2020

A FRAMEWORK FOR BUILDING ASSESSMENT AND LEARNING TOOLS FOR DIGITAL SKILLS

Ann Ciesla
Michigan Technological University, aeciesla@mtu.edu

A FRAMEWORK FOR BUILDING ASSESSMENT AND LEARNING TOOLS FOR DIGITAL SKILLS

By

Ann E. Ciesla

A REPORT

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Computer Science

MICHIGAN TECHNOLOGICAL UNIVERSITY

2020

This report has been approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE in Computer Science.

Department of Computer Science

Report Advisor:	*Dr. Charles Wallace*
Committee Member:	*Dr. Kelly Steelman*
Committee Member:	*Dr. Keith Vertanen*
Department Chair:	*Dr. Linda Ott*

Table of Contents

List of figures

List of tables

Acknowledgements

I would like to thank my advisors Dr. Wallace of the Department of Computer Science and Dr. Steelman of the Department of Cognitive and Learning Sciences at Michigan Tech. Their support and creative thinking taught me how to put users at the center of design and think critically about the challenges a design may impose.

I would also like to thank the members of the Building Adult Skills in Computing (BASIC) group at Michigan Tech and the Portage Lake District Library for welcoming me into the BASIC program and providing a beautiful place to teach community members how to use technology on Saturday mornings. It was always a joy to share stories of helping people and I am thankful for the support they gave when I could not figure out a problem myself.

Finally, I would like to thank those who attend a BASIC program meeting. I will always cherish the relationships that I built with members of the community, and I am thankful to have heard about family trees, seen pictures of grandchildren, and shared a laugh about the funny things technology can do. I am also thankful for the perspective working with older community members provided me on designing technology, because I have learned that design does not work unless it works for all.

Abstract

Technology continues to intertwine with aspects of our everyday lives, requiring a new set of skills in exploring and experimenting with user interfaces. For those who grow up and keep up with technology, these skills are second nature. But for those who do not, such as older adults, these skills can be hard to master. User interfaces are changing regularly, and it is no longer suffices to teach older adults how to solve a specific problem on a specific interface. The knowledge they gain is insubstantial because it does not help when the technology changes. They no longer understand how to solve the problem when it does not appear in the exact context. This technique also does not teach them how to communicate their technical problems effectively. Older adults may also lack the confidence needed to explore and experiment, since they fear they could do permanent damage to their devices. Therefore, it is critical that older adults develop digital competency and confidence, as these skills are a necessity for navigating the everchanging world of technology today.

The Digital Skills Framework project seeks to understand the skills older adults may already have in exploring and experimenting with an interface. This provides a baseline that can help others teach them the skills that they lack. The project also seeks to teach older adults how to perform common user interface interactions, since this fundamental knowledge can help them approach technical problems they face in the future. Finally, the project provides older adults with a simple language they can use to communicate their technical problems. By building these skills, older adults can increase their confidence in using technology and solving technical problems.

This report details the design and development of the Digital Skills Framework. It begins by introducing the project and its goals. Then the process for choosing and evaluating the set of common user interface interactions is discussed. This is followed by explaining the design of the application and concludes with a discussion of future work.

1 Background

The population of older adults continues to increase at a time when technology is becoming increasingly prevalent in everyday life. However, older adults often struggle to learn new technology. They may not be equipped with the skills needed to learn how to use an interface, navigate around it, get what they need from it, or change information on it (Steelman & Wallace, 2017). This lack of knowledge may make them unconfident and anxious. Older adults can become so afraid of permanently damaging the computer that they do not want to interact with it at all (Kumar, Ureel, King, & Wallace, 2013).

Often these older adults reach out to friends or family for help, and they may provide step-by-step solutions to solve the specific problem. While their intentions are good, this does not teach the older adults the higher-order skills needed to repeat the task again or complete the task when the user interface changes. Therefore, it is critical to teach older adults how to explore an interface to understand what affordances exist, how to experiment with an interface to understand what can be manipulated, and how their interactions change the state of the interface. It is also critical to instill the confidence they need to feel comfortable exploring and experimenting.

It is also necessary to understand what knowledge an older adult may already have. Assessing current knowledge can help tutors understand what skills an older adult has and what skills need to be worked on, so the tutor can tailor the learning experience the older adult's specific needs. But there are a variety of other settings beyond formal tutoring where this knowledge would be useful. A friend or family member who is trying to help an older adult would benefit from understanding their current knowledge, since the friend or family might assume certain interactions are easy or obvious when they are not easy or obvious to an older adult. This knowledge can benefit programs that seek to teach older adults computer skills because understanding what knowledge attendees already have help program managers decide on what skills need to be taught in future programs.

The Digital Skills Framework aims to instill confidence in older adults by teaching them the skills they need to interact with user interfaces. It teaches them how to explore a user interface by incentivizing this exploration. It also teaches them how to experiment by giving them a safe space to make mistakes and try again. Finally, it can be used to assess an older adult's current knowledge by tracking their interactions and determining their digital competency score.

2 Related Work

There are a variety of existing digital competency assessments and frameworks that can be used to describe a user's digital skills and knowledge. A comprehensive review from Ferrari (Ferrari, 2012) considers fifteen frameworks that are used to quantify and develop digital competency through curricula, courses, certification programs, or academic papers. These frameworks provide a structured approach for understanding and developing digital competency in users. However, some are geared towards children, who likely grow up with some exposure to technology. In a survey from the Erikson Institute of 1,000 parents, 84% said they use technology with their children at home (Erikson Institute, 2016). The frameworks geared towards include BECTA, CML, DCA, e-safety kit, NCCA ICT framework, and The Scottish Information Literacy Project. Because of their exposure to technology, children already have some of the problem-solving skills required to complete a computer task and understand some of the vocabulary used to describe a computer task. In the survey 69% of parents agreed that using technology increased their child's computer and technology skills (Erikson Institute, 2016). Therefore, these frameworks do not capture how to quantify the digital competency of a true digital newcomer.

Others are certification-based assessments, where users must take an exam at a scheduled time in a designated location. These include ACTIC, ECDL, IC^3 Internet and Computer Core Certification, iSkills, and Pedagogic ICT license. These types of assessments may increase the anxiety a user feels around using a computer because of the perceived pressure to perform well. They do not provide a safe environment for users to make mistakes and experiment with the interface during assessment. Therefore, they may not reflect a user's true digital competency.

Another existing assessment is the Digital Wheel of Competence (Digital Dannelse, 2009). This is an online self-assessment that asks users to rate themselves on sixteen digital competencies under the categories of security, production, information, and communication. The goal of the assessment is to determine which digital competencies are user already has and which need to be developed. This assessment is also not friendly to true digital newcomers since some of the language used in the questions necessitates understanding vocabulary used to describe computer tasks. While it can be useful to know where the user's knowledge gaps exist, the language used in this assessment can cause users to feel intimidated by technology.

