

**SIMPLIFYING SUSTAINABILITY GAME DESIGN: A USABILITY EVALUATION OF
THE MAKAHIKI VIRTUAL MACHINE INSTALLATION AND THE SMART GRID
GAME DESIGNER**

A Senior Honors Undergraduate Project Thesis

Presented to

The Faculty of the Department of Information and Computer Science

University of Hawai'i at Mānoa

In partial fulfillment of the requirements for the degree

Bachelor of Computer Science with Honors

By

Jordan Takayama

Thesis Committee:

Philip Johnson, Chairperson

Dennis Streveler

Depeng Li

April 2014

Version 1.0.0

This project is dedicated
to my family.

Acknowledgments

This project would not have been possible without the love and encouragement of my family, and I will always be grateful for the support you have given me throughout this project and throughout my academic journey. Thank you.

My academic advisor, Dr. Philip Johnson, was an instrumental voice of reason when I was unsure of what direction to take my research in. His experience in software development, sustainability research, and gamification was instrumental in starting my research efforts, refining the design of my study, and guiding me through the IRB approval process. My thanks also goes out to the other members of the Collaborative Software Development Laboratory, especially Carleton Moore and Yongwen Xu, who provided valuable insights as I worked to develop the scripts and documentation which became the Vagrant installation tool.

I would like to thank my thesis committee members, Professor Depeng Li and Professor Dennis Streveler, for putting forth the effort and taking the time to evaluate this thesis.

Thanks also go out to Gerald Lau, assistant faculty specialist, who has advised me throughout my time in the Information and Computer Science department and convinced me to begin an undergraduate research project in the first place.

I thank the Undergraduate Research Opportunities Project for providing funding for software and test subject compensation. Your support for this research is much appreciated.

I would like to thank Dr. Yvonne Chan, Joseph Mendez, and Kiana Shiroma for their guidance throughout the University of Hawai‘i at Mānoa Honors Program and the Upper Division Honors Project process. The Honors Program has many staff and faculty that truly care about their students, and without your help I would never have been able to take advantage of the opportunities available at the University of Hawai‘i.

Abstract

The usability of an application is a measure of how effectively it can be used to perform the tasks it was designed for in its target environment. A user interface – the toolbars, menus, and other elements that control an application – determines how quickly and correctly users can complete tasks. Makahiki is an application framework for designing serious games (games which teach a serious subject) focused on energy conservation, recycling, and clean energy issues. Two features were added to Makahiki in response to user feedback: support for a cross-platform installation method for virtual machines, and a simplified drag-and-drop graphical user interface called the Smart Grid Game Designer (SGG). Usability testing data and feedback on these new features was compared to data and feedback from the previous iteration of Makahiki to determine the effect of these features on the user experience. It was found that the virtual machine installation produced significant improvements in user experience and configuration time. However, users who tested the Smart Grid Game Designer reported issues in understanding Makahiki's "predicate system" of relationships between game tasks that were similar to issues reported by users of the previous iteration of Makahiki.

Keywords

Virtual Machine, Vagrant, Smart Grid Game, Serious Game, Game Design, Usability, Kukui Cup, Makahiki

Table of Contents

Acknowledgments	ii
Abstract	iii
List of Tables	vii
List of Figures	xi
1. Introduction	1
1.1. Overview of Makahiki	1
1.2. Overview of Usability and HCI	1
1.2.1. User Interfaces and Usability	2
1.2.2. Interface Usability and Productivity	2
1.3. Gamification and Serious Games	3
1.3.1. Makahiki as a "Serious Game" Framework	4
1.3.2. Overview of Makahiki Functionality	4
1.3.2.1. Makahiki Installation and Developer Experience Fragmentation	5
1.3.2.2. Responsibilities of Makahiki-Based Challenge Administration	7
1.3.2.3. Makahiki User Interfaces for the Smart Grid Game	8
1.3.2.4. Makahiki, the Kukui Cup, and Other Sustainability Challenges in Hawai‘i ..	10
2. Thesis	11
3. Methodology	11
3.1. Action Research and Iterative Development	11
3.1.1. Usability Testing Procedure	12
3.2.1. Testing Equipment	13
4. Results	14

Table of Contents, continued

4.1. Prior Experience Assessments	14
4.2. Comparison of Manual and Vagrant Installation Methods	17
4.2.1. Difficulty Ratings for the Vagrant Installation	20
4.2.2. Comparison of Usability Issues for the Manual and Vagrant Installation	21
4.2.3. Analysis of Usability Issues in Vagrant Installation Screencast Data	23
4.3. Comparison of the 2013 Web Administration Interface and the SGG Designer ..	24
4.3.1. Difficulty Ratings for the SGG Designer	27
4.3.2. Comparison of Usability Issues for the 2013Web Administration Interface and the SGG Designer	28
4.3.3. Analysis of Usability Issues in SGG Designer Screencast Data	30
4.4. Task Time and Difficulty Rating Analysis	31
5. Discussion of Results	33
5.1. Vagrant Installation Usability Testing	33
5.2. SGG Designer Usability Testing	34
5.3. Limitations	35
5.4. Objectives for Future Research	36
6. Conclusion	37
Appendix A. Survey Instruments for Group 1 in Spring 2013	38
Appendix A.1. Makahiki Local Installation Assessment	38
Appendix A.2. Makahiki Configuration and Management Assessment	43
Appendix B. Survey Instruments for Group 2 in Spring 2014	49
Appendix B.1. Vagrant Installation Tool Assessment	49
Appendix B.2. Makahiki SGG Designer Assessment	55

Table of Contents, continued

Appendix C. Selected Data from 2013 Usability Testing	65
Appendix C.1. Makahiki Local Installation Log: Survey Data	65
Appendix C.1.1. Task Time Data	65
Appendix C.1.2. Survey Form Comments	69
Appendix C.2. Makahiki Configuration and Management Log: Survey Data	75
Appendix C.2.1. Task Time Data	75
Appendix C.2.2. Survey Form Comments	76
Appendix D. Selected Data from 2014 Usability Testing	78
Appendix D.1. Makahiki Vagrant Installation Log Data.....	78
Appendix D.1.1. Survey Data.....	78
Appendix D.1.2. Screencast Activity Logs	91
Appendix D.2. Makahiki SGG Configuration Log Data	100
Appendix D.2.1. Survey Data	100
Appendix D.2.2. Screencast Activity Logs	114
Bibliography	120

List of Tables

Table 1. Supported Makahiki Installation Methods as of Spring 2013	5
Table 2. Self-Reporting of Computer Experience	15
Table 3. Self-Reporting of Makahiki Experience	15
Table 4. Group 1: Operating Systems Used for Makahiki Manual Installations	16
Table 5. Group 2: Operating System the Subject Was Most Experienced With	16
Table 6. Group 2A: Vagrant Installation: Self-Reporting of Application Experience	17
Table 7. Group 2B: SGG Configuration: Self-Reporting of Browser Familiarity	17
Table 8. Comparison of Manual Installation and Vagrant Installation	18
Table 9. Vagrant Installation Time as a Percentage of Manual Installation Time	19
Table 10. Completion Time for Individual Tasks in the Vagrant Installation (Minutes) ..	20
Table 11. Difficulty Ratings By Task	20
Table 12. Summary of Subject Comments for Group 1 (Manual Installation)	21
Table 13. Summary of Subject Comments for Group 2A (Vagrant Installation Tool)	22
Table 14. Issues Identified in the Screencast	23
Table 15. Comparison of the 2013 Interface and SGG Configuration Methods	25
Table 16. SGG Designer Time as Percentage of 2013 Interface Time for Selected Tasks ..	25
Table 17. SGG Configuration Completion Times in Minutes	26
Table 18. Difficulty Ratings By Task	28
Table 19. Summary of Subject Comments for Group 1 (2013 Interface)	28
Table 20. Summary of Subject Comments for Group 2B (SGG Designer)	29
Table 21. Issues Identified in the Screencast	30
Table 22. Mean Task Time and Difficulty Correlation	31
Table 23. Vagrant Correlation between Task Completion Time and Difficulty Warning .	32

List of Tables, continued

Table 24. SGG Correlation between Task Completion Time and Difficulty Warning	32
Table C1.1.1. Subject 1 Task Completion Times	64
Table C1.1.2. Subject 2 Task Completion Times	65
Table C1.1.3. Subject 3 Task Completion Times	65
Table C1.1.4. Subject 4 Task Completion Times	66
Table C.1.1.5. Subject 5 Task Completion Times	66
Table C.1.1.6. Subject 6 Task Completion Times	67
Table C.1.1.7. Subject 7 Task Completion Times	67
Table C.1.1.8. Subject 8 Task Completion Times	68
Table C.1.2.1. Subject 1 Survey Form Comments	68
Table C.1.2.2. Subject 2 Survey Form Comments	69
Table C.1.2.3. Subject 3 Survey Form Comments	70
Table C.1.2.4. Subject 4 Survey Form Comments	70
Table C.1.2.5. Subject 5 Survey Form Comments	71
Table C.1.2.6. Subject 6 Survey Form Comments	71
Table C.1.2.7. Subject 7 Survey Form Comments	72
Table C.1.2.8. Subject 8 Survey Form Comments	72
Table C.2.1.1. Subject 1 Task Completion Times	75
Table C.2.1.2. Subject 2 Task Completion Times	75
Table C.2.1.3. Subject 3 Task Completion Times	75
Table C.2.1.4. Subject 4 Task Completion Times	75
Table C.2.1.5. Subject 5 Task Completion Times	75
Table C.2.1.6. Subject 6 Task Completion Times	76

List of Tables, continued

Table C.2.1.7. Subject 7 Task Completion Times	76
Table C.2.2.1. Subject 1 Survey Form Comments	76
Table C.2.2.2. Subject 2 Survey Form Comments	76
Table C.2.2.3. Subject 3 Survey Form Comments	77
Table C.2.2.4. Subject 4 Survey Form Comments	77
Table C.2.2.5. Subject 5 Survey Form Comments	77
Table C.2.2.6. Subject 6 Survey Form Comments	78
Table C.2.2.7. Subject 7 Survey Form Comments	78
Table D.1.1.1. Subject 13 Survey Form	79
Table D.1.1.2. Subject 5 Survey Form	80
Table D.1.1.3. Subject 15 Survey Form	82
Table D.1.1.4. Subject 1 Survey Form	84
Table D.1.1.5. Subject 3 Survey Form	86
Table D.1.1.6. Subject 7 Survey Form	88
Table D.1.1.7. Subject 9 Survey Form	89
Table D.1.2.1. Subject 13 Screencast Activity Log	91
Table D.1.2.2. Subject 5 Screencast Activity Log	93
Table D.1.2.3. Subject 15 Screencast Activity Log	94
Table D.1.2.4. Subject 1 Screencast Activity Log	95
Table D.1.2.5. Subject 3 Screencast Activity Log	96
Table D.1.2.6. Subject 7 Screencast Activity Log	98
Table D.1.2.7. Subject 9 Screencast Activity Log	99

Table D.2.1.1. Subject 18 Survey Form	101
Table D.2.1.2. Subject 20 Survey Form	103
Table D.2.1.3. Subject 12 Survey Form	105
Table D.2.1.4. Subject 4 Survey Form	108
Table D.2.1.5. Subject 2 Survey Form	110
Table D.2.1.6. Subject 6 Survey Form	112
Table D.2.2.1. Subject 18 Screencast Activity Log	114
Table D.2.2.2. Subject 20 Screencast Activity Log	115
Table D.2.2.3. Subject 12 Screencast Activity Log	117
Table D.2.2.4. Subject 4 Screencast Activity Log	117
Table D.2.2.5. Subject 2 Screencast Activity Log	118
Table D.2.2.6. Subject 6 Screencast Activity Log	118

List of Figures

Figure 1. Partial code from the provisioning script	7
Figure 2. Sample navigation process to add a level in the 2013 interface	8
Figure 3. Sample navigation process to add a level in the SGG interface	9
Figure 4. An example of the "control-c" error	24
Figure 5. An example of the "literal file path" error	24
Figure 6. A slug name being used without a predicate	31

1. Introduction

1.1. Overview of Makahiki

The Makahiki serious game design framework is a software application which provides tools and activity modules that can be used to create games which educate players about sustainability and renewable energy issues. Makahiki tracks player participation and point earnings from each activity and awards prizes based on the criteria of the contest administrators. Though Makahiki has been deployed to support sustainable games in the past, two things which have prevented it from achieving wider use has been the difficulty of installing a functioning instance of the application as a software developer, and the difficulty of using its user interface to design a functioning sustainability education game. Based on usability testing data collected for the Makahiki software as it existed in Spring 2013, I have implemented an "installation tool" for Makahiki that streamlines the process for creating a functioning instance of the Makahiki application suitable for software development work. Another member of the Makahiki development team, Carleton Moore, has implemented a "Smart Grid Game Designer" feature to streamline the process of creating new sustainability education games. As both of these features were created with the intent to fix bugs, implement desired functionality, and address areas of user confusion, usability testing was necessary to evaluate the degree to which the changes accomplished their goals. This project evaluated the relative usability of different user interface styles and improved the usability of a product that helps people educate others about sustainability.

1.2. Overview of Usability and HCI

The design of user interfaces falls within the domain of human-computer interaction, or HCI. According to the Association for Computing Machinery (ACM), HCI is

SIMPLIFYING SUSTAINABILITY GAME DESIGN

the study of the design, evaluation, and implementation of computer systems that are designed to interact with humans [6]. The ACM committee of Hewett et al. defined user interface elements and dialogue interaction techniques to include icons, menus, forms, and speech and video input and output methods through which a user interacts with an application. The ACM also noted that the quality of a user interface had become a more important factor in the sales of an application over time [6]. User interface design is important in software engineering because the clarity or obtuseness of an application plays a large role in determining how useful it is and how useful customers perceive it to be.

1.2.1. User Interfaces and Usability

It is generally agreed that an interface's usability is evaluated based on the ease or difficulty with which it can be used to perform the tasks that its application was designed for, in the environment it was designed for. The level of difficulty is dependent on how well the interface satisfies its users' needs in addition to environmental constraints [5, 6]. This perception is dependent on the user's qualitative perceptions of the system and the processes required to interact with it via the interface [10]. These perceptions can result in more difficult systems being rated as less useful [2]. The usability of an application is heavily dependent on how easy the application's user interface is to use.

1.2.2. Interface Usability and Productivity

The design of a user interface can have a significant effect on the productivity and accuracy of its users. McFarlane's study found that the group which was randomly interrupted by the user interface and forced to deal with the interruptions immediately had the highest number of key-pressing errors and the greatest number of incorrect actions in the interrupting task. This group performed worse than the group interrupted at regular intervals, the group allowed to delay

their interruptions, and the group for which a scheduling program delayed interruptions until the main workload decreased [11]. Fang and Holsapple's study of website user interfaces found that a user interface with options organized by highest frequency of use was correlated with users completing tasks more quickly and correlated with a higher ratio of correct answers to incorrect answers, as compared to a user interface in which options were organized only by subject [5]. In addition, the evaluation by Wanderer et al. of two graphical interfaces for an anesthesia management system found that the interface which was redesigned to include color-coded feedback and context-sensitive menu items required fewer steps to complete medical documentation and had higher accuracy [12]. In these studies, differences in the design of an interface produced significant differences in accuracy and speed, which affected the productivity of an application's users. This means that it is possible for changes in the design of a user interface to significantly increase the productivity of its users. These studies confirm that differences in the design of a user interface have been found to be correlated with statistically significant differences in a user's perceived experience with an application and in a user's actual performance with it.

1.3. Gamification and Serious Games

Makahiki is a framework for designing "serious games" which teach players about sustainability using the practices of gamification. Based on Deterding et al.'s definition of a game as play structured by rules and competition in pursuit of a goal, gamification is defined as the use of elements of game design for non-game elements of any digital or physical application or product [3]. Within the game industry, "gamification" refers to the increasing adoption and use of video games and video game-like applications in daily life, or to the idea that game elements can be used to make non-game products and services more engaging to consumers [3]. In

SIMPLIFYING SUSTAINABILITY GAME DESIGN

addition, a serious game is any game which uses game elements, mimics the structure of a game, but is not designed for entertainment purposes. Game elements can include badges, leader boards, levels, and other interface design elements often associated with games. Serious games have been used in education and military training programs for several millennia, but only became widely used during the twentieth century [3]. Makahiki is a tool for creating serious games that educate players about sustainability.

1.3.1. Makahiki as a “Serious Game” Framework

The goal of Makahiki as a serious game framework is to allow the designers of sustainability challenges to create competitions which increase players’ knowledge of issues related to renewable energy while incentivizing the development of sustainable habits. The developers of Makahiki were motivated by a desire to reduce dependence on oil by educating players about the potential of renewable energy, stating that “Moving away from petroleum involves technological, political, and social changes, requiring citizens to not only think differently, but behave differently with respect to energy policies, methods of generation, and their own consumption [9].” Makahiki evaluates the performance of players and teams by measuring team energy use against predefined baselines, tracking user-earned points from activities, and tracking the activity participation percentages of individual teams.

1.3.2. Overview of Makahiki Functionality

The Makahiki framework is a software application designed to collect power and energy consumption data from sensors installed within a building’s electrical grid [8]. Challenge administrators expressed concerns about the high amount of time required to create a sustainability competition in Makahiki’s administrative user interface. Makahiki provides tools and a library of sample activities for creating and managing sustainability competitions.

1.3.2.1. Makahiki Installation and Developer Experience Fragmentation

Makahiki is an open-source project available under the MIT License through a file repository hosted by GitHub [13]. Once a developer downloads the source code for Makahiki onto a local computer, that developer must install dependencies so that Makahiki can function. This includes the Django web application framework; a PostgreSQL backend database; C and Python compilers; and software libraries for image processing, file parsing, and memory management. After installation, special configuration files must be edited to allow the installed software packages and applications to interact with each other [14]. Prior to the creation of the "installation tool," there were already five officially supported ways to install and configure Makahiki and its dependencies, as seen in Table 1.

Table 1. Supported Makahiki Installation Methods as of Spring 2013.

Base Operating System	Method
Ubuntu Linux	Native installation of software dependencies
Mac OS X	Native installation of software dependencies
Microsoft Windows	Dual-booting (Configuring a new disk partition for Ubuntu Linux) and installing dependencies on the Linux partition
Microsoft Windows	Installing software dependencies on an Ubuntu Linux virtual machine for VirtualBox provided by the developers
Microsoft Windows	Native installation of software dependencies

Three of these methods were for Windows alone. Notably, two of those methods installed Linux in order to avoid running Makahiki natively on Windows. The Makahiki documentation still states that "native installation of Makahiki on Windows is more complicated and error-prone than on a Unix [OS X or Linux] environment, because Makahiki uses two technologies (Django and Memcached) that are more difficult to install in a Windows environment [14]." The developers recommended that Windows developers try dual-booting into Windows and Linux or using a virtual machine rather than attempt a native Windows installation [14]. In addition, the only subject to use Mac OS X for a native installation in Group 1 in 2013 experienced so many

SIMPLIFYING SUSTAINABILITY GAME DESIGN

unique errors that the subject's times needed to be removed from the data set as an outlier (see Appendix C.1.2.,). While having complex operating-system-specific requirements is often unavoidable, it results in very different developer experiences for each operating system. Furthermore, installing a large number of software packages and editing settings values in configuration files leaves room for user error in steps which could be automated instead.

