidoco.com/en>  
(accessed Oct. 29, 2020).

- [24] Produre, “MockFlow - Wireframe Tools, Prototyping Tools, UI Mockups, UX Suite, Remote designing.” <https://mockflow.com> (accessed Oct. 29, 2020).
- [25] “Online Mockup, Wireframe & UI Prototyping Tool · Moqups.” <https://moqups.com> (accessed Oct. 29, 2020).
- [26] “Fluid UI Features,” *Fluid UI*. <https://www.fluidui.com/> (accessed Oct. 29, 2020).
- [27] “Wireflow - Free Wire / User Flow Tool,” *Wireflow.co*. <https://wireflow.co> (accessed Oct. 29, 2020).
- [28] “Presentator.” <https://presentator.io/> (accessed Oct. 29, 2020).
- [29] “Symu.co - display your design in a web browser.” <https://symu.co/> (accessed Oct. 29, 2020).
- [30] “Principle.” <https://principleformac.com/> (accessed Oct. 29, 2020).
- [31] “Flinto.” <https://www.flinto.com/> (accessed Oct. 29, 2020).
- [32] “Avocode App - Collaborate on Design Files with Anyone,” *Avocode*. <https://avocode.com/> (accessed Oct. 29, 2020).
- [33] “Mockplus - Design, Prototype & Collaborate better and faster.” <https://www.mockplus.com/> (accessed Oct. 29, 2020).
- [34] “Proto.io - Prototypes that feel real.” <https://proto.io> (accessed Oct. 29, 2020).
- [35] “Photo, image & design editing software | Buy Adobe Photoshop.” <https://www.adobe.com/products/photoshop.html> (accessed Oct. 30, 2020).
- [36] “Free prototyping tool for web & mobile apps - Justinmind.” <https://www.justinmind.com/> (accessed Oct. 29, 2020).
- [37] “MockingBot - The simple, easy-to-use prototyping&collaboration tool.” <https://mockingbot.com/> (accessed Oct. 29, 2020).
- [38] “Framer: A Free Prototyping Tool for Teams.” <https://www.framer.com/> (accessed Oct. 29, 2020).

- [39] J. Hassel, “Flux Studio is a Prototyping Tool for Cross-Device Interactions.” <https://getflux.studio> (accessed Oct. 29, 2020).
- [40] H.-J. Kim, J.-W. Kim, and T.-J. Nam, “miniStudio: Designers’ Tool for Prototyping Ubicomp Space with Interactive Miniature,” in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, May 2016, pp. 213–224, doi: 10.1145/2858036.2858180.
- [41] H.-J. Kim, C. M. Kim, and T.-J. Nam, “SketchStudio: Experience Prototyping with 2.5-Dimensional Animated Design Scenarios,” in *Proceedings of the 2018 Designing Interactive Systems Conference*, New York, NY, USA, Jun. 2018, pp. 831–843, doi: 10.1145/3196709.3196736.
- [42] D. D. Mauro and F. Cutugno, “A Framework for Interaction Design in Intelligent Environments,” in *2016 12th International Conference on Intelligent Environments (IE)*, Sep. 2016, pp. 246–249, doi: 10.1109/IE.2016.56.
- [43] J. C. Augusto, V. Callaghan, D. Cook, A. Kameas, and I. Satoh, “Intelligent Environments: a manifesto,” *Hum.-Centric Comput. Inf. Sci.*, vol. 3, no. 1, p. 12, Jun. 2013, doi: 10.1186/2192-1962-3-12.
- [44] K. O’hara, R. Harper, H. Mentis, A. Sellen, and A. Taylor, “On the naturalness of touchless: Putting the ‘interaction’ back into NUI,” *ACM Trans. Comput.-Hum. Interact.*, vol. 20, no. 1, p. 5:1-5:25, Apr. 2013, doi: 10.1145/2442106.2442111.
- [45] Z. Huang and J. Kong, “A Toolkit for Prototyping Tabletop-Centric Cross-Device Interaction,” *Int. J. Human-Computer Interact.*, vol. 36, no. 6, pp. 536–552, Apr. 2020, doi: 10.1080/10447318.2019.1662667.
- [46] “Topiary | Proceedings of the 17th annual ACM symposium on User interface software and technology.” [https://dl.acm.org/doi/abs/10.1145/1029632.1029671?casa\\_token=S85KfE1vO3EAAAAA:jXSicJXOvHLjezLsjUG12X3F4nJitiy2x1K6cSWh5A0HyKvsVoFZJdg1oHooWHgzQpQrKLqgPJncPA](https://dl.acm.org/doi/abs/10.1145/1029632.1029671?casa_token=S85KfE1vO3EAAAAA:jXSicJXOvHLjezLsjUG12X3F4nJitiy2x1K6cSWh5A0HyKvsVoFZJdg1oHooWHgzQpQrKLqgPJncPA) (accessed Nov. 01, 2020).