For example, a question about storage options asks if a user "knows the pros and cons of storing data in the cloud, hard drive or portable device". For a user who does not understand what each of the options are, this can be intimidating. A question about hardware ass if a user "understands the connection between elements such as CPU, RAM, motherboard, cables (eg hdmi) and network routers". Clearly, only users with a technical background would be comfortable with this question and this could increase the anxiety of newcomers who may not understand the basics of using computer hardware,

let alone understanding how it works. Therefore, this assessment would not be sufficient to understand the digital competency of a newcomer.

Lastly, previous work on developing the Digital Skills Framework has been completed. The assessment of digital skills was chosen to be gamified because games are effective at simulating real user interface interactions while providing a safe space for a user to explore and experiment (Lumsden, Edwards, Lawrence, Coyle, & Munafo, 2016). They also ease the anxiety that a user may feel when taking an assessment, making them effective for collecting information about a user's performance without the negative impacts of test anxiety (Lumsden, Edwards, Lawrence, Coyle, & Munafo, 2016). The previous design was a treasure hunt game. Users would be vacationers who can travel to different locations that are represented as different webpages. Users score points by collecting gems on these pages. Points would be awarded based on the depth of exploration needed to find the gem and a user's score would be used to determine their digital competency. There would also be a toolbar with features that would aid the user in finding gems.

3 Building Adult Skills in Computing

The Building Adult Skills in Computing (BASIC) program at Michigan Tech seeks to equip older adults with these skills. The program pairs tutors who are student volunteers with members of the community who are seeking help with different technologies, including personal computers, tablets, and smartphones (Steelman, Tislar, Ureel, & Wallace, 2016). Student tutors work with community members one on one to help them solve the problems they are facing. They begin by working with the community member to understand their problem, develop a plan to solve it, and then implement that plan (Steelman, Tislar, Ureel, & Wallace, 2017). Throughout this process, tutors model how to explore and experiment with an interface and encourage the community member to take the lead on developing and implementing the plan to solve the problem.

The program focuses on building digital competency and confidence in older adults. Tutors build digital competency by explaining concepts behind technology and the problem-solving skills needed to approach a technology problem. They instill confidence by letting the community member drive, meaning they complete the operations to solve the problem instead of the tutor doing it for them (Kumar, Ureel, King, & Wallace, 2013). They also show that it is okay not to know how to do something. Sometimes community members come in with questions on software that no tutor has experience in, and tutors will admit they might not know the answer to the problem right now, but they are willing to work through it together. This shows community members than even "experts" do not always know the answer and gives them a chance to model problem-solving skills (Steelman & Wallace, 2017).

The BASIC program is open to community members at any experience level. Some community members come in come in with a basic understanding of how to use their device, but want to learn how to do more with it, while others are wondering how they can use their device at all. Some community members can use technical terminology to describe their problem while others cannot describe the problem at all. The program invites community members to come with their specific needs and tutors tailor learning appropriately (Steelman, Tislar, Ureel, & Wallace, 2016). As a tutor, it can be hard to determine what skills a community member already has. Sometimes, a tutor will begin to explain a concept and realize that there are underlying knowledge gaps that need to be filled before the community member will be able to the solve their problem on their own. It can also be hard to communicate with the community member when they do not have a way to express their problem or what is confusing them about a technology. Therefore, it is necessary to understand their current digital competencies to teach them the higher-order skills they need.

4 Project Goals

The Digital Skills Framework project has many goals. One goal is to understand an older adults' current digital competencies. Their skills in exploring and experimenting with an interface need to be quantified to help tutors understand where they are at. By using the framework as an assessment, a tutor would understand what skills need to be improved and therefore focus their attention on building those skills. This prevents tutors from trying to explain concepts that may be too advanced for an older adult, since they do not have the fundamental knowledge that they need. It also helps a tutor understand how an older adult has progressed in their learning. After spending time teaching them about one skill, an older adult can retake the assessment to see if the skill has been mastered and what skills need to be worked on next.

Another goal of the Digital Skills Framework is to give older adults a language they can use to describe their technical problems. Most interactions with a user interface can be sorted into four categories: learn, get, navigate, and change. Learn captures what a user must know to perform an interaction, such as learning which page of a website contains help information. Get captures what a user must acquire to perform an interaction, such as getting an item from an online store. Navigate captures how a user must move to perform an interaction, such as clicking on an option in a menu to move to a different webpage. Lastly, change captures what data a user must manipulate to perform an interaction, such as changing a password to login to a website. This simple language is easy for older adults to use to communicate their problems while teaching them ways to solve them.

A final goal of the Digital Skills Framework is to teach older adults how to complete common user interface interactions. Once they have completed the assessment to get a score, it can be used as learning tool. By going back through the interactions, they can build their problem-solving skills by learning what steps are needed to complete them. This can also reinforce categorizing interactions into the categories above: learn, get, navigate, and change. Framing the interactions in this way gives older adults examples of these interactions in practice which can help them apply it to their problems.

5 Included User Interface Interactions

5.1 Common User Interface Interactions

To build the Digital Skills Framework, a common set of user interface interactions needed to be identified. A literature review was completed to determine if an existing set of common user interface interactions existed and if an existing categorization of user interface interactions existed.

5.1.1 Taxonomies and Design Patterns

Some early taxonomies were found in the literature including a taxonomy of user-oriented functions from Carter (Carter, 1986). This sought to classify and organize a set of high-level user interface functions that could be used to design user-centered applications for a variety of domains. Carter categorized functions into object functions, task-execution functions, information functions, peripheral functions, special functions, bibliographic functions, spreadsheet functions, graphics functions, and text-editing functions. While his taxonomy was exhaustive, it proved to be too broad for the purposes of the Digital Skills Framework since function names were too generic. For example, the change file and change window functions do not specify how these objects should be changed. The taxonomy was also too technical for the Digital Skills Framework since some functions were outside the scope of what an average user does. These functions include find protection status, output a directory, obtain status of current processes.

Carter, Lukey, and Schweighardt refined Carter's earlier work by reframing it into the AMOA classification (Carter, Lukey, & Schweighardt, 1991). This classification allows functions to be defined by specifying an action, modifier, object, and attribute. This classification was more useful, but it was also too broad for the use in the Digital Skills Framework. The defined functions hinted at common user interface interactions used today, such password protection, but this was not enough to define a specific interaction. There did not appear to be a specific taxonomy for current user interface interactions in the literature, specifically those that have appeared since the adoption of the Internet.