The "installation tool" evaluated in this thesis supports a cross-platform procedure for installing the Makahiki framework on an Oracle VirtualBox virtual machine managed by the Vagrant software. For maximum compatibility with 32-bit and 64-bit operating systems, the virtual machine used is currently a 32-bit version of Ubuntu Linux 12.04 LTS. The advantages of VirtualBox according to Saylor et al. are that it provides "a single unified platform for Windows, [Mac] OSX, and Linux," and is "both Free and Open Source [15]." Vagrant manages VirtualBox virtual machines from the command line, creating virtual machines and automatically installing software onto those machines by running commands listed in special text files called "provisioning scripts [16]." The use of the Vagrant-based installation tool provides a consistent way for users with different operating systems to have the same development environment, installed and configured by a Vagrant "provisioning script" file. The script can perform all the tasks that a manual installation would perform so long as it is used on a virtual machine that supports its shell script commands and keeps its configuration files in locations that match what the script expects. By using the Ubuntu 12.04 LTS virtual machine, the procedure ensures that the same virtual environment will exist each time that the provisioning script is run to install Makahiki on a new virtual machine. Source code from the provisioning script as of March 2014 appears in Figure 1 below.

```

# Begin libmemcached-0.53 installation
echo "Installing libmemcached-0.53: started at $(date)"
if [ ! -f /usr/local/lib/libmemcached.so ]
then
    echo "apt-get install -y build-essential"
    apt-get install -y build-essential
    echo "apt-get install -y g++"
    apt-get install -y g++
    echo "apt-get install -y libcloog-pp1-dev"
    apt-get install -y libcloog-pp1-dev
    echo "apt-get install -y libcloog-pp10"
    apt-get install -y libcloog-pp10
    # make creates symlinks, so it must occur in a non-shared folder.
    echo "mkdir /home/vagrant/makahiki-temp-downloads"
    mkdir /home/vagrant/makahiki-temp-downloads
    echo "cd /home/vagrant/makahiki-temp-downloads"
    cd /home/vagrant/makahiki-temp-downloads
    echo "wget http://launchpad.net/libmemcached/1.0/0.53/+download/libmemcached-0.53.tar.gz"
    wget http://launchpad.net/libmemcached/1.0/0.53/+download/libmemcached-0.53.tar.gz
    echo "tar xzf libmemcached-0.53.tar.gz"
    tar xzf libmemcached-0.53.tar.gz
    echo "cd libmemcached-0.53"
    cd libmemcached-0.53
    echo " ./configure"
    ./configure
    echo "make"
    make
    echo "make install"
    make install

```

Figure 1. Partial code from the provisioning script which builds and installs the libmemcached software library.

Though the Vagrant virtual machine installation is not able to serve as a production instance of Makahiki without extensive modification, it is a full installation of Makahiki that can support the creation of temporary sustainability games for software development use.

1.3.2.2. Responsibilities of Makahiki-Based Challenge Administration

Challenge administrators are responsible for configuring and managing game widgets through Makahiki's web browser-based user interface. Each widget is a software module which provides in-game functionality such as scoreboards, energy use displays, user activity submissions, the Smart Grid Game widget that links users to activity pages, or raffle prize allocation [8]. This incorporation of game-like elements such as a points system, statistical performance tracking, and physical prizes in the pursuit of environmentalist goals is what makes Makahiki useful for creating a "serious game." However, the educators and other non-IT participants who have designed content for Makahiki-based games in the past have been unable

SIMPLIFYING SUSTAINABILITY GAME DESIGN

to add content to their games directly. Instead, they have had to rely on their systems administrators or directly on the Makahiki developers themselves to add content to the system.

The Smart Grid Game Designer was intended to reduce the burden on administrators and simplify the game design process, with the goal of enabling people to add and modify activities for players with a minimum of prior knowledge about Makahiki and its underlying software.

1.3.2.3. Makahiki User Interfaces for the Smart Grid Game

The user interface for configuring the Smart Grid Game during the 2012-2013 Kukui Cup is a hierarchical menu system organized by subject [5]. It defines a set procedure for accessing any given configuration option, but does not bring frequently-used options to its top level.

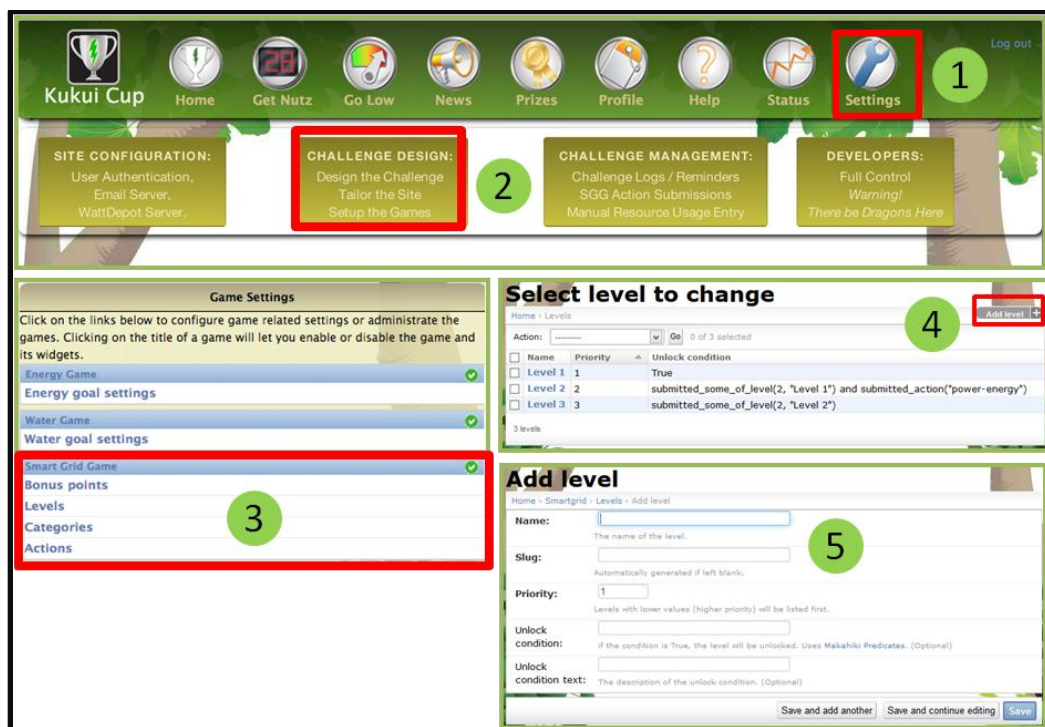


Figure 2. Sample navigation process to add a level in the 2013 interface. Starting from the Settings icon and traversing to the Add Level page takes five screens and four clicks on the areas outlined in red.

The Smart Grid Game Designer, developed in 2013 - 2014 by Carleton Moore of the Collaborative Software Development Laboratory, attempts to streamline the process of editing the Smart Grid Game by providing an alternate hierarchy that brings the features known to be

SIMPLIFYING SUSTAINABILITY GAME DESIGN

frequently used to edit the Smart Grid - level settings, categories, and access to the library of pre-existing activities - to the top level. This reduces the number of clicks required for certain actions, as demonstrated in Figure 3.

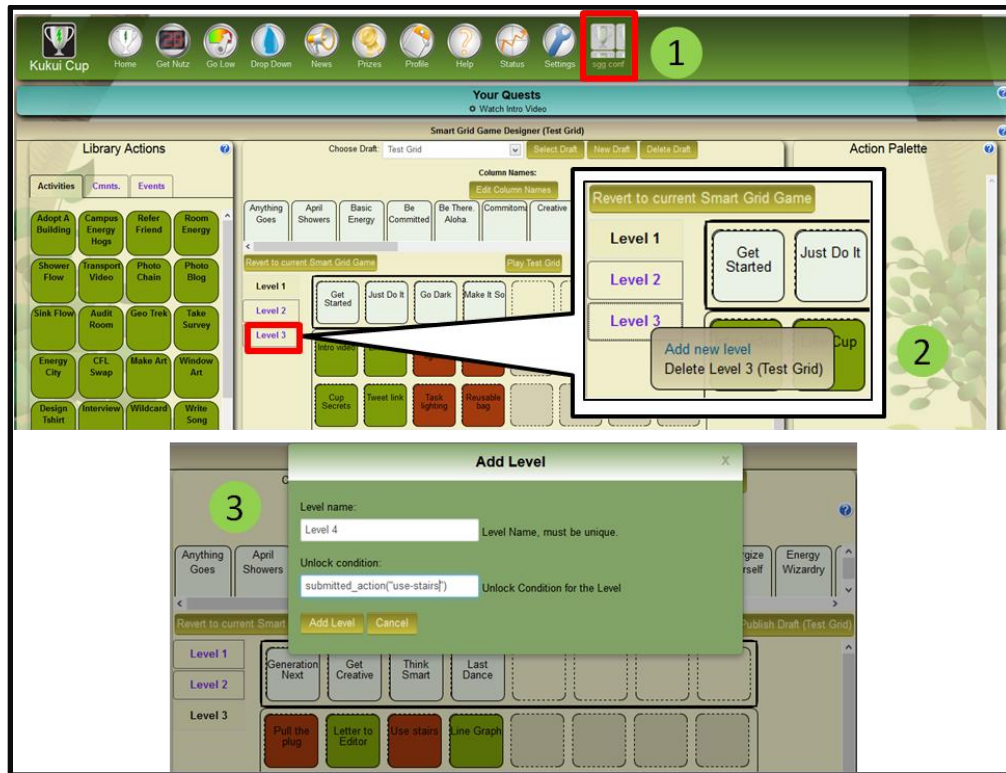


Figure 3. Sample navigation process to add a level in the SGG interface. Clicking on the "sgg conf" icon takes the user directly to the game grid they are currently working on in 2 clicks and 3 screens. When a new level is added, a popup appears that allows the user to set the level name and unlock condition.

In addition, the SGG separates the functions related to game design, which may be used by educators and others not responsible for administration of the Makahiki server itself, from configuration settings such as email and authentication that are related to systems administration. Administrators who are only responsible for the Smart Grid Game will be able to use the SGG Designer without having to navigate the "2013 interface."

1.3.2.4. Makahiki, the Kukui Cup, and Other Sustainability Challenges in Hawai‘i

By improving the usability of Makahiki, this project has the potential to make Makahiki usable by and usable for a wider audience of players and event organizers. Since 2011, the Makahiki framework has been used to manage the Kukui Cup competition at the University of Hawai‘i at Mānoa. The Kukui Cup is an energy conservation and sustainability competition that takes place in the first-year student dormitories at the University. Using a server running the WattDepot power meter data collection software, Makahiki gathers data from the power meters that have been installed in the central electrical panels on each floor, sampling data at regular intervals. Points are awarded to floors for low energy consumption and to individual players for earning points through the completion of tasks in the Smart Grid widget [1]. The annual cycling-out of the dormitories' students (who must move to other dormitories or off campus after the first year) provides a key research opportunity due to the constantly renewed experimental population [1]. An expansion of the use of Makahiki to other settings could provide new opportunities for research on attitudes regarding renewable energy and sustainable habits.

Since the first three-week Kukui Cup round in 2011, Makahiki has begun an expansion to other competitions at the collegiate level. The 2012 Kukui Cup ran through Spring 2013 [8]. In addition, the Makahiki framework was used for sustainability competitions at Hawai‘i Pacific University (HPU) and the East-West Center in 2012 [8]. Though future plans for expansion target schools at the elementary education level, these goals of reaching a wider audience will be difficult to realize until Makahiki becomes easier to use for people who are unfamiliar with the system but still want to create simple competitions [8].

2. Thesis

A redesigning of the Makahiki installation process and administrative user interface to address user-reported difficulties in creating new instances of Makahiki and configuring new sustainability challenge games will significantly affect the time required to configure challenges and the perceived ease of use of the administrative interface.

3. Methodology

3.1. Action Research and Iterative Development

The Makahiki application is developed in accordance with the iterative development model of "action research," a software engineering methodology which follows an iterative cycle of development and testing. The action research cycle begins with the identification, through surveys or other qualitative methods, of the conditions under which the software system is being used. This information is used to attempt to change those circumstances. Further qualitative research is then used to identify the changed circumstances created by that action [4]. This iterative approach is compatible with a typical software engineering design cycle. Hevner et al. [6] and Kushniruk and Patel [10] describe a development process which begins with the evaluation of a product based on data gathered from case studies, experiments, or simulations. This usability feedback is used to develop and test a new version of the product. The new version is then tested, which produces further usability feedback which can be used to redesign other features of the system. This methodology, over many cycles of development, leads to further refinement of the software product and the methods used to determine its usability.

3.1.1. Usability Testing Procedure

The usability testing procedure for the study was based on the methods used by Kushniruk and Patel [10] and Fang and Holsapple [5], in which test subject groups which were using different user interfaces performed the same task in each interface and were surveyed to determine the time it had taken them to complete each task and to record their qualitative opinion of the interface's usability. The two test subject groups in this study are distinguished by the version of Makahiki which they used.

Group 1, composed of students from a Spring 2013 graduate-level course in Information and Computer Science, completed a representative set of configuration tasks that installed a new instance of Makahiki on their personal computers, or on a VirtualBox Linux virtual machine running on their personal computers. Once each student had installed an instance of Makahiki, that instance was used to carry out a defined set of configuration tasks to create a new sustainability game. This was carried out over the course of two course assignments, and students were given approximately seven days to complete each one. Throughout this process, the students could request online assistance from members of the Makahiki development team. After completing each set of tasks - the first set related to the installation of Makahiki, and the second set related to designing a sustainability game in Makahiki - each student filled out a usability assessment survey. Each survey question asked the test subject to record the time taken to complete each task and to describe any problems they experienced while using Makahiki. The surveys are reproduced in Appendix A.1 and A.2. In addition to these surveys, students completed long-form blog entries which provided further qualitative evaluation of their experiences with the system.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Group 2 was composed of students recruited from a Spring 2014 introductory-level Information and Computer Science course. This group was subdivided into two subgroups: Subgroup 2A, which completed tasks related to the installation tool, and Subgroup 2B, which completed tasks related to the Smart Grid Game Designer.

Group 2A completed a subset of the tasks completed by Group 1 which were related to the installation of Makahiki and its software dependencies. Each test subject created an identical Ubuntu Linux 12.04 LTS virtual machine using Vagrant according to the instructions in the survey in Appendix B.1. During the installation, subjects were asked if they had questions about the system. If the Vagrant installation left the virtual machine in an unusable state, the error was noted in the survey form comments, the installation task was marked as not completed, and the user was switched to a second fully-configured virtual machine to complete the remaining tasks.

Group 2B completed a subset of the tasks completed by Group 1 which were related to the configuration of a new sustainability game, using the Mozilla Firefox web browser to access the Makahiki web administration interface and its Smart Grid Game Designer. Each test subject used an identical copy of the same Ubuntu Linux 12.04 LTS virtual machine on VirtualBox. Each installation of Makahiki was accessed according to the instructions in the survey in Appendix B.2.

3.2.1. Testing Equipment

Subjects in Group 2A and 2B conducted their usability testing on a laptop with a quad-core Intel i5-3230M CPU running at 2.60 GHz, with 7.89 GB of RAM available. Vagrant, VirtualBox, and Git for Windows (required to download the Makahiki source code from GitHub) were pre-installed on a Windows 8 partition which was the sole partition on the laptop's hard disk. The laptop was connected to a second monitor. The activity on the laptop's main monitor

and the audio from the laptop's microphone were recorded by Camtasia Studio screencasting software. Subjects were verbally instructed to keep the command line and browser windows on this monitor to ensure that they were recorded by the screencast. Kushniruk and Patel recommend creating video recordings of user sessions in usability testing to create a complete record of user interactions with a system [10].

The second monitor, which was not recorded due to technical limitations of Camtasia Studio, was used to complete the survey forms and view the Makahiki documentation without obstructing the subject's view of the main screen. After completing each of their assigned tasks, each member of Group 2A or 2B completed a usability assessment survey with questions adapted from those used for Group 1, in addition to a new set of questions that assessed general familiarity with the software used for each task. Each survey question asked the test subject to record the time taken to complete each task, to describe any problems they experienced while using Makahiki, and to rate each task for its perceived difficulty on a scale from 1 to 5. The survey administered to Group 2A is reproduced in Appendix B, and the survey administered to Group 2B is reproduced in Appendix C.

Screencast recordings were saved to an external 1TB hard disk drive. Virtual machine shared folders for downloading the Makahiki source code were also located on this external drive. The external drive was used to keep the virtual machine files and screencast videos from taking up space on the laptop's hard disk.

4. Results

4.1. Prior Experience Assessments

The surveys for Groups 2A and 2B each included a section called the "Prior Experience Assessment." These sections consisted of a series of multiple-choice and Likert-scale questions

SIMPLIFYING SUSTAINABILITY GAME DESIGN

on a scale of one to five which were intended to gauge the familiarity of each subject with the software which was used for each task. Much of this data was not collected for the 2013 students in Group 1, so in many cases comparisons of prior experience between Group 1 and Group 2 are not possible.

When asked to self-assess their level of experience with computers on a scale of one to five, Group 2A reported a mean self-assessment of 3.86, with a mode rating of 4. Group 2B reported a mean self-assessment of 3.5, with a mode rating of 3. In both group 2A and 2B, no subject had ever used Makahiki. These results appear in Table 2 and Table 3.

Table 2. Self-Reporting of Computer Experience

	Group 2A: Vagrant Installation	Group 2B: SGG Configuration
Mean	3.9	3.5
Mode	4	3
Minimum	3	3
Maximum	4	4
Subjects were asked "How experienced do you consider yourself with computers?" Possible ratings range from 1 ("not experienced") to 5 ("very experienced").		

Table 3. Self-Reporting of Makahiki Experience

	Group 2A: Vagrant Installation		Group 2B: SGG Configuration	
Have you used the Makahiki software before participating in this study?	Count	%	Count	%
Yes	0	0%	0	0%
No	7	100%	6	100%

This indicates that the subjects in both groups consider themselves moderately competent with respect to the use of personal computers. It also indicates that they had no prior experience with the Makahiki serious game design framework, a quality which is beneficial in usability testing that is meant to simulate the experience of new users.

Subjects in Group 1 described the operating system they installed Makahiki on as part of their required blog posts for the assignment. The majority of subjects used a native, dual-boot, or

SIMPLIFYING SUSTAINABILITY GAME DESIGN

virtual machine installation of Ubuntu Linux or another Linux distribution to run Makahiki. This information appears in Table 4, below.

Table 4. Group 1: Operating Systems Used for Makahiki Manual Installation

Operating System	Count	Percent of total
Ubuntu Linux 12.04, 12.04.1, or 12.10	5	62.5%
Debian Linux 7	1	12.5%
Linux Mint 14	1	12.5%
Mac OS X	1	12.5%

Subjects in Group 2A were required to access a Linux virtual machine running on a Microsoft Windows host operating system and to use the Windows command line with integrated Git commands for some tasks. Subjects in Group 2B configured Makahiki in the cross-platform Mozilla Firefox web browser and had little interaction with Windows-specific features. The familiarity of subjects with major families of operating systems appears in Table 5.

Table 5. Group 2: Operating System the Subject Was Most Experienced With

	Group 2A: Vagrant Installation		Group 2B: SGG Configuration	
Option	Count	Percent of total	Count	Percent of total
Microsoft Windows	6	85.7%	5	83.3%
Mac OS X	1	14.3%	1	16.7%
Linux	0	0%	0	0%
BSD	0	0%	0	0%
Other	0	0%	0	0%

These results indicate that the use of Microsoft Windows as the host operating system, in and of itself, is not likely to have created difficulties for the majority of test subjects in either group.

Group 2A was polled to determine their experience with command line or terminal shell applications similar to the Windows command prompt that was used to execute the installation tool, as well as with the Git and Vagrant applications which were used in other tasks. The results from these questions appear in Table 6.

Table 6. Group 2A: Vagrant Installation: Self-Reporting of Application Experience

Have you used [name of application] before participating in this study?	A Command Prompt, Terminal, or other command-line shell application at least once		Git		Vagrant	
Option	Count	Percent of total	Count	Percent of total	Count	Percent of total
Yes	6	85.7%	1	14.3%	0	0%
No	1	14.3%	6	85.7%	7	100%

This indicates that a majority of users have used a command prompt or similar shell application before, though using an application "at least once" does not guarantee familiarity. However, it indicates that a majority of subjects have never used Git or Vagrant before.