- [47] A. L. Liu and Y. Li, "BrickRoad: a light-weight tool for spontaneous design of location-enhanced applications," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, Apr. 2007, pp. 295–298, doi: 10.1145/1240624.1240673.
- [48] D. Zachhuber, T. Grill, O. Polacek, and M. Tscheligi, "Contextual Wizard of Oz," in *Ambient Intelligence*, Berlin, Heidelberg, 2012, pp. 224–239, doi: 10.1007/978-3-642-34898-3\_15.
- [49] S. Dow, B. MacIntyre, J. Lee, C. Oezbek, J. D. Bolter, and M. Gandy, "Wizard of Oz support throughout an iterative design process," *IEEE Pervasive Comput.*, vol. 4, no. 4, pp. 18–26, Oct. 2005, doi: 10.1109/MPRV.2005.93.
- [50] N. Alshammari, T. Alshammari, M. Sedky, J. Champion, and C. Bauer, "OpenSHS: Open Smart Home Simulator," *Sensors*, vol. 17, no. 5, Art. no. 5, May 2017, doi: 10.3390/s17051003.
- [51] B. Foundation, "blender.org - Home of the Blender project - Free and Open 3D Creation Software," *blender.org*. <https://www.blender.org/> (accessed Nov. 06, 2020).
- [52] I. Armac and D. Retkowitz, "Simulation of Smart Environments," in *IEEE International Conference on Pervasive Services*, Jul. 2007, pp. 257–266, doi: 10.1109/PERSER.2007.4283924.
- [53] V. Reynolds, V. Cahill, and A. Senart, "Requirements for an ubiquitous computing simulation and emulation environment," in *Proceedings of the first international conference on Integrated internet ad hoc and sensor networks*, New York, NY, USA, May 2006, pp. 1-es, doi: 10.1145/1142680.1142682.
- [54] F. Campuzano, T. Garcia-Valverde, A. Garcia-Sola, and J. A. Botia, "Flexible Simulation of Ubiquitous Computing Environments," in *Ambient Intelligence - Software and Applications*, Berlin, Heidelberg, 2011, pp. 189–196, doi: 10.1007/978-3-642-19937-0\_24.

- [55] A. A. Nazari Shirehjini and F. Klar, “3DSim: rapid prototyping ambient intelligence,” in *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies*, New York, NY, USA, Oct. 2005, pp. 303–307, doi: 10.1145/1107548.1107621.
- [56] R. F. Dam and Y. S. Teo, “5 Stages in the Design Thinking Process,” *The Interaction Design Foundation*. <https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process> (accessed May 03, 2020).
- [57] H. Plattner, C. Meinel, and U. Weinberg, *Design-thinking*. Springer, 2009.
- [58] A. Leonidis *et al.*, “CognitOS Board: A Wall-Sized Board to Support Presentations in Intelligent Environments,” *Technologies*, vol. 8, no. 4, Art. no. 4, Dec. 2020, doi: 10.3390/technologies8040066.
- [59] J. M. Carroll, “Five reasons for scenario-based design,” *Interact. Comput.*, vol. 13, no. 1, pp. 43–60, 2000.
- [60] E. B.-N. Sanders and P. J. Stappers, “Co-creation and the new landscapes of design,” *CoDesign*, vol. 4, no. 1, pp. 5–18, Mar. 2008, doi: 10.1080/15710880701875068.
- [61] “OpenAPI Specification - Version 3.0.3 | Swagger.” <https://swagger.io/specification/> (accessed Nov. 24, 2020).
- [62] J. Rieman, M. Franzke, and D. Redmiles, “Usability evaluation with the cognitive walkthrough,” in *Conference companion on Human factors in computing systems*, 1995, pp. 387–388.
- [63] M. H. Blackmon, P. G. Polson, M. Kitajima, and C. Lewis, “Cognitive walkthrough for the web,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, Apr. 2002, pp. 463–470, doi: 10.1145/503376.503459.