More recent literature on design patterns proved to be helpful for creating a set of common user interface interactions. User interface design patterns are recurring components that designers use to solve problems in user interface design (Interaction Design Foundation, n.d.). In particular, *Designing Interfaces* by Tidwell (Tidwell, 2011) was helpful in providing high-level design patterns to be considered. The book broadly describes twelve design patterns based on human behavior instead of interface design elements, which include safe exploration, instant gratification, satisficing, and incremental construction. Later, the book goes into detail of multiple specific design patterns that are categorized into user goals. For example, under the goal of getting around, are the patterns for menu pages, breadcrumbs, and sitemap footers. This was helpful for understanding what higher-level goals users may have when approaching an

interface. It lay the foundation for the learn, get, navigate, and change categories used to sort users' interactions with the Digital Skills Framework. These high-level goals are common to most user interface interactions and capture what the user is looking to gain from the user interface. The book also helped to considering what specific interactions were common for completing these goals, which could be included in the Digital Skills Framework.

5.1.2 Included Interactions

After understanding the high-level categorization of design patterns, a final list of user interface interactions to be used in the Digital Skills Framework was compiled. Two websites were helpful, namely: *UI Patterns* from Toxboe (Toxboe, n.d.) and *Pattern Library* from Welie (Van Welie, 2008). The list was created by extracting the interactions common to the above resources and combining it with what the researchers found was common in their experience with helping community members with technical problems during the BASIC program. The chosen list of patterns and their definition can be seen below.

Table 5.1. List of User Interface Interactions Included in the Framework

Interaction	**Definition**
Multi-layer dropdown menu	Menu where each menu item has subsequent menu items below that lead to more menu items.
Icon menu	Menu where menu items are represented by images and labels.
Map navigator	Map is shown with points of interest and navigational links are provided to more information.
Product page	Page that display information about a product on an ecommerce site.
Shopping cart	Page that displays a collection of selected products that the user can add to, update, and remove from.
Forgot Password	Allows the user to reset their password. Typically, the password is verified through the email address linked to the account.
Finding an image in a set	Photos may appear in random order and may have ambiguous labels that are not descriptive of the contents of the photo.
Move files from one place to another	Users may want to move a file from one location to another via drag and drop or copy and paste.

Settings	Allows the user to manipulate various settings for their account or device.
Sharing	Allows the users to quickly share particular interactions with their social networks.
Filtering/Sorting	Allows users to find the information they are looking for easily by narrowing or rearranging data.
Favoriting	Allows a user to create a personalized list of items of interest.
Chat	Allows a user of a system to contact other users of the system.
Opening/Closing	Allows user to open or close a window on the screen.
Resizing	Allows the user to make content bigger or smaller.

5.2 Heuristics for Categorizing Interactions

Next, two heuristics were considered to help quantify the difficulty of these interactions. The first comes from the Organization for Economic Co-operation and Development (OECD) and the second comes from the European Commission.

5.2.1 Four Levels of Technology Proficiency

The OECD skills research sought to quantify the difference between the general population and the technologically elite. Researchers tested the skills of people from age 16 to 65 by asking them for perform 14 computer tasks that ranged in difficulty levels. Afterwards, they defined four levels of technology proficiency based on the types of tasks users completed successfully (Nielsen, 2016).

Below Level 1. Tasks in this level are based on well-defined problems that only require the use of one function. The task can be completed without categorical or inferential reasoning in a few steps. No information needs to be manipulated, and there is no sub-goal that must be completed first. An example task in this level is deleting an email.

Level 1. Tasks in this level require little no to use of navigation or other functions. They take few steps to solve and simple reasoning must be used. The goal of the task can be inferred from the task statement. There are some monitoring demands, which are defined as the checking whether the appropriate procedure has been used to solve the problem. An example task in this level is finding all emails from a specific person.

Level 2. Tasks in this level require some use of navigation and other functions. They take multiple steps to solve and require some evaluation inferential reasoning. The goal of the

task may be defined by the user, though the criteria for success is defined by the task statement. There are more monitoring demands, and unexpected results are unlikely to appear. An example task in this level is finding meeting minutes sent over email by a specific person last month.

Level 3. Tasks in this level require the use of navigation and other functions. They take a large number of steps to solve and require lots of evaluation and inferential reasoning. The goal of the task and the criteria for success is defined by the user. There are high monitoring demands, and unexpected results are very likely to appear. An example task in this level is calculating the percentage of emails sent by a specific person regarding a specific topic.

5.2.2 Digital Competence Framework for Citizens

The European Commission provides the Digital Competence Framework for Citizens as a tool for improving digital competence by supporting the development of learning materials and assessment of citizen's current digital competence. It defines eight levels under the categories of foundational, intermediate, advanced, and highly specialized skills (Carretero, Vuorikari, & Punie, 2017).

The first and second levels include simple tasks. Users can identify their needs and perform simple searches to find information. In the first level, they can complete tasks with guidance and in the second level, they can complete tasks autonomously with guidance where needed. Users are remembering how to complete the task, instead of understanding how to complete it. An example of a task in these levels is identifying resources and accessing them to use for writing a report with help from a teacher.

The third and fourth levels include well-defined tasks that can be completed autonomously. Users can understand how to complete the task. In level three, tasks are routine while in level four, they are not. Users can explain their needs and explain how to find the information they need. An example of a task in these levels is giving examples to a teacher or classmates of online resources that can be used to write a report.

The fifth and sixth levels include a variety of tasks and problems where users can apply their knowledge and evaluate solutions. In level five, users can guide others through completing a task and in level six, users can adapt to complexities that may arise while completing a task. They can assess their needs and explain how to access the information they need. An example of a task in these levels is explaining to classmates how to use online resources to write a report.

The seventh and eight levels include complex tasks that may have limited solutions or many interacting factors. Users are able to create problem solving strategies and contribute to the domain's knowledge base. They can also integrate knowledge and propose new ideas. An example of a task in these levels is creating a resource that can be used by others to write reports.

The Digital Competence Framework for Citizens also defines five competence areas: information and digital literacy, communication and collaboration, digital content creation, safety, and problem solving (Carretero, Vuorikari, & Punie, 2017).

The information and digital literacy competence area includes tasks that are related to finding, evaluating, and managing information. The task of identifying resources and accessing them to use for writing a report would fall in this area.

The communication and collaboration competence area includes tasks that are related to interacting, sharing, engaging, and collaborating through digital technologies and managing digital identity. An example of a task in this area would be choosing a digital communication tool to use for working with a group on a report.

The digital content creation competence area includes tasks for developing and integrating digital content, working with copyright and licenses, and programming. The task of creating a resource that could be used by others to write a report would fall in this area.

The safety competence area includes tasks for protecting devices, personal data, personal health and well-being, and the environment. An example of a task in this area would be protecting data and content on a school's digital learning platform.

The problem solving competence area includes tasks that are related to solving technical problems, identifying needs and technological solutions, using digital technologies creatively and identifying digital competence gaps. An example of a task in this area would be identifying a technical problem that can occur when using a digital learning platform and the IT support needed to solve it.