Group 2B was polled to determine their exposure to different web browsers, since a web browser is the main means of interaction with the Smart Grid Game Designer. The results of this question appear below in Table 7.

Table 7. Group 2B: SGG Configuration: Self-Reporting of Browser Familiarity

Which desktop browser are you most familiar with?	Count	Percent of total
Microsoft Internet Explorer	1	16.67%
Mozilla Firefox	1	16.67%
Google Chrome	4	66.67%
Safari	0	0%
Opera	0	0%
Other	0	0%

The majority of users are most familiar with Google Chrome. Since the ability to administer the software in a desktop environment is the primary consideration for usability in this test, mobile browsers were not included as options.

4.2. Comparison of Manual and Vagrant Installation Methods

The results from some tasks in Group 1 ("Unix Manual Installation") were directly comparable to Group 2A when the tasks were aggregated into categories based on what overall purpose they served.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

"Download the Software" represents the task of downloading the Makahiki repository from GitHub. "Setup runtime environment, install dependencies, install the software" consists of installing a Python compiler, C compiler, Git, virtualenv, the Python Imaging Library, the PostgreSQL backend database, the Memcache caching library, the Django web framework and web server, and other packages installed via the Pip package management software. It also includes the editing of the necessary configuration files for each application and library, and the initialization of the Makahiki database. Group 1 accomplished these tasks in many separate steps, while Group 2A accomplished this task in a single step by running the installation tool. For Group 1, "Start the server" consists of issuing commands in a command line terminal to start Makahiki's web server. For Group 2A, "start the server" consists of initiating an SSH connection to the server, then changing to the correct directory to run the web server, then running commands in a command line terminal to start the server. The task for "Verify that Makahiki is running," was the same for Group 1 and 2A: view the Makahiki landing page in a web browser, then stop the web server in the terminal. Completion times for each category appear in Table 7.

Table 8. Comparison of Manual Installation and Vagrant Installation

	Group 1: Manual Installation (minutes)			Group 2A: Linux on Vagrant Installation (minutes)		
Task Category	Mean	Minimum	Maximum	Mean	Minimum	Maximum
Download the Software	6.00	5.00	15.00	5.17	1.62	13.93
Setup runtime environment, install dependencies, install the software*	105.00	77.00	170.00	18.80	16.63	21.87
Start the server	1.00	1.00	1.00	2.42	1.00	4.62
Verify that Makahiki is running	2.00	1.00	5.00	1.78	0.50	3.83
Total time**	84.00	54.00	120.00	34.13	25.85	45.55
*This category represented many separate steps in the manual installation, but the Vagrant installation combines all of these steps into a single step. **This is not a sum of any other cells in this table. Rather, it is the sum of the mean, minimum and maximum times recorded by individual subjects.						

SIMPLIFYING SUSTAINABILITY GAME DESIGN

In both Group 1 and 2A, the largest portion of time is used to set up the runtime environment, install dependencies, and install software. Group 2A has lower times in all categories except "start the server." A percentage comparison appears in Table 9, below.

Table 9. Vagrant Installation Time as a Percentage of Manual Installation Time

Task Category	Percentage			Percent difference		
	Mean	Minimum	Maximum	Mean	Minimum	Maximum
Download the Software	86.11%	32.33%	92.89%	-13.89%	-67.67%	-7.11%
Setup runtime environment, install dependencies, install the software	17.90%	21.60%	12.86%	-82.10%	-78.40%	-87.14%
Start the server	241.67%	100.0%	461.67%	+141.67%	0.00%	+361.67%
Verify that Makahiki is running	89.17%	50.00%	76.67%	-10.83%	-50.00%	-23.33%
Total time*	40.63%	47.87%	37.96%	-59.37%	-52.13%	-62.04%
*This is not a sum of any other cells in this table. Rather, it is based on the mean, minimum and maximum times recorded by individual subjects.						

The category which sees the largest time reductions in the Vagrant installation is "setup runtime environment, install dependencies, install the software," which achieves reductions of more than 75% relative to the manual installation. The "total time" category records smaller but related reductions of more than 50% relative to the manual installation. Results in other categories are less drastic. "Download the software" and "verify that Makahiki is running" experience reductions of less than 30% for mean and maximum times. "Start the server" has at best identical results to the manual installation, and in the worst recorded time takes 361.67% as much time; however, this works out to a gain of less than five minutes in Table 9, an increase which is cancelled out by the reductions in other categories.

Breaking down the Vagrant task times into times for individual survey tasks in Table 10 confirms that "Clone the Repository" and "Set up Makahiki in the Virtual Machine" have two of the three highest mean completion times out of all Vagrant-related tasks. The third-highest time

SIMPLIFYING SUSTAINABILITY GAME DESIGN

goes to "Initialize Makahiki," a step which was included in Group 1's calculations for "Setup runtime environment, install dependencies, install the software." For Group 2A, this is a manual repetition of a task that the Vagrant script does automatically when it runs.

Table 10. Completion Time for Individual Tasks in the Vagrant Installation (Minutes)

Task	Mean	Standard Deviation	Minimum	Maximum
2.1.1.1.1.5. Open a Command Line Application	0.72	0.36	0.27	1.08
2.1.1.1.1.7.2. Clone the Repository	4.45	3.94	1.35	12.85
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	18.79	1.95	16.63	21.87
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	0.59	0.26	0.33	1.12
2.1.1.1.1.2.3. Start the Server	1.82	0.88	0.67	3.50
2.1.1.1.1.2.4. Verify that Makahiki is Running	1.78	1.18	0.50	3.83
2.1.1.1.1.5.1. Initialize Makahiki	4.44	1.00	3.28	6.02
2.1.1.1.1.5.2. Update the Makahiki Instance	2.48	0.87	1.42	3.52

4.2.1. Difficulty Ratings for the Vagrant Installation

In the survey, subjects in Group 2A were given the opportunity to rate each task, and to rate the overall usability of the installation tool and its supporting documentation. With the exception of the maximum (i.e., highest difficulty rating assigned by a subject) rating of 4 for "Overall Documentation," each step in the usability test was rated as having low to medium difficulty on average. This is displayed in Table 11.

Table 11. Difficulty Ratings By Task

Task	Mean	Standard Deviation	Minimum	Maximum
2.1.1.1.1.5. Open a Command Line Application	1.43	0.53	1	2
2.1.1.1.1.7.2. Clone the Repository	2.29	0.76	1	3
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	2.00	0.71	1	3
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	1.17	0.41	1	2
2.1.1.1.1.2.3. Start the Server	1.29	0.49	1	2
2.1.1.1.1.2.4. Verify that Makahiki is Running	1.57	0.53	1	2
2.1.1.1.1.5.1. Initialize Makahiki	1.43	0.53	1	2

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.1.1.1.1.5.2. Update the Makahiki Instance	1.14	0.38	1	2
Overall Usability	2.14	0.38	2	3
Overall Documentation	2.29	1.25	1	4
Ratings are on a scale of 1 to 5, with 1 being "very easy" and 5 being "very difficult."				

In Table 11, "2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine," the step in which the installation tool is run, was rated an average of 2.00 on the 5.0 scale (lower scores being considered better in terms of usability) and had a maximum rating of 3, which indicates "medium" difficulty of use. This task and "2.1.1.1.1.7.2. Clone the Repository," with a mean rating of 2.29, have the highest mean difficulty ratings and the highest standard deviations of any of the non-"Overall" tasks, indicating that test subjects had widely varying assessments of its difficulty.

In Table 11, "Overall Usability" has a mean rating of 2.14 and "Overall Documentation" has a mean rating of 2.29. This suggests that the installation process and its documentation are not considered very difficult to use by most test subjects. Ratings of the documentation have the widest variation, with a standard deviation of 1.25 and ratings ranging from 1 ("very easy") to 4 ("difficult").

4.2.2. Comparison of Subject Comments for Manual and Vagrant Installations

Examining some of the subject comments from Group 1 and Group 2A, which are summarized in Table 12 and Table 13 respectively, provides possible explanations for why certain tasks received the ratings and completion times that they did, and indicates possible bugs which were encountered during usability testing.

Table 12. Summary of Subject Comments for Group 1 (Manual Installation)

Issue summary	Task number in Appendix 1A survey	Number of users reporting problem	Issue reported by Group 2A?
Users did not know how to edit the .bashrc or .profile files.	2.1.1.1.6, 2.1.1.1.8	1 of 9	No
The PostgreSQL configuration file is difficult to find.	2.1.1.1.8	3 of 9	No

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Users could not find libz: it is stored in the zlib.so file, not a file called "libz."	2.1.1.1.7	1 of 9	No
Users were not made aware that environment variables must be set for Makahiki's database to work correctly.	2.1.1.1.13	1 of 9	No
More of the installation process could be handled by scripts.	General comment	2 of 9	No

The installation tool, by automating the installation and configuration of Makahiki software, eliminated several issues for Group 2A which had been reported by Group 1 when they installed and configured Makahiki manually. However, it created new issues of its own. Comments were open-ended, so subjects could raise more than one issue per task.

Table 13. Summary of Subject Comments for Group 2A (Vagrant Installation Tool)

Task	Summarized issue	Users reporting issue	Mean difficulty rating by users
2.1.1.1.1.7.2. Clone the Repository	It was difficult for the user to identify the current working directory relative to the directory the repository is downloaded into.	1 of 7	3
	The user made input errors in the command line.	1 of 7	3
	A Windows "Blue Screen of Death" error occurred for unknown reasons, requiring a restart of the computer.	1 of 7	2
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Running the "vagrant up" provisioning script while on a wireless network fails to install Django, creating an unusable installation. (Note: After two consecutive subjects experienced this, later trials were conducted using a wired Internet connection, and the two subjects' times for this task were counted as invalid, since it did not complete.)	2 of 7	No rating, marked as incomplete
	It is difficult for users to identify their location in the file system relative to the "makahiki" directory.	1 of 7	3
2.1.1.1.1.2.4. Verify that Makahiki is Running	The documentation tells users to stop the server with "control-c," which means pressing the keys "Ctrl" and "c." Some users typed a literal "control-c" into the command prompt instead.	2 of 7	2
2.1.1.1.1.5.1.	The initialization script's prompt for "Y/n" is	1 of 7	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Initialize Makahiki	case-sensitive: "Y" is required.		
Overall Usability	"No easy-to-use interface."	1 of 7	2
	It is difficult for users with little prior command line experience to navigate between directories.	2 of 7	2
	The documentation provided clear step-by-step instructions.	3 of 7	2
Overall Documentation	Users are not familiar with the command line and require more detailed instructions.	2 of 7	3
	The documentation is easy to read and/or is helpful because it specifies commands and example output.	4 of 7	1.5

Just over half of the Group 2A subjects considered the documentation easy to use, while roughly a third of Group 2A subjects requested more documentation on basic use of the Windows command line. Almost all other reported issues are related to the use of the command line or bad command line input, with the exception of the wireless connection issue or the unexplained Blue Screen of Death error. The lack of an "easy-to-use" interface is also criticized.

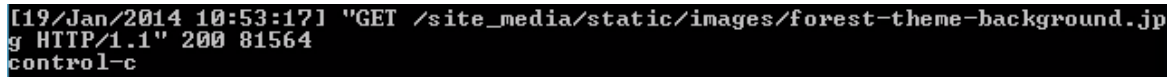
4.2.3. Analysis of Usability Issues in Vagrant Installation Screencast Data

The issues encountered during Vagrant installation which were not reported by subjects are related to misunderstandings of the documentation or a lack of information in the documentation. Examples of some of these errors can be seen in Figure 4 and Figure 5.

Table 14. Issues Identified in the Screencast

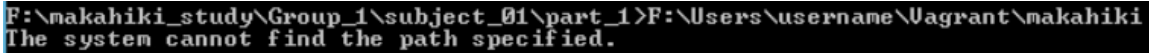
Issue description	Count of subjects experiencing issue	% of subjects experiencing issue
Subject cannot identify the location of the Makahiki directory without assistance from the researcher.	5 of 7	71.4%
Subject reads an example file path in the documentation, such as "<path-to-makahiki>," as a literal file path.	2 of 7	28.6%
Subject is not familiar with terminal commands such as "dir" (for Windows) or "cd" (for Unix / Linux).	2 of 7	28.6%
Subject attempts to stop the web server by typing "control-c" exactly as written in the documentation, rather than pressing the Ctrl (Control) key and "c" key that "control-c" represents.	3 of 7	42.9%
Subject sees the ">," "%," or "\$" that is used to represent	4 of 7	57.14%

a terminal prompt in the documentation's examples and uses it as part of the command.		
---	--	--



```
[19/Jan/2014 10:53:17] "GET /site_media/static/images/forest-theme-background.jpg HTTP/1.1" 200 81564
control-c
```

Figure 4. An example of the "control-c" error.



```
F:\makahiki_study\Group_1\subject_01\part_1>F:\Users\username\Vagrant\makahiki
The system cannot find the path specified.
```

Figure 5. An example of the "literal file path" error.

Example file paths in the Makahiki documentation assumed that Windows users were working on the default main disk partition, C:\. However, the design of this study placed the Makahiki source code and virtual machine shared folders on the F:\ external drive to avoid using up disk space on the C:\ drive. This is likely to have contributed to the problem of users being unable to identify the correct file path to the "makahiki" directory. In Figure 5, the example file path shown, "C:\Users\username\Vagrant\makahiki," will not work even if the "username" is replaced with the logged-on username, because the directory tree on the F:\ drive did not have a Users directory.

The errors which were caused by subjects making literal interpretations of generic command examples indicate the need for documentation to be as explicit as possible, even if the notation the documentation uses is considered normal to the author.

4.3. Comparison of the 2013 Web Administration Interface and the Smart Grid Game Designer

One caveat of comparing task completion time data between the 2013 Interface and the SGG Designer is that differences between the design of the studies made some direct comparisons impossible. The "2013 interface" testers in Group 1 were completing their survey tasks at home under no time limits, and were asked to modify existing actions into new actions (designed in a previous step) to add to a new level in the Smart Grid. By contrast, the SGG

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Designer survey asked subjects to reuse preexisting actions with researcher-specified modifications to maintain consistency. The results of the actions implemented in both the "2013 interface" and the SGG are displayed in Table 15.

Table 15. Comparison of the 2013 Interface and SGG Configuration Methods

Group 1 Task / Group 2B Task	2013 Interface (minutes)			SGG Configuration (minutes)		
	Mean	Minimum	Maximum	Mean	Minimum	Maximum
7.1 Create a Level / Create a Level*	11.00	5.00	15.00	7.80	3.70	14.97
7.2. Create a New Activity Action / Add a Fourth Action	16.00	5.00	30.00	2.42	1.03	5.45
7.3. Create a New Event Action / Add an Event Action	16.00	5.00	30.00	3.62	2.60	5.77
7.4. Create a New Commitment action / Add a Commitment Action	11.00	5.00	15.00	0.22	0.15	0.32
* In the SGG, this is the total time required to complete "2.3.6.3.4.3.2. Step 2: Design the "paths" that players take through the SGG," "2.3.6.3.4.3.4. Step 4: Design the layout: Create a level," and "2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names." In the 2013 interface, it was a single task.						

Table 16. SGG Designer Time as Percentage of 2013 Interface Time for Selected Tasks

Task	Percentage			Percent Difference		
	Mean	Minimum	Maximum	Mean	Minimum	Maximum
7.1. Create a Level / Create a Level	70.88%	74.00%	99.78%	-29.12%	-26.00%	-0.22%
7.2. Create a New Activity Action / Add a Fourth Action	15.13%	20.67%	18.17%	-84.87%	-79.33%	-81.83
7.4. Create a New Event Action / Add an Event Action	22.63%	52.00%	19.22%	-77.38	-48.00%	-80.78

As displayed in Table 16, the maximum time to "create a level" is almost equal in both interfaces and the mean time in the SGG is 74% that of the time in the "2013 interface." However, the minimum time to "create a level" in the SGG is approximately three minutes shorter than the minimum time in the "2013 interface."

SIMPLIFYING SUSTAINABILITY GAME DESIGN

In both interfaces, the Event Action task required users to set a date and location, define the task's level and category, and define an unlocking predicate. The mean and maximum completion times for this task in the SGG configuration are less than 25% that of its mean and maximum completion time in the "2013 interface," and the SGG minimum time is 52% of the "2013 interface" minimum time.

The SGG Designer task "Add a fourth action" required users to add a specific Activity action to the Smart Grid in Level 4 (created in a previous task) in a specific category and to specify its unlocking predicate using a specific action already placed in the grid. This was analogous to the task "7.2. Create a new Activity Action" in the "2013 interface," which required users to set an action's level, category, and unlocking predicate. Here, the SGG Designer has mean, minimum, and maximum completion times that are less than 25% of their equivalent times in the "2013 interface."

The result for the remaining comparable task, adding a Commitment action, is inconclusive because the task as implemented in this survey was significantly different from the task used in the 2013 survey. This is discussed further in Section 5.3, "Limitations."

Table 17. SGG Configuration Completion Time in Minutes

Task	Mean	Median	Minimum	Maximum
2.3.6.3.1. About the Smart Grid Game	2.13	2.24	0.87	3.35
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	2.40	2.25	1.72	3.22
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	1.52	1.47	1.15	1.85
2.3.6.3.4.3.2. Step 2: Design the "paths" that players take through the SGG	3.00	3.27	1.22	5.13
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	3.43	2.81	1.55	7.93
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	1.37	1.25	0.93	1.90
Add a premade Activity action	0.32	0.30	0.23	0.45
Add a premade Commitment Action	0.22	0.22	0.15	0.32

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Add a premade Event action	3.62	3.48	2.60	5.77
Add a fourth action	2.42	1.35	1.03	5.45
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	1.53	1.52	1.00	2.12
Total time*	22.24	23.05	14.20	28.80
Times are in minutes. *This is not a total of the values in each column. It is the mean, medium, minimum, or maximum total time value out of the total time values recorded for each subject.				

In Table 17, with the exception of "Open the Smart Grid Game Designer" and "About the Smart Grid Game," the tasks with mean completion times of two minutes or more are the tasks that involve the use of predicates. Makahiki predicates are a system of statements that are used by the 2013 Interface and the SGG Designer. They define how to unlock levels and activities based on "true" or "false" statements which describe which other levels and activities have been completed [14]. The task "Design the "paths" that players take through the SGG" required subjects to read documentation on Makahiki predicates. Creating a level, the task to "Add a premade Event action," and the task to "Add a fourth action" required users to input a Makahiki predicate to use as the condition for unlocking the level or action when given the "slug name" (name used by Makahiki) of the action that needed to be submitted before the level or action would unlock.

4.3.1. Difficulty Ratings for the SGG Designer

The most difficult tasks as rated by users were those which involved the predicate system: all of these tasks have a mean difficulty rating of 2.0 or higher in Table 18. Task 2.3.6.3.4.3.2 required subjects to read documentation on the predicate system and has a mean rating of 2.5. "Create a level," "Add a premade Event action," and "Add a fourth action" all involved specifying an unlock condition for a level or action using a Makahiki predicate.

Table 18. Difficulty Ratings By Task*

Task	Mean	Standard Deviation	Minimum	Maximum
2.3.6.3.1. About the Smart Grid Game	2.00	0.63	1	3
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	2.33	0.82	1	3
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	2.33	1.51	1	5
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	2.50	0.55	2	3
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	3.00	0.89	2	4
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	1.67	0.82	1	3
Add a premade Activity action	1.00	0	1	1
Add a premade Commitment action	1.00	0	1	1
Add a premade Event action	2.40	0.55	2	3
Add a fourth action	2.00	0.71	1	3
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	1.80	0.84	1	3
Overall Usability	2.67	0.52	2	3
Overall Documentation	2.83	0.41	2	3

4.3.2. Comparison of Usability Issues for the 2013 Web Administration Interface and the SGG Designer

The majority of issues reported by subjects for both the 2013 Interface and the SGG Designer interface were unrelated to the design of the user interface. Instead, subjects reported issues that were related to Makahiki's predicate system.

Table 19. Summary of Subject Comments for Group 1 (2013 Interface)

Issue summary	Task number in Appendix 1B survey	Number of users reporting issue	Problem reported by Group 2B?
It is difficult to understand how unlock condition predicates for actions relate to other actions without being able to test grids before making them active for players.	7.0	2 of 9	Yes (though the ability to use "draft" grids before going live has been added)
The "Supported Predicates" documentation is confusing or poorly explained.	Applies to 7.0, 7.1, 7.2, 7.3, 7.4, and 7.5	4 of 9	Yes
A feature should be added that	General comment	2 of 9	No (feature

SIMPLIFYING SUSTAINABILITY GAME DESIGN

allows the unlock conditions connecting Smart Grid Game actions to be visualized.			added as the Grid Consistency Checker)
Note: There were other issues reported in 2013 which were unrelated to the SGG.			

Table 20 summarizes the issues reported by subjects in Group 2B when testing the SGG Designer. Comments were open-ended, so subjects could raise more than one issue per task. The majority of issues reported by Group 2B subjects are related to their difficulties in using the predicate system. A third of Group 2B users had a positive opinion of the new drag-and-drop user interface introduced in the SGG Designer, but another third considered the interface too cluttered. 22.2% of Group 2B users requested more example images in the documentation.

Table 20. Summary of Subject Comments for Group 2B (SGG Designer)

Task	Summarized issue	Users reporting issue	Mean difficulty rating by those users
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Contrary to documentation, the user is not presented with a "First Draft" popup upon starting the SGG for the first time. The user must use the "New Draft" tool.	1 of 6	1
	The SGG template named "default" is not the default template.	1 of 6	2
	Server experienced errors when trying to access the SGG Designer.	1 of 6	5
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	The predicate system is confusing.	1 of 6	3
	The user did not understand the difference between submitted and approved actions in the predicate system.	1 of 6	2
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	The instructions do not explain why slug names are needed for submitted-action predicates.	1 of 6	3
	It is not clear that a predicate needs to go into the Unlock field when creating a new level.	1 of 6	3
	The predicate directions should be on the same page as the level creation instructions.	1 of 6	2
	"The picture for the unlock commands need to be bigger and closer to the instruction. so the users don't overlook [them]."	1 of 6	4
	A user can create two levels with the same name. Changes to these level will not save.	1 of 6	N/A (error affected

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Editing actions in a duplicate level or trying to delete the level will cause errors. Deleting the grid is the only way to remove the levels. The duplicate levels caused errors when adding an Event action and a fourth action.		multiple later tasks)
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	The column names were difficult to find because they look similar to the Library Actions.	1 of 6	3
	The documentation images do not specify where column names should go in the SGG.	1 of 6	2
Add a premade Activity action	It was hard to find specific activities due to the way in which they were sorted.	1 of 6	1
Add a premade Event action	It is easy to confuse the "Unlock condition" and "Unlock condition text" form fields when editing an Event action.	1 of 6	3
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	It was hard to find the GCC tools on the SGG page.	2 of 6	2
Overall Usability	The drag-and-drop GUI for actions and column names is easy to use.	3 of 6	2.67
	The documentation requires more relevant images, or more images.	2 of 6	2.5
	It is difficult to find the specific elements of the user interface that you need.	3 of 6	2.5
Overall Documentation	Some terms (levels, actions, slug names) are not clearly explained.	2 of 6	3
	The documentation needs more images.	2 of 6	3
	Predicate system documentation is unclear.	3 of 6	2.67

The level duplication issue encountered in "2.3.6.3.4.3.4. Step 4: Design the layout: Create a level" was previously unknown. Makahiki's level creation and management system was in place before the Smart Grid Game Designer was introduced, and it is not known whether or not this issue is a result of the introduction of the SGG Designer.

4.3.3. Analysis of Usability Issues in SGG Designer Screencast Data

After reading the predicate documentation, SGG test subjects still did not understand how predicates were structured or how slug names (the shortened names of SGG actions used by Makahiki to uniquely identify each action) were supposed to be used. Table 21 summarizes the types of mistakes which were made, and Figure 6 provides an example of one type of error.

Table 21. Issues Identified in the Screenscasts

Issue description	Count of subjects experiencing issue	Percentage of subjects experiencing issue
Subject uses the human-readable name of a Smart Grid Game action (e.g., "Hot Spots") in a Makahiki predicate instead of the slug name (e.g., "hot-spots").	3 of 6	50.0%
Subject uses the slug name by itself to set an unlock condition (e.g., using "hot-spots" instead of "submitted_action("hot-spots")."	2 of 6	33.3%
Subject types the "Unlock condition" Makahiki predicate into the "Unlock condition text" field instead.	1 of 6	16.7%
Subject uses predicate "approved_action("hot-spot") instead of required predicate "submitted_action("hot-spots")."	1 of 6	16.7%

Figure 6. A slug name being used without a predicate.

4.4. Task Time and Difficulty Rating Analysis

For the Vagrant installation tool's tasks as a whole, there was a moderate +0.573 positive correlation between the time spent to complete a task and the perceived difficulty that a user associated with that task. The Smart Grid Game tasks as a whole had a strong +0.906 positive correlation between completion time and perceived difficulty.

Table 22. Mean Task Time and Difficulty Correlation

Subject Group	Coefficient
Group 2A: Installing Makahiki on Vagrant	+0.573
Group 2B: Smart Grid Game	+0.906
This correlation was obtained by comparing the mean task times for each task to the mean rating for each task, for all tasks in a group's set of tasks.	

SIMPLIFYING SUSTAINABILITY GAME DESIGN

In Table 23, individual tasks from the Vagrant installation have low to moderately positive correlations between task completion time and perceived task difficulty. Only the correlations for "Open a Command Line Application" and "Connect to the Vagrant Virtual Machine" are higher than +0.500. Only "Initialize Makahiki" and "Update the Makahiki Instance" have negative correlations, though those correlations are below -0.500. The installation step, "Set up Makahiki in the Virtual Machine," has a weak positive correlation of +0.033. In general, for the Makahiki installation on Vagrant, higher task completion times are moderately associated with higher difficulty ratings, and vice versa.

Table 23. Vagrant Correlation between Task Completion Time and Difficulty Rating

Task	Coefficient
2.1.1.1.1.5. Open a Command Line Application	+0.861
2.1.1.1.1.7.2. Clone the Repository	+0.076
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	+0.033
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	+0.883
2.1.1.1.1.2.3. Start the Server	+0.204
2.1.1.1.1.2.4. Verify that Makahiki is Running	+0.397
2.1.1.1.1.5.1. Initialize Makahiki	-0.566
2.1.1.1.1.5.2. Update the Makahiki Instance	-0.538

In Table 24, the first task for SGG configuration, reading documentation in "About the Smart Grid Game," is strongly negatively correlated at -0.885. The task "Design the "paths" that players take through the SGG," which involves reading predicate documentation, is moderately negatively correlated at -0.234. This implies a moderate negative correlation: if a longer time is spent reading the documentation, it is likely that the perceived difficulty will also be lower, and vice versa. All other correlations are moderately positive with respect to task time and difficulty rating, implying that higher task times are associated with higher perceived difficulty scores.

Table 24. SGG Correlation between Task Completion Time and Difficulty Rating

Task	Coefficient
2.3.6.3.1. About the Smart Grid Game	-0.885
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	+0.229
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	+0.003
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	-0.234
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	+0.385
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	+0.485
Add a premade Activity action	None (all ratings were 1)
Add a premade Commitment Action	None (all ratings were 1)
Add a premade Event action	+0.397
Add a fourth action	+0.783
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	+0.084

In general, the Vagrant installation tool and SGG usability testing results indicate that longer times taken to complete a given task in either feature are associated with higher perceived difficulty ratings: a task which takes a long time in either one will likely also be perceived to be hard.

5. Discussion of Results

5.1. Vagrant Installation Usability Testing

The use of an automated Vagrant-and-VirtualBox-based solution for virtualizing a consistent cross-platform Linux environment and installing a consistent set of applications and software libraries resulted in significant decreases in installation and setup time in all categories except the time needed to start the web server. Test subjects in Group 2A rated the installation procedure as "very easy" or "easy to use," and the configuration issues reported by Group 1 in 2013 did not recur with Group 2A in 2014. This performance improvement is likely the result of the formerly human-handled configuration steps being incorporated into the automated Vagrant provisioning script.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

The problems experienced when subjects attempted to clone the Git repository into the correct directory or change their working directory to the Makahiki directory can be partly attributed to the experiment design. The survey used for Group 2A, in Appendix A.2., specified that a Windows Command Prompt would be opened for the subject by the researcher such that its working directory would be the directory that the Makahiki source code needed to be downloaded into. The Makahiki documentation, however, directs the user to change their working directory into the directory that they want to download Makahiki into, a directory that normally would have been chosen by a user but which needed to be a specific directory for the purposes of the experiment. The confusion of test subjects may also have been caused by the documentation being unclear about the fact that downloading the Makahiki source code created the "makahiki" directory in the directory where the "git clone" command was run. As a result, some test subjects searched for the directory without realizing that a single use of the command "cd makahiki" would have changed their working directory to the correct one. In addition, the test subjects' "makahiki" directories were on an external drive partition with the F:\ partition letter, but the Windows command prompt examples assumed that the user was working from the default C:\ partition. This part of the study design, in which the file system hierarchy conflicted with the general examples in the documentation, may have increased the difficulty that test subjects experienced in finding the correct directories.

5.2. SGG Designer Usability Testing

Test subjects in Group 2B considered its drag-and-drop interface easy to use, though a third of subjects commented that the interface's attempts to make many options available at the top level of the SGG designer made it cluttered and still made specific options difficult to find.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

The SGG documentation that was referenced by the survey was criticized for a lack of relevant pictures. However, the majority of issues raised by the test subjects were related to the predicate system, a feature that was unchanged from its state in 2013 when Group 1 tested it. Even though Group 2B subjects were required to read the predicate documentation, they found it difficult to understand. Subjects consistently used the slug name ("hot-spots") or the human-readable name ("Hot Spots") by itself. This may indicate that their instinct is to drop the enclosing predicate that specifies the condition that the action in the predicate needs to meet. This suggests that a more abstracted revision of the predicate system in which the task can be specified by its human-readable name and the predicate requirement (e.g., "submitted_action" or "approved_action") can be selected in some other way might match users' expectations more closely.

5.3. Limitations

Group 2A and 2B were limited by their smaller sample sizes. The ICS 691 course which supplied the test subject students for Group 1 usability testing had nine students, six of whom completed the Makahiki Local Installation Log survey and eight of whom completed the Makahiki Configuration and Management survey. Group 2A and 2B were intended to have ten subjects each in order to match this size. However, in Group 2A, one subject never scheduled an appointment. The first two subjects in Group 2A encountered a bug in the Makahiki installation procedure that made the installation unusable. The patch that was implemented changed the installation procedure, invalidating their results and leaving only seven subjects. Four Group 2B subjects cancelled their appointments, leaving only six test subjects.

Another limitation of these procedures is the comparability of their data sets. The surveys distributed to Group 1 allows for task lengths of an hour or longer, reflecting the greater time available for users who are performing individual testing outside a classroom setting. However,

SIMPLIFYING SUSTAINABILITY GAME DESIGN

the surveys distributed to Group 2 members were limited to the range of interactions with the system that could fit into an hour or less. The Makahiki installation in Vagrant fit the same tasks as its Group 1 manual installation into less than an hour due to the installation time decrease that it enabled. However, subjects in the SGG usability testing only tested the Smart Grid Game Designer's usability for a subset of the tasks carried out by Group 1. In particular, the "Add an Activity Action" and "Add a Commitment Action" tasks in the SGG, which add new actions to a specific level and category in the grid but do not change their unlock predicates, are not directly comparable to their "2013 interface" equivalents, "7.2. Create a New Activity Action" and "7.4. Create a New Commitment Action."

Limited generalizability of results is also a potential problem of this study. Though the survey results and the survey design will be generally applicable to research in the field of user interface design, Makahiki's status as a software framework for creating sustainability games marks it as a product targeted at a highly specific market that, at least initially, will be small.

5.4. Objectives for Future Research

A future method for collecting usability metrics such as time spent on specific tasks without needing to observe the subject in the lab might be to set up an instance of the Makahiki version to be tested on a "cloud" application platform such as Heroku. On this cloud instance, website metrics collection software could be installed to track user interactions with the web interface, and ask subjects to complete a list of tasks on their own time within a set number of days using only the Makahiki documentation and external sites that it links to, and report their results. In this scenario, the researcher would still be available to be contacted by email if assistance was necessary, as in the 2013 study. The use of a cloud service to host the virtual machines would give the researchers the ability to make uniform virtual hardware and software available to all subjects, as with the Vagrant virtual machine testing. It would also remove the time restriction of 1 hour that was placed on the Group 2 testing by allowing subjects to work on their own time as the Group 1 students did.

A significant limitation of the SGG and Vagrant usability studies with Group 2 is a small sample size. The 2013 study which collected data on the usability of the manual installation procedure and the "2013 interface" was administered to a class of nine students. The SGG study and Vagrant study were intended for ten subjects each, but cancellations and the need to remove invalid results due to a critical bug which was discovered during the first two Vagrant trials reduced the eventual total to six students for the SGG study and seven students for the Vagrant study. Though the usability testing produced useful insights into the effectiveness of the features and documentation added in response to feedback from the 2013 study, being able to conduct Makahiki usability testing with a much larger set of subjects in each group would reduce the need to correct for outliers.

6. Conclusion

For the Vagrant installation tool, the automated redesign resulted in significant reductions in the time required to install instances of Makahiki for use by software developers, and in the removal of installation and configuration issues that were reported by subjects who used the legacy manual installation procedure. The Vagrant installation procedure was generally perceived as easy to use as well.

The Smart Grid Game Designer had more mixed results. In the configuration tasks most directly comparable between the Smart Grid Game and the legacy interface of the 2013 study, reductions in task time were low to moderate. Subjects' feedback was divided between criticism of the Designer's options-rich menus for making it difficult to find things, and praise for the ease of use of its drag-and-drop interface. Though the SGG designer did address some user interface concerns, lower-level issues related to the documentation of the predicate system continued to be reported by test subjects.

In general, the redesigning of the Makahiki installation method and the Smart Grid Game interface in response to user feedback from the 2013 study resulted in significant reductions in task time and in the creation of installation methods and user interfaces that were perceived as being easy to use.

Appendix A. Survey Instruments for Group 1 in Spring 2013

Both surveys in this appendix were administered as Google Forms documents to the students of an Information and Computer Science 691 (graduate-level) course in the Spring 2013 semester. Completion of each survey was required as part of the completion of an assignment. All surveys in Appendix A were designed by Collaborative Software Development Laboratory member Yongwen Xu.

Appendix A.1. Makahiki Local Installation Assessment

The Makahiki Local Installation Log survey was administered as a Google Forms document to ICS 691 in Spring 2013 students from March 12 through March 18, 2013. The assignment required users to install Makahiki and its software dependencies on their personal computers or on a virtual machine running on their personal computers, based on online documentation provided for recent versions of Ubuntu Linux, Mac OS X, or Microsoft Windows. The survey is reproduced below.

Makahiki Local Installation Log

Please follow the steps outlined in this form to install Makahiki locally (including Virtualbox Linux Guest) and log the time you spent for each step.

Please choose the closest value from the list that best matches the time you spent during the installation.

Thank you!

*Required

2.1.1.1.1.2. Install Python *

Complete the "Install Python" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installpython>), record the time you spent for this section only:

- 0 minute (come with the OS install)

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- 5 minutes
- 15 minutes
- 30+ minutes

Record any problem(s) you encountered when installing Python:

2.1.1.1.1.3. Install C Compiler *

Complete the "Install C Compiler" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installccompiler>), record the time you spent for this section only:

- 0 minute (come with the OS install)
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered when installing C compiler:

2.1.1.1.1.4. Install Git *

Complete the "Install Git" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installgit>), record the time you spent for this section only:

- 0 minute (already installed from previous assignments)
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered when installing Git:

2.1.1.1.1.5. Install Pip *

Complete the "Install Pip" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installpip>), record the time you spent for this section only:

- 0 minute (already installed from previous assignments)
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered when installing Pip:

2.1.1.1.1.6. Install Virtual Environment Wrapper *

Complete the "Install Virtual Environment Wrapper" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html>)

SIMPLIFYING SUSTAINABILITY GAME DESIGN

#installvirtualenvironmentwrapper), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record the problem you encountered when installing virtual environment wrapper:

2.1.1.1.1.7. Install Python Imaging Library *

Complete the "Install Python Imaging Library" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html>

#installpythonimaginglibrary), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem(s) you encountered when installing Python imaging library:

2.1.1.1.1.8. Install PostgreSQL *

Complete the "Install PostgreSQL" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installpostgresql>), record the time you spent for this section only:

- 0 minute (already installed from previous assignments)
- 30 minutes
- 1 hour
- 2 hours
- 3+ hours

Record any problem(s) you encountered when installing PostgreSQL:

2.1.1.1.1.9. Install Memcache *

Complete the "Install Memcache" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#installmemcache>), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem(s) you encountered when installing Memcache:

2.1.1.1.1.10. Download the Makahiki source *

Complete the "Download Makahiki source" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html>

#downloadthemakahikisource), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hours

Record the problem you encountered when download the Makahiki source:

2.1.1.1.1.11. Workon Makahiki *

Complete the "Workon Makahiki" section in Makahiki Local Installation Manual

(<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#workonmakahiki>), record the time you spent for this section only:

- 0 minute (Already in the Makahiki virtual environment)
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered when activating Makahiki virtual environment:

2.1.1.1.1.12. Install required packages *

Complete the "Install required packages" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html>

#installrequiredpackages), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2 hours
- 3+ hours

Record any problem(s) you encountered when Installing required packages:

2.1.1.1.1.13. Setup environment variables *

Complete the "Setup environment variables" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html>

#setupenvironmentvariables), record the time you spent for this section only:

- 2 minutes
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record the problem you encountered when setting up environment variables:

2.1.1.1.1.14. Initialize Makahiki *

Complete the "Initialize Makahiki" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#initializemakahiki>), record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hours
- 2 hours
- 3+ hours

Record any problem(s) you encountered when initializing Makahiki:

2.1.1.1.1.15. Start the server *

Complete the "Start the server" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#starttheserver>), record the time you spent for this section only:

- 1 minutes
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when starting the server:

2.1.1.1.1.16. Verify that Makahiki is running *

Complete the "Verify that Makahiki is running" section in Makahiki Local Installation Manual (<http://makahiki.readthedocs.org/en/latest/installationmakahikiunix.html#verifythatmakahikiisrunning>), record the time you spent for this section only:

- 1 minutes
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem you encountered when verifying that Makahiki is running:

Your UH email: *

Appendix A.2. Makahiki Configuration and Management Assessment

The Makahiki Configuration and Management Log survey was administered as a Google Forms document to ICS 691 in Spring 2013 students from March 19 through March 24, 2013. The assignment required users to configure a sustainability challenge in Makahiki consisting of several smaller configuration tasks, based on online documentation. References to Heroku are related to a cloud application platform on which students' individual copies of Makahiki were deployed. Note that data from some of the survey questions are outside the scope of this thesis, as the procedures for those steps were not changed by the introduction of the Smart Grid Game Designer that was tested with Group 2B in 2014. The survey is reproduced below.

Makahiki Configuration and Management Log

Please follow the steps outlined in this form to configure and manage Makahiki, and log the time you spent and problems encountered for each step. Record the time you actually spent doing the tasks by choosing the closest value from the list that best matches the time you spent.