5.2.3 Heuristics Applied to Interactions

The first heuristic was used to understand which interactions in the Digital Skills Framework could be expected to be completed by users at varying digital competency levels. The number of steps and amount of reasoning required to complete each interaction was defined and this was used to categorize interactions by the Technology Proficiency Levels. This helped create an ordering for the interactions, so they appear in order of the proficiency levels in the framework. This way, users move through simpler tasks to more advanced tasks in a linear way.

The second heuristic was used to understand what knowledge and skills users of the Digital Skills Framework would need to complete the defined interactions. The complexity of steps was used to categorize interactions by the levels in the Digital Competence Framework and the tasks required to complete each interaction was used to determine the competence area of the interaction. While this heuristic did not have as much of an impact on the ordering of the interactions, it helped information what kind of hint information would be needed to help the users complete the interactions.

Finally, these interactions were sorted by their heuristic values to create a set of levels to be used in the framework, which can be seen in the table below. For the Digital Competence Area value, 1.1 represents the first task in the first competence area, namely identifying information. A value of 4.2 represents the second task in the fourth competence area, namely protecting personal data.

Table 5.2. List of Digital Skills Framework Levels Ordered by Heuristic Value

Task	**Steps**	**Reasoning Required?**	**Technology Proficiency Level**	**Digital Competence Area**
Scrolling	1	No	<1	1.1
Closing a pop up	1	No	<1	1.1
Finding an image in files of photos	2	No	<1	1.1
Favoriting something	1	No	<1	2.2
Resizing	2	No	<1	1.1
Filtering/Sorting	3	Some	1	1.1
Product page	3	Some	1	1.2
Shopping cart	6	Some	1	2.1
Sharing	3	Some	1	2.2
Map navigator	2	Some	1	1.2
Finding a setting	3	Some	2	1.3
Changing a setting	3	Some	2	1.3
Chatting	3	Some	2	2.1
Forgot password	6 or more	Yes	2	4.2
Multi-layer dropdown menu	3 or more	Yes	2	1.2
Move files from one place to another	6 or more	Yes	2	2.2

6 Digital Skills Framework

6.1 Overview

The current version of the Digital Skills Framework implements the goals of giving older adults a language to describe their technical problems and teaching older adults how to perform common user interface interactions. The design of the framework is described in the following subsections.

6.2 Gamification

As mentioned in Section 2, the assessment portion of the framework was chosen to be gamified because games provide a safe space for users to explore and experiment and they reduce the anxiety users feel when taking an assessment (Lumsden, Edwards, Lawrence, Coyle, & Munafo, 2016). The game is designed to be a spy game. Users become secret agents at a spy agency whose goal is to understand computers and the internet. They are tasked with missions that outline the interaction they must complete. A jewel is placed somewhere on the screen and is revealed after the user completes the interaction. A mission ends when the user has found and clicked on the jewel.

Users begin the game on a welcome screen that explains the mission of the spy agency and allows them to click a button to begin. From there, users move through missions seamlessly from the welcome page to the final page. Originally, a home page was added that users would return to between missions. Users were introduced to the mission on this screen by receiving a mission from the spy agency's headquarters. However, this was removed since it added unnecessary complexity to the game and required users to understand why they moved back to this page after every mission. At the end, users are presented with a final page that congratulates them on completing their missions and gives them information about their performance.

The spy game design was chosen because it inspires problem solving and better reflects the need for exploration and experimentation when approaching an interface. Spies are known for cleverness, thinking outside of the box, having good observation skills which transfers well to the skills needed to become digitally literate. The idea of missions clearly defines the task that needs to be completed while leaving room for ambiguity that is used in later missions to test users' reasoning. It also complements the goal of teaching users how to complete the interactions. The game views users as spies in training, and by completing missions they are becoming elite spies. This perspective shows users that the missions they complete are meant to help them expand their knowledge and get better at using computers.

6.3 Front-End Framework

The Digital Skills Framework has been implemented as a web application and is designed to be used on a personal computer. Since most people have some exposure to personal computers, this would likely be the device they choose to play the game on. The web application is implemented in React. React is a JavaScript framework for building user interfaces and it was selected because it allows for code to be reused as components, renders the pages efficiently, and is compatible with existing JavaScript and Node libraries (Facebook Open Source, n.d.).

The code is divided into components that are reused in different part of the application. For example, there is a component for the navigation bar found at the bottom of all pages. Each mission is its own component and may rely on other smaller, reused components. An example can be seen below, where the mission component for favoriting an item is made up of smaller reused card components.

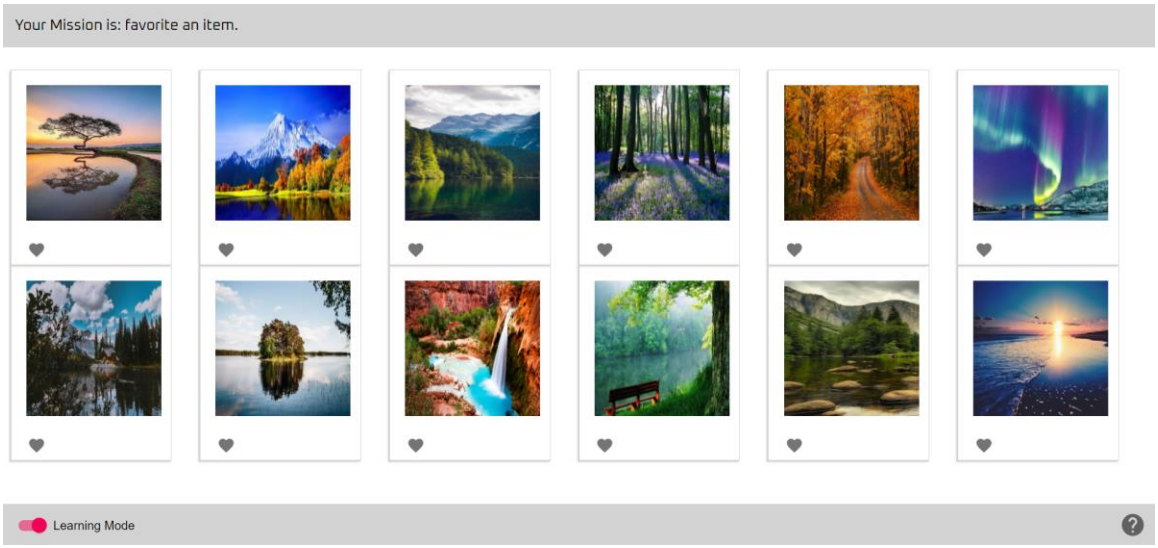

Figure 6.1 Example of smaller card components contained in a mission component.

There is a template mission component that handles the logic needed to update the application's state when the user finds the jewel and changes missions. This makes it easier to implement a new mission since only the mission's content needs to be added to the template.