The Makahiki manual referenced below may use the local instance 127.0.0.1 as the example. For this assignment, you should use the Makahiki instance you deployed in Heroku instead of your local instance.

Thank you !

* Required

0. Update your Heroku Makahiki instance *

Read the "Updating your Makahiki instance" section in Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/installation-makahiki-heroku.html#updating-your-makahiki-instance>). Follow the instructions to update your Heroku instance with any changes from the Makahiki Git repository. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hour

Record any problem(s) you encountered in this step:

1. Getting to the challenge design page *

Read the "Getting to the challenge design page" section in Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design.html#getting-to-the-challenge->

SIMPLIFYING SUSTAINABILITY GAME DESIGN

design-page). Then go to the challenge design setting page of your Heroku instance. Record the time you spent for this step only:

- 1 minutes
- 5 minutes
- 15 minutes
- 1+ hours

Record any problem(s) you encountered in this step:

2. Design the global settings *

Read the "Design the global settings" section in Makahiki Manual

(<http://makahiki.readthedocs.org/en/latest/challenge-design-name-settings.html>). In your Heroku instance, change the "Name" of the challenge and the "Logo" fields to ones of your choosing.

Test that your change is in effect by checking the Logo image and label at the top of any page.

Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hours

Record any problem you encountered in this step:

3. Design the teams *

Read the "Design the teams" section in Makahiki Manual

(<http://makahiki.readthedocs.org/en/latest/challenge-design-teams-settings.html>). In your Heroku instance, add a new team called "Lehua-C" with the same group membership as the other teams in the default instance. Record the time you spent for this step only:

- 2 minutes
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hours

Record any problem you encountered in this step:

4. Set up users *

Read the "Set up users" section in Makahiki Manual

(<http://makahiki.readthedocs.org/en/latest/challenge-design-players-settings.html>). Add two new users of your choosing to the team "Lehua-C". Make sure you assign the players to their team by going to the user's profile link. Test your changes by logging in as one of the new players, and verifying that the player is on the right team. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

5. Specify the games to appear in your challenge *

Read the "Specify the games to appear in your challenge" section in Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-enable-disable.html>). Disable the "Water Game", and leave the other games enabled. You should see that the "Drop Down" page disappears from the top navigation bar. Record the time you spent for this step only:

- 2 minutes
- 5 minutes
- 15 minutes
- 30 minutes
- 1+ hours

Record any problem you encountered in this step:

6. Learn about how to design the resource goal games *

Read the "Design the Resource Goal Games" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-resource-game.html>). Record any questions or confusion that arises from reading this section:

6.1. Configure the Energy Goal Game for your new team *

Change the energy goal setting for the team "Lehua-C" to use manual data, and specify a time for the manual data input time. Test your changes by logging in as a player of Lehua-C, then go to "Go Low" page. You should see the calendar view of the daily energy goal game instead of the stop light visualization. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7. Learn about how to design Smart Grid Games *

Read the "Design the Smart Grid Game" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html>). Record any questions or confusion that arises from reading this section:

7.0. Design on paper *

The default installation defines a Smart Grid Game (SGG) with 3 levels. For this task, design a new Level 4 that extends the existing SGG. Level 4 will have a total of four actions: 3 new actions (Activity, Event, Commitment) that you create yourself, and one old action that you choose from the existing library of actions in the default installation. Design Level 4 with a 2x2 grid layout, including 2 categories of your choice. For this step, you will only design your Level 4 on a piece of paper or a spreadsheet, as described in Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html#designing-your-smart-grid-game>). Specify the unlock conditions for each action to achieve some kind of unlocking sequence("path"), such as depending on the completion of other actions. Record the time you spent in this step:

- 5 minutes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7.1. Create a Level *

Add a new level "Level 4", with priority higher than Level 3, and some unlock condition depending on some actions from Level 2. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7.2 Create a new Activity action *

Create a new activity action with your own content. Make the content meaningful. Fill in the required fields. You will also specify the level (should be level 4), category (your choice), as well as the unlock condition field, which determines the action "path" of your SGG design as described in step 7.0. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7.3 Create a new Event action *

Create a new event action with your own content. Make the content meaningful. Fill in the required fields. You will also specify the level field (should be level 4), category field (your choice), as well as the unlock condition field, which determines the action "path" of your SGG design as described in step 7.0. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7.4 Create a new Commitment action *

Create a commitment action with your own content. Make the content meaningful. Fill in only the required fields. You will also specify the level field (should be level 4), category field (your

SIMPLIFYING SUSTAINABILITY GAME DESIGN

choice), as well as the unlock condition field, which determines the action "path" of your SGG design as described in step 7.0. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

7.5 Finalize the grid *

At this point, you should have created 3 new actions and put them in Level 4 of your SGG. For this step, find the final action to complete your 2x2 grid.. Go to the admin interface, find an action in the action library, and modify the level, category and unlock condition field according to your SGG design. Play-test your grid by logging in as normal player, go to the "Get Nutz" page, unlock Level 4 and all actions in Level 4. Record the time you spent for this step only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2 hours
- 3+ hours

Record any problem you encountered in this step:

8. Design the Top Score Game *

Read the "Design the Top Score Game" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-topscore-game.html>), create a new topscore prize of your choice. Test your changes by going to the "Prizes" page to see your newly created prize. Record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

9. Design the Raffle Game *

Read the "Design the Raffle Game" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-raffle-game.html>). Create a new raffle prize of your choice. Test your changes by going to the "Prizes" page to see your newly created raffle prize and you can add raffle ticket to it. Record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- 2+ hours

Record any problem you encountered in this step:

10. Design the Badge Game Mechanics *

Read the "Design the Badge Game Mechanics" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-badge.html>). Create a new badge with an award trigger type of "smartgrid". Specify some kind of awarding condition depending on the smartgrid operations. Verify that your badge shows up in the badge catalog page and you can be awarded the new badge by doing the specified smartgrid action. Record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record any problem you encountered in this step:

11. Manage Action submissions *

Read the "Manage Action submissions" section in the Makahiki Manual (<http://makahiki.readthedocs.org/en/latest/execution-manage-smartgrid-game.html#manage-action-submissions>). Approve some actions submitted by you during your playtesting. Record the time you spent for this section only:

- 5 minutes
- 15 minutes
- 30 minutes
- 1 hour
- 2+ hours

Record how many actions you approved, and record any problem you encountered in this step:

Your UH email: *

Appendix B: Survey Instruments for Group 2 in Spring 2014

Both surveys in this appendix were administered as Google Forms documents to students who were recruited from an ICS 111 section in the Spring 2014 semester. All surveys in Appendix B were designed by Jordan Takayama.

Appendix B.1. Vagrant Installation Tool Assessment

The Makahiki Vagrant Installation Log was completed by test subjects randomly assigned to Group 2A. Students completed the installation, provisioning, and initial configuration of Makahiki on a Vagrant virtual machine. The Makahiki Vagrant Installation Log survey is reproduced below.

Makahiki Vagrant Installation Log

Follow the instructions on this survey form to install and configure the Vagrant software on Makahiki.

Complete the questions for each section while you follow the instructions.

Using the provided stopwatch, measure and record the time it took for you to complete each task.

If you experience problems or cannot understand the online instructions that this form links to, ask the researcher for help.

Make sure to record your problems in the "Record any problems you encountered in each step" text field.

If a task cannot be completed even with assistance from the researcher, state in the "Record any problems you encountered in this step" field that you were unable to complete it, and continue to the next task. If this made you unable to complete a later task, state in that task's "Record any problems you encountered in this step" field that you were unable to complete that task, and continue to the next task.

Thank you for your participation.

* Required

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Disclaimer

Prior to participating in this study, you agreed to a consent form. The consent form specified conditions which included the following:

The computer screen activity is being recorded as a screencast. This includes audio from the microphone.

Your name will be associated with your survey responses and your screencast video for purposes of data collection.

No personal information will be published.

You may stop participation in this study at any time during this appointment. If you do so, your results will be cancelled.

Completing and submitting this survey form implies your consent to the use of your data for research.

Part 0. Prior Experience Assessment

The following questions assess your familiarity with applications related to the Makahiki software.

1. Computer Experience *

How experienced do you consider yourself with computers? Mark only one oval.

- Not experienced 1 2 3 4 5 Very experienced

2. Command Line Experience *

Have you used a Command Prompt, Terminal, or other command-line shell application at least once before participating in this study? Mark only one oval.

- Yes
- No

3. Operating System Experience *

Which type of operating system are you most familiar with? Mark only one oval.

- Microsoft Windows
- Mac OS X
- Linux (e.g., Ubuntu, Debian, CentOS, Red Hat, Fedora, Linux Mint)
- Other: _____

4. Git Experience *

Have you used the Git software before participating in this study? Mark only one oval.

- Yes
- No

5. Vagrant Experience *

Have you used the Vagrant software before participating in this study? Mark only one oval.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- Yes
- No

6. Makahiki Experience *

Have you used the Makahiki software before participating in this study?

Mark only one oval.

- Yes
- No

Part 1. Installing Makahiki on Vagrant

This section installs the Makahiki software onto a Vagrant virtual machine by asking the user to follow instructions from the Makahiki documentation site.

7. 2.1.1.1.1.5. Open a Command Line Application *

A Windows Command Prompt has already been opened for you and pointed at the correct directory, so you will not need to open one. Read the documentation at <http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrant-environment-setup.html#open-a-command-line-application>. Read until the end of the "Windows" section, then stop. Did you complete this task? Mark only one oval.

- Yes
- No

8. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

9. Rate the difficulty of this task:

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

10. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

Important:

VirtualBox, Vagrant, and Git for Windows have already been installed on this computer. You do not need to follow the documentation to install them.

11. 2.1.1.1.1.7.2. Clone the Repository *

You will see a Windows Command Prompt terminal window that has had its directory changed to a "part_1" directory. Follow the instructions at <http://makahiki.readthedocs.org/en/latest/installationmakahiki-vagrant-environment-setup.html#clone-the-repository> to clone the Git repository into this directory. Did you complete this task? Mark only one oval.

- Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- No

12. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

13. Rate the difficulty of this task:

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

14. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

15. 2.1.1.1.2.1. Set up Makahiki in the Virtual Machine *

You are currently in the directory that contains "makahiki." Run "vagrant up" for the first time by following the instructions in <http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrantquickstart.html>. Count this task as completed once the installation finishes. Did you complete this task? Mark only one oval.

- Yes
- No

16. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

17. Rate the difficulty of this task:

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

18. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

Part 2. Starting the Makahiki Server

At this point, let the researcher know that you have completed Part 1. If the installation failed due to external factors, the researcher will switch you to another virtual machine.

19. 2.1.1.1.2.2. Connect to the Vagrant Virtual Machine *

In the terminal, connect to the Vagrant virtual machine with SSH. Do this by following the instructions from "2.1.1.1.2.2. Connect to the Vagrant Virtual Machine"

(<http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrant-quickstart.html#connect-to-the-vagrant-virtual-machine>). Leave your SSH session open. Did you complete this task? Mark only one oval.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- Yes
- No

20. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

21. Rate the difficulty of this task:

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

22. Record any other problems(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

23. 2.1.1.1.2.3. Start the Server *

Start the "runserver" server by following the instructions in "Start the Server"

(<http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrant-quickstart.html#start-theserver>). Do not start the gunicorn server. Did you complete this task? Mark only one oval.

- Yes
- No

24. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

25. Rate the difficulty of this task:

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

26. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

27. 2.1.1.1.2.4. Verify that Makahiki is Running *

Visit the landing page at 192.168.56.4:8000 using a new window of the Mozilla Firefox web browser on the main Windows computer. (If your installation failed and you are on the second virtual machine, use 192.168.56.5:8000 instead.) Move this window to the main monitor. When you have done this, stop the web server in the terminal window. To do this, follow the instructions in "Verify that Makahiki is Running" (<http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrantquickstart.html#verify-that-makahiki-is-running>). Did you complete this task? Mark only one oval.

- Yes
- No

28. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

29. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

30. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

31. 2.1.1.1.5.1. Initialize Makahiki *

Initialize the Makahiki database by following the instructions in "Initialize Makahiki" (<http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrant-runningmakahiki.html#initialize-makahiki>). Did you complete this task? Mark only one oval.

- Yes
- No

32. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

33. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

34. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

35. 2.1.1.1.5.2. Update the Makahiki Instance *

Update the Makahiki instance by following the instructions in "Update the Makahiki Instance" (<http://makahiki.readthedocs.org/en/latest/installation-makahiki-vagrant-runningmakahiki.html#update-the-makahiki-instance>). Did you complete this task? Mark only one oval.

- Yes
- No

36. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds)

37. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

38. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

Summary of Parts 1 and 2

These questions record your overall impressions of the process in Parts 1 and 2, which installed the Makahiki software on a Vagrant virtual machine.

39. Overall Usability Evaluation: Rating *

How easy or difficult was it to complete the tasks in Parts 1 and 2? Mark only one oval.

- Very easy to install 1 2 3 4 5 Very difficult to install

40. Overall Usability Evaluation: Long-Form Response *

In what ways was the installation easy or difficult?

41. Overall Documentation Evaluation: Rating *

How easy or difficult was it to understand the instructions that this survey linked to? Mark only one oval.

- Very easy to understand 1 2 3 4 5 Very difficult to understand

42. Overall Documentation Evaluation: Long-form Response *

In what ways was the documentation easy or difficult to understand?

When You Are Finished

When you are finished, submit this form. Let the researcher know that you are done. Leave the Command Prompt with the SSH session open. The researcher will stop the screen recording and close all open programs.

Appendix B.2. Makahiki SGG Designer Assessment

The Makahiki Vagrant Installation Log was completed by test subjects randomly assigned to Group 2B. Students edited a draft Smart Game Grid on an instance of Makahiki running on a preconfigured Vagrant virtual machine. The Makahiki SGG Configuration Log survey is reproduced below.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Makahiki SGG Configuration Log

Follow the instructions on this survey form to configure the Makahiki software. Complete the questions for each section while you follow the instructions. Using the provided stopwatch, measure and record the time it took for you to complete each task.

If you experience problems or cannot understand the online instructions that this form links to, ask the researcher for help.

Make sure to record your problems in the "Record any problems you encountered in each step" text field.

If a task cannot be completed even with assistance from the researcher, state in the "Record any problems you encountered in this step" field that you were unable to complete it. If this made you unable to complete a later task, state in that task's "Record any problems you encountered in this step" field that you were unable to complete that task.

Thank you for your participation.

* Required

Disclaimer

Prior to participating in this study, you agreed to a consent form. The consent form specified conditions which included the following:

The computer screen activity is being recorded as a screencast. This includes audio from the microphone.

Your name will be associated with your survey responses and your screencast video for purposes of data collection.

No personal information will be published.

You may stop participation in this study at any time during this appointment. If you do so, your results will be cancelled.

Completing and submitting this survey form implies your consent to the use of your data for research.

Part 0. Prior Experience Assessment

The following questions assess your familiarity with applications related to the Makahiki software.

1. Computer Experience *

How experienced do you consider yourself with computers? Mark only one oval.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

- Not experienced 1 2 3 4 5 Very experienced

2. Operating System Experience *

Which type of operating system are you most familiar with? Mark only one oval.

- Microsoft Windows
- Mac OS X
- Linux (e.g., Ubuntu, Debian, CentOS, Red Hat, Fedora, Linux Mint)
- Other: _____

3. Web Browser Experience *

Which desktop web browser are you most familiar with? Mark only one oval.

- Mozilla Firefox
- Microsoft Internet Explorer
- Google Chrome
- Safari
- Opera
- Other: _____

4. Makahiki Experience *

Have you used the Makahiki software before participating in this study? Mark only one oval.

- Yes
- No

Part 1. Makahiki Configuration with the Smart Grid Game Designer

This section configures parts of the Makahiki application via its Smart Grid Game Designer.

The instructor will start the web server for you before you begin this section. Open the Mozilla Firefox web browser. Go to the URL 192.168.56.4:8000 to view the landing page. Log in with username "admin" and password "admin."

5. 2.3.6.3.1. About the Smart Grid Game *

Read the "About the Smart Grid Game" section in the Makahiki Manual

(<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgridgame.html#about-the-smart-grid-game>). Did you complete this task? Mark only one oval.

- Yes
- No

6. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

7. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

8. Record any questions or confusion that arise from reading this section:

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

9. 2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer *

Open the Smart Grid Game Designer. To do this, follow the instructions in <http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgridgame.html#implementing-your-smart-grid-game-with-the-designer>. Did you complete this task? Mark only one oval.

- Yes
- No

10. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

11. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

12. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

13. 2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid. *

You need to create a New Draft using template "Default" and Draft name "New Demo Game."

Do this by following the instructions in <http://makahiki.readthedocs.org/en/latest/challenge-design-gameadmin-smartgrid-game.html#implementing-your-smart-grid-game-with-the-designer>. Did you complete this task? Mark only one oval.

- Yes
- No

14. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

15. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

16. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

17. 2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG *

To complete the rest of the tasks, you will need to understand the Makahiki predicates system. Read the documentation on Supported Predicates logic in <http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html?id2> to learn about the predicate system. Did you complete this task? Mark only one oval.

- Yes
- No

18. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

19. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

20. Record any confusion or questions that arise from reading this section: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

21. 2.3.6.3.4.3.4. Step 4: Design the layout: Create a level *

Create a new level named "Level 4." Using Makahiki predicates, set its unlock condition so that it will unlock when action "Hot Spots" (slug name: "hot-spots") is submitted. Do this by referring to the predicate documentation (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html?id2>) and following the instructions in <http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html?id4>. Did you complete this task? Mark only one oval.

- Yes
- No

22. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

23. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

24. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

25. 2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names *

Add the column names "Anything Goes" and "Energy Wizardry" to Level 4. Do this by following the instructions in <http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgridgame.html#id4>. Did you complete this task? Mark only one oval.

- Yes
- No

26. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

27. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

28. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

29. Add a premade Activity action *

In "Library Actions," drag the Activities action "Take Survey" into the first empty box under the "Anything Goes" column. Did you complete this task? Mark only one oval.

- Yes
- No

30. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

31. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

32. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

33. Add a premade Commitment Action *

In "Library Actions," drag the Commitment (Cmnts.) action "Turn off vampires" into the first empty box under the "Energy Wizardry" column. Did you complete this task? Mark only one oval.

- Yes
- No

34. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

35. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

36. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

37. Add a premade Event action *

First, in "Library Actions," drag the Event action "Your Future" into the second empty box under the "Anything Goes" column. Next, in the popup window, set the "Event Date" to one day after the current date. Next, use the sliders in the popup window to set the time to 05:30. Click "Done." Set the "Location" to "Hale Aloha Courtyard." Click "Set Date" to save changes. After this, click on "Your Future" in the Smart Grid. In the new page, change its Unlock Condition so that it will unlock after submitting action "Take Survey" (slug name: "take-survey"), reviewing the predicate documentation (<http://makahiki.readthedocs.org/en/latest/challenge-design-game-admin-smartgrid-game.html#id2>) if needed. Finally, click the "Save" button. Did you complete this task? Mark only one oval.

- Yes
- No

38. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

39. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

40. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

41. Add a fourth action *

In "Library Actions," drag the Activities action "OTEC video" into the second empty box under the "Energy wizardry" column. After this, click on "OTEC video" in the Smart Grid. You will be redirected to a configuration page. Change its Unlock Condition predicate so that it will unlock after submitting action "Turn off vampires" (slug name: "turn-off-vampires"), referring to the predicate documentation (<http://makahiki.readthedocs.org/en/latest/challenge-design-game->

SIMPLIFYING SUSTAINABILITY GAME DESIGN

admin-smartgrid-game.html#id2) as needed. Click the "Save" button. Did you complete this task? Mark only one oval.

- Yes
- No

42. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

43. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

44. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

45. 2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools *

Check your Designer Grid by running the GCC (<http://makahiki.readthedocs.org/en/latest/challengedesign-game-admin-smartgrid-game.html?highlight=sgg#grid-consistency-checker-gcc-tools>). Ignore any errors related to publication dates or event dates. If you have any other errors, fix them. Did you complete this task? Mark only one oval.

- Yes
- No

46. If you completed this task:

Record the time required to complete this task.

Example: 4:03:32 (4 hours, 3 minutes, 32 seconds):

47. Rate the difficulty of this task.

Do this only if you were able to attempt the task. If technical issues forced you to stop before you reached this task, do not complete this question. Mark only one oval.

- Very easy 1 2 3 4 5 Very difficult

48. Record any problem(s) you encountered in this step: *

Do this whether or not you completed the task. If technical issues forced you to stop before you reached this task, note that here.

Summary

These questions record your overall impressions of the process in this study, which configured activities using the Smart Grid Game Designer.

49. Overall Usability Evaluation: Rating *

How easy or difficult was it to use the Smart Grid Game Designer? Mark only one oval.

- Very easy to use 1 2 3 4 5 Very difficult to use

50. Overall Usability Evaluation: Long-Form Response *

In what ways(s) was it difficult or easy?

51. Overall Documentation Evaluation: Rating *

How easy or difficult was it to understand the Smart Grid Game Designer documentation that this survey linked to? Mark only one oval.

- Very easy to understand 1 2 3 4 5 Very difficult to understand

52. Overall Documentation Evaluation: Long-Form Response *

In what ways(s) was it difficult or easy to understand?

When You Are Finished

When you are finished, submit this form. Let the researcher know that you are done. Leave the Command Prompt with the SSH session open. The researcher will stop the screen recording and close all open programs.

Appendix C. Selected Data from 2013 Usability Testing

This section includes raw data from the Google Forms surveys. Archived copies of subjects' public blog entries, from which additional issues and information was identified, are available from the Collaborative Software Development Laboratory upon request.

Appendix C.1. Makahiki Local Installation Log: Survey Data

For the survey from which this data was taken, see Appendix A.1. Note that tasks with a time of 0 due to the dependency already being installed or configured on the subject's personal computer were not counted towards the average for that task.

Appendix C.1.1. Task Time Data**Table C.1.1.1. Subject 1 Task Completion Times.**

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (come with the OS install)
2.1.1.1.1.3. Install C Compiler	0 (come with the OS install)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	15
2.1.1.1.1.7. Install Python Imaging Library	15
2.1.1.1.1.8. Install PostgreSQL	30
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.1.1.1.1.11. Workon Makahiki	15
2.1.1.1.1.12. Install required packages	5
2.1.1.1.1.13. Setup environment variables	5
2.1.1.1.1.14. Initialize Makahiki	15
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	112

Table C.1.1.2. Subject 2 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (already installed)
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	5
2.1.1.1.1.5. Install Pip	5
2.1.1.1.1.6. Install Virtual Environment Wrapper	5
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	30
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	5
2.1.1.1.1.12. Install required packages	10
2.1.1.1.1.13. Setup environment variables	5
2.1.1.1.1.14. Initialize Makahiki	5
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	87

Table C.1.1.3. Subject 3 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (already installed)
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	15
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	0 (already installed from previous assignments)
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	0
2.1.1.1.1.12. Install required packages	10
2.1.1.1.1.13. Setup environment variables	2

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.1.1.1.1.14. Initialize Makahiki	5
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	5
Total	53

Table C.1.1.4. Subject 4 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (already installed)
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	5
2.1.1.1.1.6. Install Virtual Environment Wrapper	5
2.1.1.1.1.7. Install Python Imaging Library	15
2.1.1.1.1.8. Install PostgreSQL	0 (already installed from previous assignments)
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	5
2.1.1.1.1.12. Install required packages	5
2.1.1.1.1.13. Setup environment variables	2
2.1.1.1.1.14. Initialize Makahiki	5
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	54

Table C.1.1.5. Subject 5 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	5
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	15
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	60
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	15
2.1.1.1.1.11. Workon Makahiki	0 (Already in the Makahiki virtual environment)
2.1.1.1.1.12. Install required packages	5
2.1.1.1.1.13. Setup environment variables	2
2.1.1.1.1.14. Initialize Makahiki	5
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

running	
Total	119

Table C.1.1.6. Subject 6 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (already installed)
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	5
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	30
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	0 (Already in the Makahiki virtual environment)
2.1.1.1.1.12. Install required packages	5
2.1.1.1.1.13. Setup environment variables	5
2.1.1.1.1.14. Initialize Makahiki	5
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	67

Table C.1.1.7. Subject 7 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	5
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	30
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	0 (already installed from previous assignments)
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	0 (Already in the Makahiki virtual environment)
2.1.1.1.1.12. Install required packages	10
2.1.1.1.1.13. Setup environment variables	5
2.1.1.1.1.14. Initialize Makahiki	15
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	82

Table C.1.1.8. Subject 8 Task Completion Times

Task Name	Time to complete task (minutes)
2.1.1.1.1.2. Install Python	0 (already installed)
2.1.1.1.1.3. Install C Compiler	0 (already installed)
2.1.1.1.1.4. Install Git	0 (already installed)
2.1.1.1.1.5. Install Pip	0 (already installed)
2.1.1.1.1.6. Install Virtual Environment Wrapper	5
2.1.1.1.1.7. Install Python Imaging Library	5
2.1.1.1.1.8. Install PostgreSQL	180
2.1.1.1.1.9. Install Memcache	5
2.1.1.1.1.10. Download the Makahiki source	5
2.1.1.1.1.11. Workon Makahiki	0 (Already in the Makahiki virtual environment)
2.1.1.1.1.12. Install required packages	120
2.1.1.1.1.13. Setup environment variables	2
2.1.1.1.1.14. Initialize Makahiki	180
2.1.1.1.1.15. Start the server	1
2.1.1.1.1.16. Verify that Makahiki is running	1
Total	504
Note: This subject, the only subject to attempt a native Mac OS X installation, experienced unique errors during installation (see the table for Subject 8 in Appendix C.1.2) and was excluded from the data set as an outlier.	

Appendix C.1.2. Task Comments

A [No comment] indicates that the subject did not leave a comment in the form.

Table C.1.2.1. Subject 1 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	[No comment]
2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	Initially 5 minutes, but I wasn't paying attention and missed the required .bashrc config. Came back from step 11 and fixed this. Your instructions are fine: I should have tested the workon command at this point and noticed that something was wrong.
2.1.1.1.1.7. Install Python Imaging Library	More like 10 minutes. I followed your apt-get instructions and made sure to create the symlinks in /usr/lib
2.1.1.1.1.8. Install PostgreSQL	More like 10 minutes, actually. postgresql was already installed, but I needed to edit

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	<p>/etc/postgresql/9.1/main/pg_hba.conf. Giving the full path to the file in the instructions might be helpful for those less familiar with Linux.</p> <p>Ran psql -U postgres as a normal user to test and it worked.</p>
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	My fault: I missed the configuration and test step of part 6. I fixed that, and then this worked fine.
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.1.14. Initialize Makahiki	Ran fine, but I spent some time exploring the database and tables created in psql.
2.1.1.1.1.15. Start the server	Both web servers worked for me.
2.1.1.1.1.16. Verify that Makahiki is running	[No comment]

Table C.1.2.2. Subject 2 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	There were no problems.
2.1.1.1.1.3. Install C Compiler	There were no problems.
2.1.1.1.1.4. Install Git	There were no problems.
2.1.1.1.1.5. Install Pip	I needed to install setuptools first. Total time was about 2 minutes.
2.1.1.1.1.6. Install Virtual Environment Wrapper	There were no problems.
2.1.1.1.1.7. Install Python Imaging Library	I needed to create the symbolic links after installation. Otherwise, there were no problems.
2.1.1.1.1.8. Install PostgreSQL	<p>After 26 minutes of trying to find a way to install the recommended version (9.1.3) without having to recompile from source code (unless that was what was intended), I skipped this step to install memcached, made a snapshot, and came back later.</p> <p>I installed the default 9.1.8 version instead. This took about 4 minutes to install and configure.</p>
2.1.1.1.1.9. Install Memcache	Actual time was about 2.5 minutes.
2.1.1.1.1.10. Download the Makahiki source	Actual time was about 2.5 minutes.
2.1.1.1.1.11. Workon Makahiki	I did have to initialize the virtual environment with mkvirtualenv makahiki, but I think that's implied by the instructions even though it is not explicitly stated.
2.1.1.1.1.12. Install required packages	This took about 8 minutes, 15 seconds.
2.1.1.1.1.13. Setup environment variables	There were no problems.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.1.1.1.1.14. Initialize Makahiki	Actual time was about 2 minutes.
2.1.1.1.1.15. Start the server	There were no problems.
2.1.1.1.1.16. Verify that Makahiki is running	There were no problems.

Table C.1.2.3. Subject 3 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	[No comment]
2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	[No comment]
2.1.1.1.1.7. Install Python Imaging Library	[No comment]
2.1.1.1.1.8. Install PostgreSQL	[No comment]
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.1.14. Initialize Makahiki	[No comment]
2.1.1.1.1.15. Start the server	[No comment]
2.1.1.1.1.16. Verify that Makahiki is running	forgot to do collectstatic. When I did, it didn't work, so I checked the documentation, and realized I was running it from the wrong directory.

Table C.1.2.4. Subject 4 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	[No comment]
2.1.1.1.1.3. Install C Compiler	I am using Ubuntu.
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	[No comment]
2.1.1.1.1.7. Install Python Imaging Library	[No comment]
2.1.1.1.1.8. Install PostgreSQL	[No comment]
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.1.14. Initialize Makahiki	[No comment]
2.1.1.1.1.15. Start the server	[No comment]
2.1.1.1.1.16. Verify that Makahiki is	[No comment]

SIMPLIFYING SUSTAINABILITY GAME DESIGN

running	
---------	--

Table C.1.2.5. Subject 5 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	Had installed Python 2.7.3 for earlier assignments
2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	[No comment]
2.1.1.1.1.7. Install Python Imaging Library	[No comment]
2.1.1.1.1.8. Install PostgreSQL	Had a different setup for postgresQL, so I changed mine to reflect the installation manual.
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.1.14. Initialize Makahiki	[No comment]
2.1.1.1.1.15. Start the server	[No comment]
2.1.1.1.1.16. Verify that Makahiki is running	[No comment]

Table C.1.2.6. Subject 6 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	[No comment]
2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	[No comment]
2.1.1.1.1.7. Install Python Imaging Library	[No comment]
2.1.1.1.1.8. Install PostgreSQL	Less than 30 minutes, but not zero. Edited pg_hba.conf to use trust authentication with role postgres.
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.1.14. Initialize Makahiki	[No comment]
2.1.1.1.1.15. Start the server	[No comment]
2.1.1.1.1.16. Verify that Makahiki is running	[No comment]

Table C.1.2.7. Subject 7 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	Site did not contain instruction, took a moment to check it before unzipping and finding read me file (More accustomed to Windows environment over a linux one).
2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	Had issues which required looking up basic linux structure (finding .bashrc and .profile, figuring which one needed the added configuration in the installation instructions), then fixing an issue "Python could not import the module virtualenvwrapper.hook_loader" which required some Google research and ensure it works like it should.
2.1.1.1.1.7. Install Python Imaging Library	little extra time confirming installation because of lack of clarity over filenames (what exactly was the files needed for libjpeg and zlib or libz?).
2.1.1.1.1.8. Install PostgreSQL	While already installed had to run through the extra configuration changes and double check the install to make sure it works. (around 15 minutes)
2.1.1.1.1.9. Install Memcache	[No comment]
2.1.1.1.1.10. Download the Makahiki source	Extra time consumed because Vbox space limit somehow reached (not related to the project issue) Note: would be nice to inform user to go to a desired directory to put the makahiki folder first.
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	[No comment]
2.1.1.1.1.13. Setup environment variables	Was confused about if the two variables were defined previously or how to make a login script before following the postactivate concept for simplicity.
2.1.1.1.1.14. Initialize Makahiki	Forgot to actually do the two export statements in the last step in addition to setting them up in postactivate for the environment.
2.1.1.1.1.15. Start the server	[No comment]
2.1.1.1.1.16. Verify that Makahiki is running	[No comment]

Table C.1.2.8. Subject 8 Survey Form Comments

Task Name	Comment
2.1.1.1.1.2. Install Python	[No comment]

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.1.1.1.1.3. Install C Compiler	[No comment]
2.1.1.1.1.4. Install Git	[No comment]
2.1.1.1.1.5. Install Pip	[No comment]
2.1.1.1.1.6. Install Virtual Environment Wrapper	[No comment]
2.1.1.1.1.7. Install Python Imaging Library	[No comment]
2.1.1.1.1.8. Install PostgreSQL	<p>I couldn't find the pg_hba.conf file, at first. The Finder's "Find" tool didn't find it, nor did my incorrect use of the terminal's find command. The Postgres docs didn't help, nor did googling to find where the "cluster data directory" might've been located by default. On terminal, it was tricky. The correct syntax includes the base directory in the command, and it has to be run as root:</p> <pre>\$ sudo find / -name pg_hba.conf</pre> <p>I still don't know where my root "cluster data directory" is, because my system found four copies of the config file. Anyway, it wasn't fun to hunt them down—I changed them all.</p> <p>When it still wouldn't work, I eventually realized I hadn't created the "postgres" user from the psql prompt. Finally, the Makahiki installer quit complaining about the DB, but still it would throw huge stack traces and fail.</p> <p>At this point, I realized the database issues were resolved. It was the dependencies that were giving me trouble.</p>
2.1.1.1.1.9. Install Memcache	already installed on my machine
2.1.1.1.1.10. Download the Makahiki source	[No comment]
2.1.1.1.1.11. Workon Makahiki	[No comment]
2.1.1.1.1.12. Install required packages	<p>It turned out that the database configuration problems were only part of the problem. I didn't think Pip had failed earlier. It was the dependencies that were giving me trouble, I found on the third or fourth installation retry. This bit from one of many stack traces was helpful:</p> <pre>building '_pylibmc' extension clang -fno-strict-aliasing -fno-common -dynamic -g -Os -pipe -fno-common -fno-strict-aliasing -fwrapv -mno-fused-madd -</pre>

	<p>DENABLE_DTRACE -DMACOSX -DNDEBUG -Wall -Wstrict-prototypes -Wshorten-64-to-32 -DNDEBUG -g -Os -Wall -Wstrict-prototypes -DENABLE_DTRACE -arch i386 -arch x86_64 -pipe -DUSE_ZLIB -</p> <p>I/System/Library/Frameworks/Python.framework/Versions/2.7/include/python2.7 -c _pylibmcmodule.c -o build/temp.macosx-10.8-intel-2.7/_pylibmcmodule.o -fno-strict-aliasing</p> <p>clang: warning: argument unused during compilation: '-mno-fused-madd'</p> <p>In file included from _pylibmcmodule.c:34:</p> <p>./_pylibmcmodule.h:42:10: fatal error: 'libmemcached/memcached.h' file not found</p> <p>#include <libmemcached/memcached.h></p> <p>^</p> <p>1 error generated.</p> <p>error: command 'clang' failed with exit status 1</p> <p>-----</p> <p>StackOverflow helped me install the pylibmc that was needed (http://stackoverflow.com/questions/9070218/error-when-trying-to-install-pylibmc-on-mac-osx-lion). That was all I finally needed. Makahiki then worked fine on my machine.</p>
2.1.1.1.13. Setup environment variables	[No comment]
2.1.1.1.14. Initialize Makahiki	This step is where I also had to resolve the database and Pip dependency issues.
2.1.1.1.15. Start the server	[No comment]
2.1.1.1.16. Verify that Makahiki is running	[No comment]
<p>Note: This subject, the only one to attempt a native Mac OS X installation, experienced unusual errors during installation and was excluded from the time data set as an outlier for statistical purposes.</p>	

Appendix C.2. Makahiki Configuration and Management Log: Survey Data

For the survey from which this data was taken, see Appendix A.2. Though data from the other tasks is available from the Collaborative Software Development Laboratory upon request, this section only includes data for those tasks that were related to the subset of functionality which the Smart Grid Game Designer was designed to replace.

Appendix C.2.1. Task Time Data

Table C.2.1.1. Subject 1 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	15
7.2. Create a new Activity action	15
7.3. Create a new Event action	30
7.4. Create a new Commitment action	15

Table C.2.1.2. Subject 2 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	5
7.2. Create a new Activity action	15
7.3. Create a new Event action	15
7.4. Create a new Commitment action	15

Table C.2.1.3. Subject 3 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	5
7.2. Create a new Activity action	5
7.3. Create a new Event action	5
7.4. Create a new Commitment action	5

Table C.2.1.4. Subject 4 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	15
7.2. Create a new Activity action	15
7.3. Create a new Event action	15
7.4. Create a new Commitment action	5

Table C.2.1.5. Subject 5 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	5
7.2. Create a new Activity action	15
7.3. Create a new Event action	5

7.4. Create a new Commitment action	5
-------------------------------------	---

Table C.2.1.6. Subject 6 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	15
7.2. Create a new Activity action	30
7.3. Create a new Event action	30
7.4. Create a new Commitment action	15

Table C.2.1.7. Subject 7 Task Completion Times

Task Name	Time to complete task (minutes)
7.1. Create a Level	15
7.2. Create a new Activity action	15
7.3. Create a new Event action	15
7.4. Create a new Commitment action	15

Appendix C.2.2. Survey Form Comments

A [No comment] indicates that the subject did not leave a comment in the form.

Table C.2.2.1. Subject 1 Survey Form Comments

Task Name	Comment
7.1. Create a Level	After looking through the list of Supported Predicates, and the screen shot showing an unlock condition of "completed_some_of_level(2,"Level 2") ", I was confused. The Supported predicates lists a parameter called 'user', which I didn't know how to fill in. Although, I assume the system fills it in before calling the actual method. You might want to explain this, or implement a separate list of predicates for smartgrid designers with the parameters they actually need to provide.
7.2. Create a new Activity action	[No comment]
7.3. Create a new Event action	[No comment]
7.4. Create a new Commitment action	[No comment]

Table C.2.2.2. Subject 2 Survey Form Comments

Task Name	Comment
7.1. Create a Level	[No comment]
7.2. Create a new Activity action	[No comment]
7.3. Create a new Event action	[No comment]
7.4. Create a new Commitment action	[No comment]

Table C.2.2.3. Subject 3 Survey Form Comments

Task Name	Comment
7.1. Create a Level	There were no problems.
7.2. Create a new Activity action	There were no problems. However, I did make some mistakes with the dependencies, in which the wrong actions were unlocked first.
7.3. Create a new Event action	There were no problems. However, I did make some mistakes with the dependencies, in which the wrong actions were unlocked first.
7.4. Create a new Commitment action	There were no problems. However, I did make some mistakes with the dependencies, in which the wrong actions were unlocked first.