Having reusable components can make the code easier to manage since it divides it into small chunks, but it complicates global state management since state must be passed between components. Typically, this is done in React by passing the state up to a parent component and then back down to the next child. However, this method became very complex as the application grew. Libraries exist to solve this problem, and Redux was chosen as the global state manager for the application. Redux is a predictable state container for JavaScript applications that centralizes application state (Abramov, n.d.).

Using Redux allows the mission number, whether the jewel has been found, whether learning mode is on, and whether the hint is clicked to be held in global state so that it can easily be accessed by any component in the application. This reduced the complexity of state management and made application state easier to keep track of.

Styling of the application was originally implemented using Material UI, which is a library of React components that are styled according to Material Design standards (Material UI, n.d.). Since Material UI components are already styled, there is minimal work needed to make an application look nice. However, since the application is themed around a spy game, it became difficult to achieve the desired look using Material UI since it can be difficult to add custom styling. Therefore, most Material UI components were removed from the application. Now components are made of simple HTML elements and are styled using custom CSS. This made developing the application more difficult, but it was easier to achieve the desired spy theme while keeping the style of the application minimalistic.

6.4 Missions

The common user interface interactions listed in Table 5.2 are implemented in the game as missions. Most interactions are put into the context of the spy game. For example, the mission for the product page interaction is for users to add spy gear to their cart. Spy gear includes night vision googles, walkie talkies, and an audio recording pen. Other interactions are not themed, such as the resizing interaction. The mission is simply to resize a box on the screen to find the jewel. Interactions like this with little complexity are not themed since it was difficult to tie the spy game theme in when there is only one interaction that needs to be made to complete the mission.

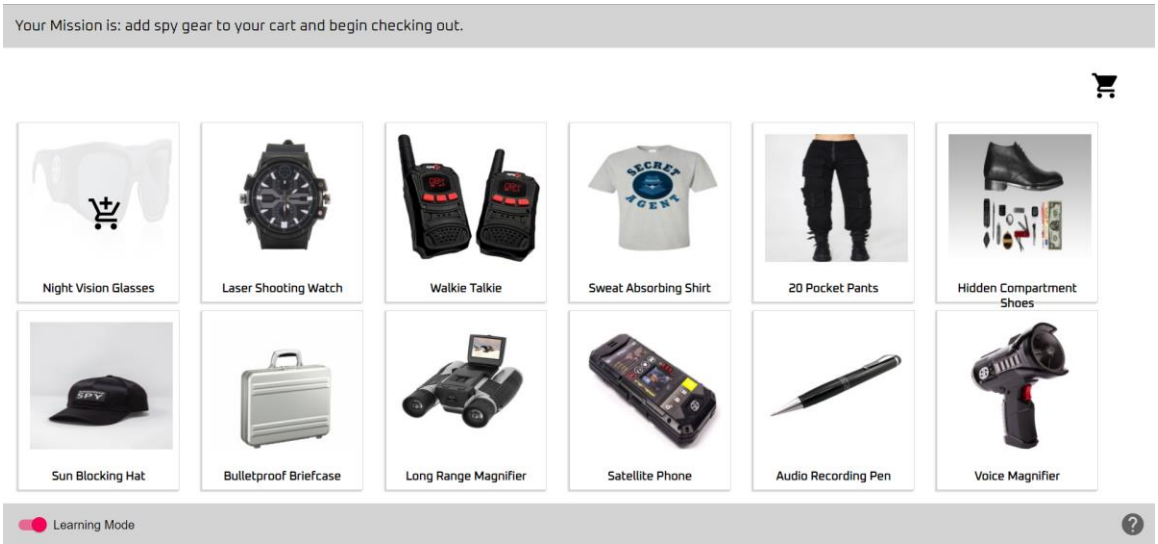

Figure 6.2 Screen capture of the product page mission.

Some interactions are designed to mimic existing interfaces. For example, the interactions for finding a setting and for changing a setting are designed to mimic the settings screen on windows computers and the map navigation interaction mimics the design of Google maps. Using similar designs helps to teach the user the underlying interactions needed to interact with the interface successfully. When the user encounters Windows settings or Google maps in the future, it will not be as intimidating because they have learned how to approach the interface.

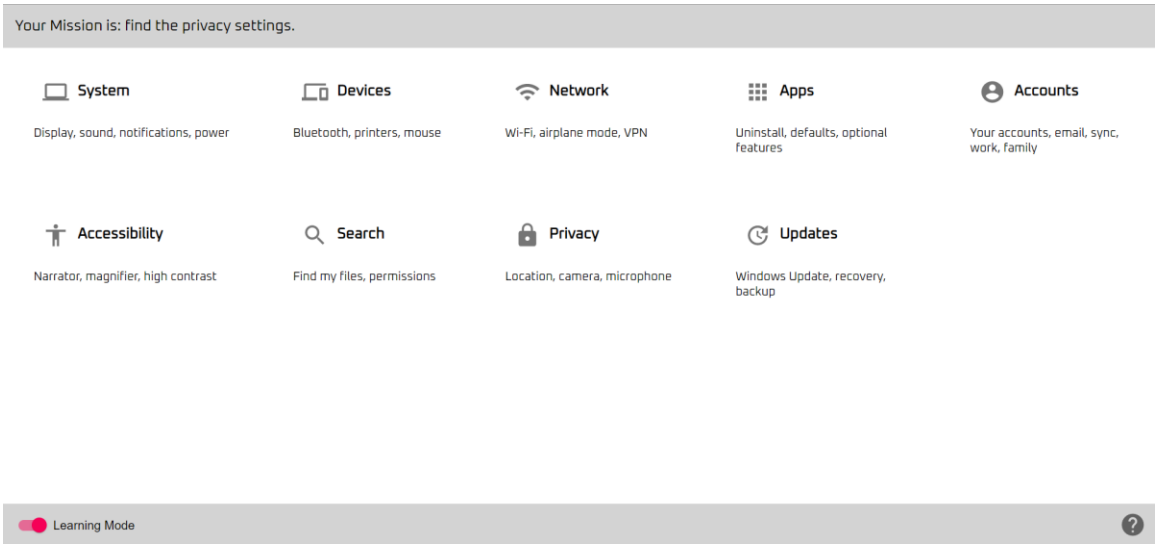

Figure 6.3 Screen capture of the finding a setting mission.

6.5 Game Modes

There are two game modes the user can play in: learning mode and regular mode.

6.5.1 Learning Mode

The goal of learning mode is to teach users how to complete each interaction. The user is presented with information about how the play the game and more information about each mission. When a user begins the game, learning mode is on by default so that all users will see information about how to play the game. This also means more users will play the game with learning mode on, since they must opt-out of using it instead of opting in, and therefore have easier access to learning than if they had to figure out how to turn learning mode on themselves.

In the beginning of the game, they are presented with modals that explain the buttons on the interface. On the navigation bar, there is a switch for turning on and off learning mode and a question mark icon for opening a hint. The modal above the switch explains what learning mode is and the modal above the question mark icon explains that this icons represents a hint, and by clicking the icon they will get more information on how to complete their mission.