Table C.2.2.4. Subject 4 Survey Form Comments

Task Name	Comment
7.1. Create a Level	[No comment]
7.2. Create a new Activity action	[No comment]
7.3. Create a new Event action	[No comment]
7.4. Create a new Commitment action	[No comment]

Table C.2.2.5. Subject 5 Survey Form Comments

Task Name	Comment
7.1. Create a Level	<p>Screenshot order for smartgrid game did not match actual screen (minor: the order of items was different). Supported Predicates was somewhat confusing and defined predicate to use was not clear as syntax does not match others,</p> <p>completed_some_of_level(user, some=1, level_name=None) though only 2 parameters are given, and no explanation of parameters</p>
7.2. Create a new Activity action	<p>Needed to revise plan from paper to try to make it sound more meaningful, rechecked that it didn't exist, and am not sure if it works in context.</p> <p>For the path, since this is the starting unlock, I wasn't sure if it needed something specific to point out the path. Though it is illogical to put a creative task as a necessary action to unlock others, it was to make a clear path.</p>
7.3. Create a new Event action	<p>had to open extra windows to consult formatting and get the slug necessary for the unlock conditions.</p> <p>Noticed when doing the finalize the grid: "Now that I'm looking through it how do you make the code to be entered for the events?" Didn't notice the code area, and had to explore it to find the generate code area.</p>

SIMPLIFYING SUSTAINABILITY GAME DESIGN

7.4. Create a new Commitment action	Wasn't sure if you wanted a straight path or a forked path.
-------------------------------------	---

Table C.2.2.6. Subject 6 Survey Form Comments

Task Name	Comment
7.1. Create a Level	The predicate documentation was a touch confusing since they don't require a user argument when entered here.
7.2. Create a new Activity action	[No comment]
7.3. Create a new Event action	I expected another action in the same category to be placed in the same category column as the existing action. I set my Activity to priority 1000 and then created my Event with a priority of 2000 so that the Event would come below the Activity in the same column. However, this created two columns, each with the same heading. I set the Event priority back to 1000 and the two were placed in the same column, but not in the order I wanted. I set the Activity to 900 and then everything was right: Activity followed by Event in a single column. I'm not sure what's going on here, but it was confusing. For a while there I thought maybe the column was determined by the priority, not the category, but this can't be right given my final settings.
7.4. Create a new Commitment action	[No comment]

Table C.2.2.7. Subject 7 Survey Form Comments

Task Name	Comment
7.1. Create a Level	[No comment]
7.2. Create a new Activity action	[No comment]
7.3. Create a new Event action	[No comment]
7.4. Create a new Commitment action	[No comment]

Appendix D. Selected Data from 2014 Usability Testing

Appendix D.1. Makahiki Vagrant Installation Log Data

Appendix D.1.1. Survey Data

The survey which provided this data can be found in Appendix B.1. Completion times in this section are written in the format hours:minutes:seconds. Students whose names appeared on a signup sheet in odd-numbered positions¹ through 19 were assigned to Group 1. Subjects were tested in this order: 11, 17, 13, 5, 15, 1, 3, 7, 9. Subjects 11 and 17 were removed from the data

SIMPLIFYING SUSTAINABILITY GAME DESIGN

set due to a change in the installation process caused by the patching of a bug that caused their installations to become unusable. Subject 19's appointment was cancelled by the subject.

Table D.1.1.1. Subject 13 Survey Form

Question or task name	Sub-question	Response
Computer Experience		3
Command Line Experience		No
Operating System Experience		Microsoft Windows
Git Experience		No
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:16
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	N/a
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:04:41
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	n/a
2.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	No
	Record the time it took you to complete this task.	0:17:19 (not counted due to installation failure)
	Rate the difficulty of this task.	No rating
	Record any problem(s) you encountered in this step.	i did not complete the task. It took 17 minutes and 19 seconds before failing to install properly. v
2.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:26
	Rate the difficult of this task.	No rating
	Record any problem(s) you encountered in this step.	n/a
2.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:10
	Rate the difficulty of this task.	1
	Record any problem(s) you	n/a

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	encountered in this step.	
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:50
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	got confused with control c
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:06:01
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	n/a
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:03
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	n/a
Overall Usability Evaluation	Rating	2
	Long-form response	It was easy when you know what to type and once you get more familiar with what you are doing. The instructions were unclear at times.
Overall Documentation Evaluation	Rating	4
	Long-form response	Some parts of the instructions were unclear. I had never types anythings on the cmd before so I did not understand some of the instructions.
Total Time		0:37:46 (not counted due to installation failure)
Note: This subject failed to complete 2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine because of the bug in which the Django software failed to install when "vagrant up" was run on a wireless connection. This subject's time for that task, and total time, were removed from the data set when the bug was fixed for later subjects.		

Table D.1.1.2. Subject 5 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Microsoft Windows
Git Experience		No
Vagrant Experience		No

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:00
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:11
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	Length of time to clone.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	No
	Record the time it took you to complete this task.	0:13:39 (not counted due to installation failure)
	Rate the difficulty of this task.	No rating
	Record any problem(s) you encountered in this step.	13:39 failed to install django
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:40
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:23
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:04
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Pressed Ctrl-C multiple times without acknowledgement of server stop
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:17
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.5.2. Update the	Did you complete this task?	Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Makahiki Instance	Record the time it took you to complete this task.	0:01:54
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	None.
Overall Usability Evaluation	Rating	2
	Long-form response	No easy-to-use interface.
Overall Documentation Evaluation	Rating	1
	Long-form response	Very clear, code boxes sample output made it easier to follow.
Total Time		0:26:08 (not counted due to installation failure)
Note: This subject failed to complete 2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine because of the bug in which the Django software failed to install when "vagrant up" was run on a wireless connection. This subject's time for that task, and total time, were removed from the data set when the bug was fixed for later subjects.		

Table D.1.1.3. Subject 15 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Microsoft Windows
Git Experience		No
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:05
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	just was a little confused at what to do. after clarification from prompter, now know what to do for each step (i.e., closely what specific lines of instruction prompt asks you to read/follow)
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:42
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	had to recall what command 'dir' was. used to figure out I was either already in the directory I needed to be in via the instructions or I had done

SIMPLIFYING SUSTAINABILITY GAME DESIGN

		the change directory and simply forgot. Some time added for the actual cloning. Keyboard on this laptop not very good.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:18:57
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	just trying to orient myself on where i was in directory; since i cant see the folders and am not familiar with the folder hierarchy beforehand, it takes me awhile to figure out where i am. Installation takes most of the time recorded.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:20
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no problems
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:28
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no issues
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:30
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	managing multiple screens little disorienting, but otherwise okay (easy)
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:04:00
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	case sensitive Y for prompting y/n to continue
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:30
	Rate the difficulty of this task.	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Record any problem(s) you encountered in this step.	no problems
Overall Usability Evaluation	Rating	2
	Long-form response	easy: step-by-step instructions on wiki page, highlighting what commands to type in (verbatim) in text-highlight.ish formatted box (whatever that is called). Showed visual cues (i.e., output to expect if it is running correctly) to tell me that what I have done or am doing is correct/ on the right path. difficult: orienting myself to where I was in the directory, initially. Afterwards, just following the instructions verbatim helped me along.
Overall Documentation Evaluation	Rating	2
	Long-form response	easy: as above, instructions listed verbatim, with highlighting of what commands to type. For the most part, reading the instructions got me to where I needed to be. I glanced over most paragraphs however, and just went straight for the highlighted steps (i.e., commands to input to cmd.exe) hard: numbering schema of steps seem too long for end-user. Maybe use simpler numbering schema.
Total Time		0:30:32

Table D.1.1.4. Subject 1 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Microsoft Windows
Git Experience		No
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.1.5. Open a	Did you complete this task?	Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Command Line Application	Record the time it took you to complete this task.	0:00:27
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no problem
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:05:04
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Input errors in command line
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:21:52
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	no problem
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:28
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no problem
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:53
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no problem
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:11
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	control-c was a vague command
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:04:25
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	no problems
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:31
	Rate the difficulty of this task.	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Record any problem(s) you encountered in this step.	no problems
Overall Usability Evaluation	Rating	3
	Long-form response	It was difficult knowing what directories I was supposed to access when dealing with the virtual machine. Some of the online documentation did not match up with this computer. Most of the commands were easy to understand however.
Overall Documentation Evaluation	Rating	4
	Long-form response	Some commands like "control-c" were too vague. Knowing what directory to access was sometimes a problem.
Total time		0:39:51

Table D.1.1.5. Subject 3 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Microsoft Windows
Git Experience		No
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:03
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	I did not encounter any problems in this step.
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:12:51
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	There was a blue screen of death problem that hindered the process a bit.
2.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:18:46
	Rate the difficulty of this task.	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Record any problem(s) you encountered in this step.	There were no problems encountered in this step.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:28
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no problems encountered in this step.
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:41
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no problems encountered in this task.
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:15
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no problems encountered in this task.
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:05:09
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no problems encountered in this step.
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:20
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no problems encountered in this task.
Overall Usability Evaluation	Rating	2
	Long-form response	The only reason why the installation was difficult at times was because I misinterpreted the instructions and because of the blue screen of death error.
Overall Documentation Evaluation	Rating	2
	Long-form response	The documentation was only slightly harder for me to understand because I am not super familiar with the command line and its functions.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Total Time		0:45:33
------------	--	---------

Table D.1.1.6. Subject 7 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Mac OS X
Git Experience		No
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:20
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No problems
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:20
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No problems
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:17:45
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No Problems
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:07
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No problems
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:30
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	no problems
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:05
	Rate the difficulty of this task.	1
	Record any problem(s) you	no problems

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	encountered in this step.	
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:04:55
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No Problem
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:40
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No Problems
Overall Usability Evaluation	Rating	2
	Long-form response	Good guiding thru the whole process
Overall Documentation Evaluation	Rating	1
	Long-form response	easy language, easy to understand because we could see the outputs and what to type in the cmd
Total Time		0:33:15

Table D.1.1.7. Subject 9 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Command Line Experience		Yes
Operating System Experience		Microsoft Windows
Git Experience		Yes
Vagrant Experience		No
Makahiki Experience		No
2.1.1.1.1.5. Open a Command Line Application	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:50
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.7.2. Clone the Repository	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:21
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to	0:16:38

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	complete this task.	
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:40
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Mistakenly went ahead and started the server.
2.1.1.1.1.2.3. Start the Server	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:40
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.2.4. Verify that Makahiki is Running	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:32
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.5.1. Initialize Makahiki	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:18
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
2.1.1.1.1.5.2. Update the Makahiki Instance	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:25
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	None.
Overall Usability Evaluation	Rating	2
	Long-form response	It was easy because the documentation outright said the steps necessary.
Overall Documentation Evaluation	Rating	2
	Long-form response	The formatting made it easy to read.
Total Time		0:25:51

Appendix D.1.2. Screencast Activity Logs

The screencasts on which these logs are based have been uploaded to Youtube as private videos. Access to these videos is available through the Collaborative Software Development Laboratory upon request.

Table D.1.2.1. Subject 13 Screencast Activity Log

Task	Estimated time marker	Notes
2.1.1.1.1.7.2. Clone the Repository	00:04:24 - 00:08:46	<p>The subject is confused over whether Git needs to be installed, and what directory the GitHub repository is meant to be installed into.</p> <p>At 00:06:25, the subject types "cd <path-to-makahiki>/makahiki" at the command line prompt, taking the example file path literally. The instructor corrects the subject and the repository is download successfully.</p>
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	00:10:16 - 00:14:50	<p>The subject types the following incorrect commands with incorrect directory paths:</p> <ol style="list-style-type: none"> 1. cd <path-to-makahiki>/makahiki 2. C:\Users\username\Vagrant\makahiki <p>The instructor tells the subject to just type "cd makahiki," which works. The subject then runs the "vagrant up" command.</p>
	00:15:00 - 00:17:07	<p>The subject is asked if there are any questions. The subject asks, "So what's this about, anyway?"</p> <p>The resulting conversation then occurs:</p> <p>Instructor: "Have you seen or heard about virtual machines before?"</p> <p>Subject: "Yeah."</p> <p>Instructor: "Okay. Vagrant is a way of starting and stopping virtual machines from the command line, and you access it through SSH."</p> <p>Subject: "What's SSH?"</p> <p>Instructor: "It's a way of accessing a machine from the command line by knowing its IP address...well, by an automated thing. When the machine is on your local machine, you just access it."</p> <p>00:15:58 (in response to some question):</p> <p>Instructor: "The Vagrant virtual machine is currently having some software installed in it that is needed</p>

SIMPLIFYING SUSTAINABILITY GAME DESIGN

		for the Makahiki application to run. The point is that you can easily set up Makahiki on a disposable virtual machine by running the previous steps and running this command. It's currently creating the virtual machine based on a "precise32" virtual machine, and you can create more than one virtual machine from the same base virtual machine."
	00:17:07 - 00:19:10	The subject asks irrelevant questions about the Information and Computer Science degree program, which are answered. The instructor states that the instructions require Makahiki to be downloaded from GitHub because "inside Makahiki are the files that we use. We have to cd into Makahiki because once we cd into Makahiki we are using it as a shared folder."
	00:26:27	Instructor: Do you have any other questions about this part? Subject: Do you actually write the script? Subject: You wrote all of this. Instructor: No, most of this is just output.
	00:27:04 - 00:27:59	By this point, the command line output from "vagrant up" indicates that the Django installation has failed. The subject is told that the instructor will switch to another virtual machine for the next task. The subject was instructed to mark the task as not completed.
N/A	00:32:44	The second Vagrant virtual machine finishes starting up.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:34:38	Subject runs "vagrant ssh" correctly, but forgets to start the timer.
	00:34:40 - 00:35:29	Subject runs "vagrant ssh" correctly, and times it.
2.1.1.1.1.2.3. Start the Server	00:35:37 - 00:37:53	Subject starts the Makahiki server correctly with "cd /vagrant/makahiki" and "./manage.py runserver 0.0.0.0:8000"
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:39:21 - 00:43:15	Subject verifies that the web server is running by reaching the landing page. However, the subject confuses "control-c" for a literal command, instead of Ctrl+C.
2.1.1.1.1.5.1. Initialize Makahiki	00:45:35 - 00:50:30	The subject mistakenly uses the command "./scripts/initialize_instance.py --py --type default," which has an extra "--py" flag, before running the correct command.
2.1.1.1.1.5.2. Update the Makahiki Instance	00:52:16 - 00:54:15	The subject begins the command with the example prompt (% sign) by mistake (i.e., "% cd

SIMPLIFYING SUSTAINABILITY GAME DESIGN

		/vagrant/makahiki") and is informed that this is not needed. The subject deletes the % and runs the command.
--	--	--

Table D.1.2.2. Subject 5 Screencast Activity Log

Task	Estimated time marker	Notes
2.1.1.1.1.7.2. Clone the Repository	00:07:37 - 00:08:50	While trying to "cd" into the makahiki directory, the subject mistypes the path to the "makahiki" directory as "F:/makahiki_study\Group 1\subject_05\part_1\makahiki" before realizing that it is missing an underscore and correcting it.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	00:08:56 - 00:11:00	<p>The subject runs "vagrant up."</p> <p>When asked if there are any questions, the subject asks "So what kind of software is this?"</p> <p>The instructor gives this answer: "It's an application that is creating web functions using Virtualbox and it uses the Virtualbox API to create these virtual machines. And using scripts - specially structured text files, you can set the network interfaces, the ports that are going to be used, and where the shared folder is. The shared folder stores all of the files you're using for Vagrant, which is why we had to go into for Makahiki first, which is the shared folder. Each virtual machine is based on a "box" which is a default virtual machine that you get from another site, and from one box you can make many virtual machines. After it's finished creating the virtual machine, it will run a provisioning script that installs all the packages - all the packages that the system will need for Makahiki to work."</p>
	00:11:00 - 00:22:00	The subject is warned that the installation sometimes fails due to a timeout error. When the installation finishes, it is found to have failed with ImportError for Django modules. The subject is switched to the second virtual machine for the remainder of the trial.
N/A	00:25:22	The instructor finishes booting up the second virtual machine.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:26:00 - 00:26:59	The subject runs the "vagrant ssh" command with no issues.
2.1.1.1.1.2.3. Start the Server	00:27:50 - 00:29:12	The subject starts the "runserver" server with no issues.
2.1.1.1.1.2.4. Verify that	00:30:35 -	The subject accesses the Makahiki landing page at

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Makahiki is Running	00:31:54	192.168.56.5:8000 and types Ctrl+C (the correct command) to stop the web server. The Ctrl+C keypress is displayed as a Unicode heart character ♥ in the command line terminal.
	00:32:06 - 00:32:35	The subject types "control-c," not realizing that the server has already been stopped by Ctrl+C. The researcher tells the subject that the task is complete.
2.1.1.1.1.5.1. Initialize Makahiki	00:34:00-00:35:45	The subject runs the script once, but forgets to start the timer. The script's run is aborted with Ctrl+C.
	00:37:36 - 00:40:27	The script runs for the second time, and is timed.
2.1.1.1.1.5.2. Update the Makahiki Instance	00:41:11 - 00:43:10	Subject successfully updates the installation.

Table D.1.2.3. Subject 15 Screencast Activity Log

Task	Estimated time marker	Notes	
2.1.1.1.1.1.7.2. Clone the Repository	00:04:05 - 00:06:47	The subject asks the instructor if the command line working directory is already in the correct directory, and correctly assumes that it is. The subject completes the "git clone" command correctly.	
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:08:40 - 00:10:54	The subject uses the "dir" command and several "cd" commands to attempt to find the "makahiki" directory:	
		Commands	Description
		> dir	1. Lists the current directory (output omitted).
		> cd makahiki	2. Switches the working directory into the "makahiki" repository.
		> cd makahiki/makahiki	3. Here the subject attempts to find a directory that does not exist.
		The system cannot find the path specified.	
		> dir	4. (output omitted)
		> clear	5. On a Linux system, "clear" would clear the terminal display of all text except for the current command prompt. On Windows, it does nothing.
		'clear' is not recognized as an internal or external command, operable program or batch file.	
		> dir	6. (output omitted)
		> vagrant up	7. The subject realizes that the current working directory is correct.
	00:11:00 -	While the Vagrant installation runs, the subject asks if	

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	00:12:30	there is a way to drag a Windows Explorer folder into the command line to put its file system path into the command line. The subject is told that that would only be possible if the subject knew exactly what directory needed to be used.
	00:13:30 - 00:27:50	<p>The subject asks about the relationship between Vagrant and VirtualBox, and is told that Vagrant connects to VirtualBox's API to manage virtual machines, and uses script files to install software packages.</p> <p>Approximately ten minutes of irrelevant discussion about the ICS program and the undergraduate project in response to subject questions occurs, ending at 00:27:50 when the installation script finishes.</p>
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:30:31 - 00:03:52	Subject executes the "vagrant ssh" command without any issues.
2.1.1.1.1.2.3. Start the Server	00:32:19 - 00:33:46	Subject starts the runserver server without any issues.
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:35:00 - 00:35:45	Subject visits the landing page and stops the web server without any issues.
2.1.1.1.1.5.1. Initialize Makahiki	00:36:40 - 00:40:44	The subject types "y" instead of "Y" at the initialize_instance.py script's Y/n prompt, causing an error message. The subject asks if it is case-sensitive, and corrects the error after being told that it is.
2.1.1.1.1.5.2. Update the Makahiki Instance	00:41:43 - 00:43:19	Subject updates the Makahiki instance without any issues.