At the end of each mission, users see an explanation of the mission, how long the mission took them, their sequence of moves, and an optimal sequence of moves. An example can be seen in the screen capture below.

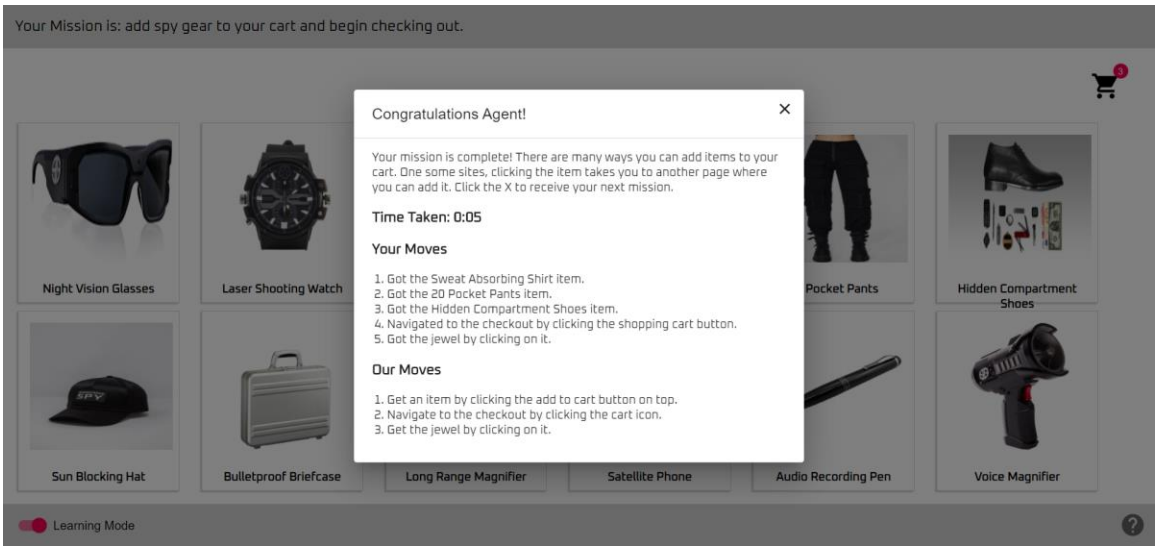

Figure 6.4 Screen capture of information presented at the end of a mission.

The explanation of the mission gives users context of where the interaction would occur in other places, an explanation of a different way to complete the interaction, or a longer explanation of the interactions they did to complete the mission. This is meant to give the user some background on the mission so if they were able to find the jewel by luck or without understanding their moves, the explanation would help explain what they did. The time taken is displayed so that users can see how they progress through missions. If a user were to play the game multiple times, they could see how their time taken per mission changes as they become more comfortable with the interactions required. It also indicates the complexity of the mission, as later missions should take more time than missions in the beginning. All the user's moves, including those that do not lead directly to a jewel, are displayed so they can be compared to the optimal sequence of moves. By comparing, users can understand where they got stuck and learn how to complete the interaction in the future. The optimal sequence of moves is based on the moves that someone with a technical background would take. All moves are categorized by learn, move, get, and change to build the users' problem-solving skills by giving them examples of these interactions in different contexts. Users can begin to learn the pattern of which interactions fit into each category which they can use in more complex missions to understand which interactions they need to complete to find the jewel.

6.5.2 Regular Mode

In regular mode, the above information is suppressed. The user can move through the game by simply completing each mission. Regular mode is intended to be used when users are using the game as an assessment. Since the goal is to determine what their

current digital competencies are, no additional information is provided that could influence their score. However, regardless of mode users have access to the hint button.

6.6 User Interaction Categorization

As mentioned in Section 4, a goal of the Digital Skills Framework is to give older adults a language they can use to describe their technical problems. The four categories: learn, get, navigate, and change encompass a common set of high-level user goals for their interactions with a user interface. This categorization was developed to help older adults communicate their technical problems and increase their problem-solving skills by relating interactions with user interfaces to interactions they are familiar with from outside technology. For example, users are familiar with learning information from a book, getting a shirt in a retail store, navigating to a friend's house on the road, or changing their address book. These high-level goals can be transferred to a problem in a user interface and users can identify what objectives they must complete to accomplish their goal.

Some user interface interactions are simple, such as scrolling to the bottom of a page, while other are more complex, such as resetting a password. Problems with these complex interactions can be hard to describe, especially when the user does not know where to begin. The goal of resetting a password is comprised of many objectives, such as determining the inputted password is wrong, finding where to reset the password, creating a new password, and then logging in with the new password. When a user is having trouble with a complex action, they may not understand what objectives it is composed of, or they may not understand the next objective that needs to be completed. By framing interactions in the above categories, users can understand what objectives need to be completed and in what order. The goal of resetting a password can be reframed as learning the inputted password is wrong, navigating to where the password can be reset, changing the password, navigating to the login page, and getting the desired information after logging in. This flow forces the user to think sequentially and breaks the problem of resetting a password into manageable parts.

These complex interactions are implemented in missions towards the end of the game since they require the user to understand how to complete simple, atomic interactions that are testing in the beginning. Along with the mission for resetting a password, the missions for changing a setting, using a shopping cart, and navigating around a map are examples of complex missions that can be broken down into many objectives.

Consider the mission for changing a setting. The high-level goal is for a user to find a setting of interest and change it. This can be complex when the application has many settings buried in many menus, since the user must first determine where the setting they want to change is located. This is an example where the line between learn and navigate can be blurred. In the language, navigate represents the action a user must take to move, such as clicking a button. Therefore, a user may learn something before and after making a move. A depiction of the objectives needed to complete this goal can be seen below.

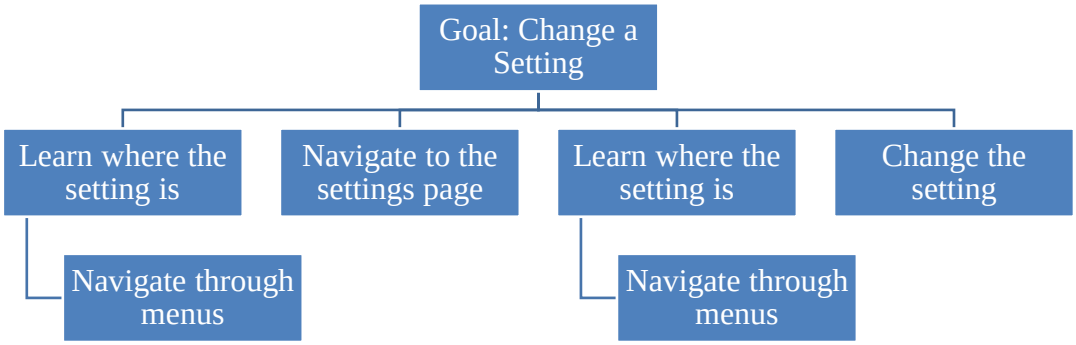

Figure 6.5 Objectives for changing a setting.

Consider the mission for using a shopping cart. The high-level goals are for a user to put an item in their shopping cart and check out. In the Digital Skills Framework, these goals are broken up into two missions: one where the user places items in a cart and another where the user removes an item from the cart and then checks out. The objectives of these missions are depicted in the figure below.

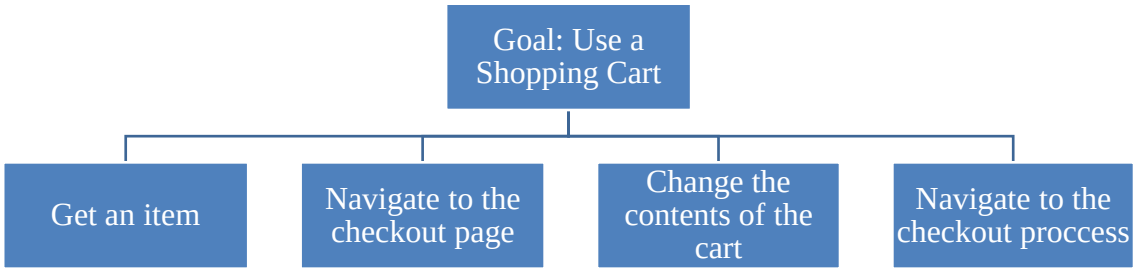

Figure 6.6 Objectives for using a shopping cart.

Finally, consider the mission of navigating around a map. The high-level goal is to determine where a specific place is located. This requires a series of interactions that are not necessarily intuitive, such as dragging or zooming on the map using a mouse. It can also be difficult if the map's default view does not include the place the user is looking for.

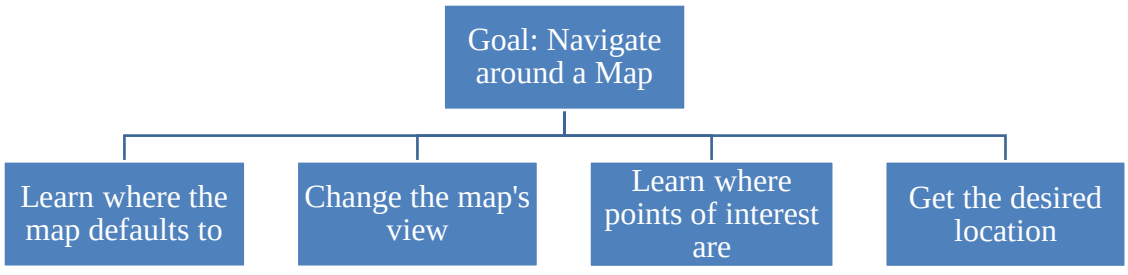

Figure 6.7 Objectives for navigating around a map.

The simple, atomic interactions in the beginning of the game can be used to create complex interactions. As seen in the mission for using a shopping cart, two or more simple interactions can be combined in sequence to create a complex mission that requires users to complete objectives before achieving their high-level goal. For example, the missions for favoriting an item and using a shopping cart could be combined into a mission where the user must favorite an item, retrieve that item from their favorites and add it to their cart, then checkout with that item. The objectives of the mission would be to get an item by favoriting it, navigating to the check out page, changing the contents of the cart by adding the favorited item, and navigating to the checkout process by clicking the checkout button. Using atomic missions as the building blocks for a complex mission tests ties the framework together, since users will recognize the objectives from simple missions they have already completed. It also demonstrates how complex interactions can be broken down into a sequence of simple interactions that the user is already confident in completing. This way when a user is faced with a complex problem, they will have the confidence and knowledge they need to complete it.

6.7 Logging

Displaying the time taken to complete missions and a user's sequence of moves requires that their interactions be logged. A backend logging service was developed using SpringBoot, which is a Java-based framework for creating microservices. SpringBoot was chosen because of its popularity, ease of use, and depth of documentation. The server handles HTTP requests made from the Digital Skills Framework web application. The service handles logging user interactions and time separately by splitting them into two domain classes.

6.7.1 User Interactions

The logging domain allows a user's interaction to be logged when they click or hover over an element on the screen. Each element has a unique identifier associated with it that tells the service was interaction was performed in a human readable form. Therefore, anyone who views a set of user interactions can understand what buttons where clicked and how the user moved through the game. The current time, the element's identifier, and the mission the user is on is sent to the service when the user performs an interaction. The service compiles and stores all interactions associated with each mission. Then, when the user reaches the end of a mission, the service returns all the interactions associated with that mission and this information is displayed. The service also allows the list of interactions to be cleared, which is done when a user begins a game. This way, no interactions from the previous game are displayed when the user plays again.

6.7.2 Time

The time domain allows the total time taken on each mission and the game overall to be calculated. When a user begins a mission, the current time and the mission number is sent

to the service to be stored. Then when the mission is complete the current time and mission number is sent, and the total time taken on the mission is calculated. This information is returned from the service and displayed to the user. Similarly, when the user begins the game, the current time is sent to the service and stored. When the user finishes their final mission, the current time is sent and is used to calculate the total time the user spent playing the game. This information is displayed on the final screen of the game. Like the logging domain, the time domain also allows the list of times taken per mission and time taken in total to be cleared. This also occurs when the game begins so that there is no leftover data from the previous game that could impact the time calculations.

7 Future Work

There are many improvements that can be made to the Digital Skills Framework. One goal is to better categorize a user's moves into the groups learn, move, get, and change. This categorization would be displayed at the end of each mission, where the sequence of the user's moves is displayed currently. For example, if a user is going down a path in a mission that does not lead to a jewel, an explanation of why that path was incorrect would be displayed at the end of the mission. This will help users solve the problem again when they see it in a different context. Categorizing user moves into these four fundamental groups will help users better understand the strategies they must use when interacting with an interface for the first time or when they are trying to solve a problem. It also gives them a framework for communicating their problems to other people, as these groupings are easily for people with technical and nontechnical backgrounds to understand.

Another improvement that can be made is showing the user's moves and the correct sequence of moves in a visual way. If the user did not understand a part of the mission, simply reading the correct steps needed to finish it might not be sufficient to help them understand. Showing their moves and the correct sequence of moves in a visual way, such as in a graph or through a video, would make it easier for a user to understand why the correct sequence of moves is correct.

Another goal of the Digital Skills Framework is for the framework to be used as an assessment. Therefore, the user must be given a score when they have completed all missions. The score would help tutors understand a user's knowledge base and help them narrow in on the skills that need to be worked on. More research must be done to understand what metrics would go into calculating the user's score. Initially, time was considered as a metric but has since been discarded because the goal is not to teach users to be fast at operating an interface, but rather understand how to operate it. It is difficult to determine a user's understanding based on the amount of time they spend on a mission. Part of completing this goal includes adding to the logging system. Currently what users click or hover on and the timestamp of the interaction is recorded, but where the user's mouse moves and how long they hover for could be added to gain a better understanding of how a user moves through the missions.

The Digital Skills Framework was developed to be played on personal computers, not tablets and other smaller technology. Ensuring the game can be used properly on smaller devices would help users whose primary device is tablet size or smaller. Instead of making users play the game on a personal computer, which may be foreign to them, they could play the game on the device that they feel most comfortable with. This would better simulate the interactions they have on their device and speed up their learning.

Finally, more complex missions should be added to the Digital Skills Framework. As mentioned in section 6.6, complex missions can be created by combining multiple simple, atomic missions together. These complex missions would test the user's ability to

move through a sequence of objectives to complete the higher-level goal. Multiple learn, navigate, get, and change interactions would be needed for the user to complete the mission, and there would also be ambiguity in the way the user could complete it. These missions would put together the skills users built through previous missions to complete complex interactions that are needed to successfully operate a computer today.

8 Conclusion

Older adults are continuing to adopt technology as it becomes an important part of our lives. They are equally as likely as people aged 23 to 54 to own a tablet and they have increased their adoption from 25% to 68% from 2011 to 2019 (Vogels, 2019). However, engaging with technology can cause anxiety, and these older adults may not be confident in their digital skills. Therefore, it is critical to teach older adults the skills they need to navigate the digital world and increase their confidence to do so.

The Digital Skills Framework provides a tool for understanding digital competence through a gamified assessment of skills on a set of user interface interactions. Understanding an older adult's current digital competencies can aid in tutoring settings, whether formal or informal. A tutor would be able to understand what skills need to be worked on, instead of spending extra time diagnosing where skill gaps exist.

The framework also provides a tool for teaching digital competence by giving older adults a safe place to explore and experiment with a set of user interface interactions. It provides users feedback on how they have completed the interaction and explains how the interaction could be found in different contexts or completed in different ways. This allows older adults to build problem-solving skills and confidence as they complete interactions.

Finally, the Digital Skills Framework gives older adults a language to describe their technical problems. This language: learn, get, navigate, and change helps older adults define technical problems in familiar terms. It also shows how complex problems can be broken down into simple, manageable parts and shows the sequence in which these parts must be completed to solve the problem.

9 References

Abramov, D. (n.d.). *Redux*. Retrieved from Redux: A Predictable State Container for JS Apps: https://redux.js.org/

Carretero, S., Vuorikari, R., & Punie, Y. (2017). *European Commission.* Retrieved from The Digital Competence Framework for Citizens: https://publications.jrc.ec.europa.eu/repository/bitstream/JRC106281/web-digcomp2.1pdf_(online).pdf

Carter, J. (1986). A Taxonomy of User-oriented Function. *International Journal of Man-Machine Studies,* 195-292.

Carter, J., Lukey, J.-A., & Schweighardt, M. (1991). The Action-Modifier-Object-Attribute (AMOA) Classification of User-oriented Functions. *International Journal of Man-Machine Studies,* 1-22.

Digital Dannelse. (2009). *The Digital Wheel of Competence*. Retrieved from Digital Dannelse: https://digital-competence.eu/digitalekompetencer/

Erikson Institute. (2016). *Erikson Institute.* Retrieved from Technology and Youn Children in the Digital Age: https://www.erikson.edu/wp-content/uploads/2018/07/Erikson-Institute-Technology-and-Young-Children-Survey.pdf

Facebook Open Source. (n.d.). *React*. Retrieved from React: A Javascript Library for Building User Interfaces: https://reactjs.org/

Ferrari, A. (2012). *European Commission.* Retrieved from Digital Competence in Practice: An Analysis of Frameworks: https://pdfs.semanticscholar.org/851f/ebe72df176a16ad6e26b00ff5df35520da34.pdf

Interaction Design Foundation. (n.d.). *User Interface (UI) Design Patterns*. Retrieved from Interaction Design Foundation: https://www.interaction-design.org/literature/topics/ui-design-patterns#:~:text=User%20interface%20(UI)%20design%20patterns%20are%20reusable%2Frecurring%20components,the%20specific%20context%20of%20use

Kumar, S., Ureel, L., King, H., & Wallace, C. (2013). Lessons from our elders: identifying obstacles to digital literacy through direct engagement. *PETRA '13: Proceedings of the 6th International Conference of PErvasive Technologies Related to Assistive Environments,* (pp. 1-8).

Lumsden, J., Edwards, E. A., Lawrence, N. S., Coyle, D., & Munafo, M. R. (2016). Gamification of Cognitive Assessment and Cognitive Training: A Systematic Review of Applications and Efficacy. *JMIR Serious Games*.

Material UI. (n.d.). *Material UI*. Retrieved from Material UI: https://material-ui.com/

Nielsen, J. (2016). *Nielsen Norman Group*. Retrieved from The Distribution of Users' Computer Skills: Worse Than You Think: https://www.nngroup.com/articles/computer-skill-levels/

Steelman, K. S., Tislar, K. L., Ureel, L. C., & Wallace, C. (2016). Breaking Digital Barriers: A Social-Cognitive Approach to Improving Digital Literacy in Older Adults. *HCI International 2016 - Posters' Extended Abstracts*, (pp. 445-450).

Steelman, K. S., Tislar, K. L., Ureel, L. C., & Wallace, C. (2017). Eliciting Best Practices in Digital Literacr Tutoring: A Cognitive Approach Task Analysis Approach. *Human Aspects of IT for the Aged Population. Aging, Design and User Experience*, (pp. 447-460).

Steelman, K., & Wallace, C. (2017). Identifying and Mitigating Technology-Related Anxiety. *ACM SIGAC-CESS Conference on Computers and Accessibility.* Baltimore.

Tidwell, J. (2011). *Designing Interfaces.* O'Reilly Media.

Toxboe, A. (n.d.). *UI Patterns*. Retrieved from Design patterns: http://ui-patterns.com/patterns

Van Welie, M. (2008). *Patterns in Interaction Design*. Retrieved from Pattern Library: http://welie.com/patterns/

Vogels, E. A. (2019). *Pew Research Center*. Retrieved from Millennial stand out for their technology use, but older generations also embrace digital life: https://www.pewresearch.org/fact-tank/2019/09/09/us-generations-technology-use/

A Accessing the Digital Skills Framework

The Digital Skills Framework web application can be viewed using the following link: https://digital-skills-assessment.netlify.app/

The Digital Skills Server can be accessed using the following link: https://digital-skills-server.herokuapp.com/

B Copyright documentation

All images in this document are screen captures from the web application, Digital Skills Framework, that I have developed. The Digital Skills Framework is hosted on a Netlify server.