Table D.1.2.4. Subject 1 Screencast Activity Log

Task	Estimated time marker	Notes
2.1.1.1.1.1.7.2. Clone the Repository	00:07:02 - 00:08:17	Subject mistypes the command without a space: "git clonehttp://GitHub.com/jtakayama/makahiki.git" The subject fixes the command on the second try and the repository is downloaded successfully.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	00:11:04	The subject asks the instructor for clarification on the location of the "makahiki" directory.
	00:12:50 - 00:13:02	The subject reads the example file path as a literal file path and tries to change directories into "F:\Users\username\Vagrant\makahiki"
	00:13:57 - 00:15:43	The subject says "I'm having trouble getting to this file." After further questioning, the instructor explains the "cd" (change directory) command.
	00:15:43 -	The subject executes the "vagrant up" command.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	00:32:35	When the instructor asks if the subject has any questions, the subject does not have any.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:35:00	The subject opens the ssh connection without any issues.
2.1.1.1.1.2.3. Start the Server	00:36:00 - 00:39:33	The subject starts the server without any issues.
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:40:32 - 00:40:57	The subject tries to stop the web server by typing "control-c" instead of "Ctrl+C."
2.1.1.1.1.5.1. Initialize Makahiki	00:44:00 - 00:47:25	The subject runs "initialize_instance.py --type default" without any issues.
2.1.1.1.1.5.2. Update the Makahiki Instance	00:49:10 - 00:51:41	The subject types the % example command prompt, mistaking it for part of the command: "% cd /vagrant"

Table D.1.2.5. Subject 3 Screencast Activity Log

Task	Estimated time marker	Notes
N/A	None	<p>Due to a blue screen of death crash, this session was recorded in 2 videos. Time markers for this video are prefaced with "Part 1" or "Part 2" to designate their position in each video.</p> <p>Part 1 was restored by Camtasia Studio after the crash, and contains graphical glitches. Some of the audio seems to have become de-synchronized from the video footage as well.</p>
N/A	Part 1 00:02:01	<p>The Command Line window turns purple and the mouse pointer turns dark blue.</p> <p>The user can be heard typing, but none of the changes are visible in the video.</p>
N/A	Part 1 00:04:20	<p>The command prompt returns to its normal color. The mouse pointer remains dark blue.</p>
2.1.1.1.1.7.2. Clone the Repository	Part 1 00:05:30	<p>The subject mistakenly ends their file path with a ">," mistaking the example command prompt for part of the command: "cd C:\Users\username\Vagrant\makahiki>"</p> <p>The subject needed to have their directory location explained, since they did not realize that they were on the F:\ drive, not C:\. The user then begins the "git clone" operation, which results in a "Permission denied" error:</p> <p>git clone http://GitHub.com/jtakayama/makahiki.git</p>

SIMPLIFYING SUSTAINABILITY GAME DESIGN

		Cloning into `makahiki`... F:/makahiki_study/Group_1/subject_03/part_1/makahiki/.git/: Permission denied
	Part 1 00:06:35	After the "Permission denied" message, USB insert and eject sounds are heard several times before the video cuts off. This is when the blue screen of death error took place.
	Part 2 00:02:10	The subject completes the "git checkout" operation.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	Part 2 00:04:03 - 00:05:57	The subject asks the instructor if typing in "cd and then this whole thing "slash makahiki"" is the correct command ; it is. The command is entered. The subject then starts the "vagrant up" command.
	Part 2 00:06:58	Subject: "I'm not very familiar with command-line stuff." Instructor: "Do you think the [unintelligible] instructions could be more specific?" Subject: "I think they could, but that's just for me."
	Part 2 00:22:08	The "vagrant up" script finishes.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	Part 2 00:24:17	The subject uses "vagrant ssh" without any issues.
2.1.1.1.1.2.3. Start the Server	Part 2 00:25:26 - 00:26:52:	The subject starts the web server without any issues.
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:28:00 - 00:28:23	The user reaches the Makahiki landing page and stops the web server with Ctrl+C.
2.1.1.1.1.5.1. Initialize Makahiki	00:30:00 - 00:34:58	The user mistypes the initialize_instance.py script: 1. "./scripts/ initiaize _instance.py--type default" ("initiaize" instead of "initialize") 2. ./scripts/initialize_instance.py--type default (missing space after ".py")
2.1.1.1.1.5.2. Update the Makahiki Instance	00:35:59 - 00:38:00	The subject expresses confusion over where the GitHub repository code needs to be downloaded into before updating the instance: Subject: "So I can't be in the makahiki folder when I download the source code?" Instructor: "In this case, no."
	00:38:18- 00:39:20	The Makahiki instance is updated successfully.

Table D.1.2.6. Subject 7 Screencast Activity Log

Task	Estimated time marker	Notes
2.1.1.1.1.7.2. Clone the Repository	00:05:00	The subject begins in F:\makahiki_study\Group_1\subject_07\part_1 and types the command "cd makahiki," but does not execute it. The subject then asks if the command line terminal already has the correct working directory. The instructor replies that it is, and that the current task is to clone the GitHub repository.
	00:06:32 - 00:07:27	The subject downloads the GitHub repository successfully.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	00:09:00 - 00:10:12	The subject is confused about the location of the "makahiki" directory: Subject: "So what do I need to do now? Just the last part?" Instructor: "You don't need to change directories because the makahiki directory is in this directory." Subject: "Just change to - what's the username? Makahiki?" Instructor: "Not the username. The directory is Makahiki. Also, write the time down here and we'll total it." (The subject had mistakenly stopped keeping time after finishing the documentation.)
	00:11:09 - 00:11:17	The subject enters "makahiki" to change to the Makahiki directory, without a "cd." This causes an error. The subject then correctly enters "cd makahiki."
	00:12:00	The "vagrant up" command begins to run. The subject is asked if there are any questions, and responds "no." The subject then asked if the operation was supposed to take this long, and was told that it is.
	00:27:27 - 00:27:43	The "vagrant up" command finishes without errors and the timer is stopped.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:30:07	The subject asks "I'm in the Makahiki directory, right?" The instructor replies "Yes."
	00:30:25 - 00:31:06	The "vagrant ssh" command finishes without errors.
2.1.1.1.1.2.3. Start the Server	00:32:27 - 00:35:08	Subject types: 1. "cd/vagrant/makahiki" (missing a space) 2. "cd /vagrant/makahiki" (There is an extra space after the /, but this is a valid path that goes to the root directory.) 3. "cd/vagrant/makahiki" (same mistake as 1.) 4. "cd /vagrant/makahiki" (correct)

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	00:35:31 - 00:36:00	Subject types <code>./manage.py runserver 0.0.0.0:8000,</code> which starts the web server correctly.
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:38:00 - 00:38:39	The subject reaches the Makahiki landing page successfully, and correctly types Ctrl+C to stop the server.
2.1.1.1.1.5.1. Initialize Makahiki	00:39:40 - 00:40:16	The subject types <code>cd /vagrant/makahiki,</code> which works, but takes the subject to the same directory as the current working directory.
	00:30:49	The subject asks how to type an underscore (the subject is unfamiliar with the keyboard layout).
	00:41:07 - 00:44:26	The subject successfully runs the command <code>./scripts/initialize_instance.py --type default.</code>
2.1.1.1.1.5.2. Update the Makahiki Instance	00:45:54 - 00:47:35	The subject updates the Makahiki instance successfully.

Table D.1.2.7. Subject 9 Screencast Activity Log

Task	Estimated time marker	Notes
2.1.1.1.1.1.7.2. Clone the Repository	00:03:17 - 00:04:37	The subject successfully clones the GitHub repository.
2.1.1.1.1.2.1. Set up Makahiki in the Virtual Machine	00:05:36 - 00:06:00	Subject types <code>cd makahiki</code> without errors.
	00:06:06	"vagrant up" begins to execute.
	00:06:29	Instructor: "Do you have any questions?" Subject: "No, I'm fine."
	00:07:42 - 00:10:04	Subject: "Actually, what is Vagrant?" Instructor: "Vagrant is a command-line application that uses the VirtualBox API, Vboxmanage, to issue commands to create and destroy virtual machines. You can change network configurations by downloading script files. The Makahiki repository that you just downloaded has one of those script files at its root, that defines it as a shared folder when the virtual machine gets created. Vagrant also allows you to define something called a provisioning script. A provisioning script is a list of commands that run the first time it starts. The idea is for the provisioning script to configure and install anything you might need when it runs automatically. You can also work with VMWare, but we're not using that today. That is a premade virtual machine called a "box." There are many "boxes" you can download. The idea is that with one box you can make many virtual machines."
	00:11:19-	Subject: "And what's Makahiki?"

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	00:13:45	<p>Instructor: "Makahiki is a web application powered by Python and Django. It provides a framework - a collection of tools used to teach about sustainability and sustainable energy use. It comes with a number of premade questions and activities, and it can hook into energy meters in the building to determine how people are using energy. If you've ever seen Kukui Cup banners around campus, Makahiki is the software that powers that, for the people in the round towers. And the next one, as I understand it, will be starting soon."</p> <p>Subject: "And it's using the virtual machines for..."</p> <p>Instructor: "The virtual machines are - are obviously not for actual use. They're for developers - people who want to try out Makahiki without setting up a full server. It enables Vagrant to start and run on Windows. Before, it was easier to install on Linux, somewhat easy on OS X, and much more difficult on Windows. So this provides a way to get the same Ubuntu Linux virtual machine running on Linux, Windows, and Mac OS X. And VirtualBox is cross-platform."</p>
	00:22:17	The "vagrant up" command finishes.
2.1.1.1.1.2.2. Connect to the Vagrant Virtual Machine	00:23:48 - 00:24:09	Subject successfully runs "vagrant ssh" command.
	00:24:50 - 00:26:45	Subject accidentally starts the server. The instructor tells the subject to stop the timer, and stops the server with Ctrl+C.
2.1.1.1.1.2.3. Start the Server	00:27:06 - 00:27:50	Subject starts the Makahiki server successfully.
2.1.1.1.1.2.4. Verify that Makahiki is Running	00:28:47 - 00:29:31	Subject views the landing page but forgets to start the timer.
	00:30:02 - 00:30:40	Subject starts the server and views the landing page again, then closes the server with Ctrl+C. This attempt is timed.
2.1.1.1.1.5.1. Initialize Makahiki	00:31:39 - 00:34:55	Subject initializes the Makahiki instance.
2.1.1.1.1.5.2. Update the Makahiki Instance	00:35:54 - 00:37:10	Subject successfully updates the Makahiki instance.

Appendix D.2. Makahiki SGG Configuration Log Data

Appendix D.2.1. Survey Data

SIMPLIFYING SUSTAINABILITY GAME DESIGN

The survey which provided this data can be found in Appendix B.2. Completion times in this section are written in the format hours:minutes:seconds. Students whose names appeared on a signup sheet in even-numbered positions 2 through 20 were assigned to Group 2. Subjects were tested in this order: 18, 20, 12, 4, 2, 6. Subjects 8, 10, 14, and 16 cancelled their appointments or did not make an appointment after signing up.

Table D.2.1.1. Subject 18 Survey Form

Question or task name	Sub-question	Response
Computer Experience		3
Operating System Experience		Microsoft Windows
Web Browser Experience		Google Chrome
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:21
	Rate the difficulty of this task.	1
	Record any questions or confusion that arise from reading this section:	There were no questions.
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:43
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no questions.
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:28
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	When creating a new draft the instructions say "If there are no Draft grids when you start the Designer Makahiki presents you with the 'First Draft' choice dialog shown below.," which did not happen. In m experience I had to select default from the a drop down and click on a new draft button.
2.3.6.3.4.3.2. Step 2: Design	Did you complete this task?	Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

the “paths” that players take through the SGG	Record the time it took you to complete this task.	0:05:08
	Rate the difficulty of this task.	2
	Record any confusion or questions that arise from reading this section:	Wasn't sure when to stop reading.
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:51
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	I was not sure why a slug name was needed for a submitted predicate.
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:50
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No questions.
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:17
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	It was a bit hard to find an activity due to its sort.
Add a premade Commitment Action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:09
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No questions.
Add a premade Event action	Did you complete this task?	No
	Record the time it took you to complete this task.	0:03:34
	Rate the difficulty of this task.	No rating (task not completed)
	Record any problem(s) you encountered in this step.	I got tell the set date button click, then after clicking on the button I come to a Lights off. Game on! sorry, but the page you requested could not be found error. @ http://192.168.56.4:8000/smart_grid_design/set_event_date/new-demo-game/
Add a fourth action	Did you complete this task?	Yes
	Record the time it took you to	0:05:27

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	complete this task.	
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	After several saving attempts errors accord; therefore we logout of the system and relogged back in. Recreated ONLY ONE level 4 as where before i had accidentally created two level 4's from multiple clicks on the save draft button. After creating only one level 4 everything ran smoothly and I was able to complete previous steps.
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:00
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No questions.
Overall Usability Evaluation	Rating	3
	Long-Form Response	The interactive GUI that let the designer drag and drop the Actions and Column Names on to the grid made it immensely easy for me to create and recreate the test grid.
Overall Documentation Evaluation	Rating	3
	Long-Form Response	I didnt really understand what the term slug refers to, but after an brief explanation from my proctor i grasped the concept easily. It made the predicate conditions easier to implement and understand as well.
Total time		0:28:48

Table D.2.1.2. Subject 20 Survey Form

Question or task name	Sub-question	Response
Computer Experience		3
Operating System Experience		Microsoft Windows
Web Browser Experience		Google Chrome
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:38

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Rate the difficulty of this task.	2
	Record any questions or confusion that arise from reading this section:	There were no problems for this section.
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:43
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	There were no problems for this section.
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:42:37
	Rate the difficulty of this task.	5
	Record any problem(s) you encountered in this step.	There were technical issues. Se[r]ver issues with the Makahiki site forced a restart of the system.
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:38
	Rate the difficulty of this task.	3
	Record any confusion or questions that arise from reading this section:	It was slightly confusing to understand what predicates to use,
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:57
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	It wasn't quite clear for what to put for the Unlock portion of creating a new level.
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:05
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	There were no problems for this task.
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:14
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no issues with this task.
Add a premade Commitment	Did you complete this task?	Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Action	Record the time it took you to complete this task.	0:00:12
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	There were no issues in completing the task.
Add a premade Event action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:39
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Slight confusion on placement of things to complete the instructions.
Add a fourth action	Did you complete this task?	No
	Record the time it took you to complete this task.	No time given (not completed before end of session)
	Rate the difficulty of this task.	No rating
	Record any problem(s) you encountered in this step.	Ran out of time.
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	Did you complete this task?	No
	Record the time it took you to complete this task.	No time given (not completed before end of session)
	Rate the difficulty of this task.	No rating
	Record any problem(s) you encountered in this step.	Ran out of time.
Overall Usability Evaluation	Rating	3
	Long-Form Response	In some ways it was easy to make the game because of color coded modules with drag and drop features. It was difficult however to find certain things because they weren't ordered in any particular manner.
Overall Documentation Evaluation	Rating	3
	Long-Form Response	The documentation was a bit difficult to understand without sufficient images that showed specifically how things were to be implemented. For example more images for the predicates would have been helpful.
Total time		0:57:43*
*All tasks were not completed in this session, so this total time was not counted towards statistical analysis of total task completion time. The researcher instructed the subject to fill out the remaining tasks as not completed when the scheduled one-hour session had ended.		

Table D.2.1.3. Subject 12 Survey Form

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Question or task name	Sub-question	Response
Computer Experience		4
Operating System Experience		Microsoft Windows
Web Browser Experience		Google Chrome
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:52
	Rate the difficulty of this task.	3
	Record any questions or confusion that arise from reading this section:	No problems occurred.
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:47
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	The instructions involved a lot of information that was not necessary to complete this specific task.
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:09
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	The default template is not the default template from the drop down menu, not a real problem, but usability could be increased.
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:13
	Rate the difficulty of this task.	3
	Record any confusion or questions that arise from reading this section:	N/A
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:33
	Rate the difficulty of this task.	4
	Record any problem(s) you encountered in this step.	Not sure if the task was completed properly or not.
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:54

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Had trouble finding the column names, they look familiar to the library action commands.
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:14
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	None
Add a premade Commitment Action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:10
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	N/A
Add a premade Event action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:36
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	N/A
Add a fourth action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:02
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	N.A
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:42
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Not sure if task was completed properly.
Overall Usability Evaluation	Rating	3
	Long-Form Response	The interface was simple upon looking at, and did not overwhelm me as a user. But I was confused by a couple of the instructions due to not noticing them inside of the interface. There is a lot going on in the interface so its difficult to completely know where you need to be when you want to do a certain thing.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Overall Documentation Evaluation	Rating	3
	Long-Form Response	It was easy because of the pictures and lack of real complication in the instructions. But the instructions seemed to miss some obvious statements, perhaps assuming the user would know. For example the GCC instruction had everything I needed, yet I had no idea where the GCC was. Perhaps that could have been mentioned in the instruction instead of letting the user search for it.
Total time		0:14:12

Table D.2.1.4. Subject 4 Survey Form

Question or task name	Sub-question	Response
Computer Experience		3
Operating System Experience		Microsoft Windows
Web Browser Experience		Microsoft Internet Explorer
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:59
	Rate the difficulty of this task.	2
	Record any questions or confusion that arise from reading this section:	Read past expected passage
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:47
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Realized opening window on separate window
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:51
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	Task completed, confusing pathing
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take	Did you complete this task?	Yes
	Record the time it took you to	0:03:13

SIMPLIFYING SUSTAINABILITY GAME DESIGN

through the SGG	complete this task.	
	Rate the difficulty of this task.	3
	Record any confusion or questions that arise from reading this section:	Did not understand whether to attempt something or just read and understand documentation
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:40
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Directions should be on the same page for easier format and solutions
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:25
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Images do not specify any particular places on the page
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:25
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No issues
Add a premade Commitment Action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:19
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No issues
Add a premade Event action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:05:46
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Specifications are needed, otherwise the text unlock condition and text will be confused
Add a fourth action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:56
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	Easier because a similar step was completed before hand
2.3.6.3.4.2. Grid Consistency	Did you complete this task?	Yes

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Checker (GCC) Tools	Record the time it took you to complete this task.	0:02:07
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No issues
Overall Usability Evaluation	Rating	3
	Long-Form Response	By taking time to read the directions, it is relatively easy, but on time based completion, the images should be more defined in specific areas.
Overall Documentation Evaluation	Rating	2
	Long-Form Response	The syntax of the performed actions were a bit confusing, but overall easy enough to grasp with some time.
Total time		0:25:28

Table D.2.1.5. Subject 2 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Operating System Experience		Microsoft Windows
Web Browser Experience		Google Chrome
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:28
	Rate the difficulty of this task.	2
	Record any questions or confusion that arise from reading this section:	N/A
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:13
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	I overlooked the instruction and skipped the sgg conf part
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:20
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	instruction is pretty clear.
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take	Did you complete this task?	Yes
	Record the time it took you to	0:01:28

SIMPLIFYING SUSTAINABILITY GAME DESIGN

through the SGG	complete this task.	
	Rate the difficulty of this task.	2
	Record any confusion or questions that arise from reading this section:	n/a
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:07:56
	Rate the difficulty of this task.	4
	Record any problem(s) you encountered in this step.	The picture for the unlock commands need to be bigger and closer to the instruction. so the users don't overlook.
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:02
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	clear instruction.
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:19
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	n/a
Add a premade Commitment Action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:14
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	n/a
Add a premade Event action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:23
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	n/a
Add a fourth action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:20
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	n/a
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	Did you complete this task?	Yes
	Record the time it took you to	0:01:20

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	complete this task.	
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	The instruction did not specify where to look for the GCC Tools.
Overall Usability Evaluation	Rating	2
	Long-Form Response	It is easy in a way that you don't need to code that much because the components are all drag-and-drop. The difficult part of this is you need to know where to look for and what you are doing when using this tool. Instructions are too wordy, should use more picture to illustrate the function of the tools.
Overall Documentation Evaluation	Rating	3
	Long-Form Response	More pictures need to be added when telling the users what to do. .
Total time		0:23:03

Table D.2.1.6. Subject 6 Survey Form

Question or task name	Sub-question	Response
Computer Experience		4
Operating System Experience		Mac OS X
Web Browser Experience		Mozilla Firefox
Makahiki Experience		No
2.3.6.3.1. About the Smart Grid Game	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:30
	Rate the difficulty of this task.	2
	Record any questions or confusion that arise from reading this section:	No questions and no confusion
2.3.6.3.4: Implementing your Smart Grid Game with the Designer: Open the Smart Grid Game Designer	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:12
	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	No issues
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:48

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Rate the difficulty of this task.	3
	Record any problem(s) you encountered in this step.	Several typos Not understanding or knowing what the Kukui Cup is
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:03:20
	Rate the difficulty of this task.	2
	Record any confusion or questions that arise from reading this section:	I did not understand the difference between a submitted action and an approved action.
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:38
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No problems
2.3.6.3.4.3.4. Step 4: Design the layout: Add Column Names	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:56
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No problems
Add a premade Activity action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:27
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No problems
Add a premade Commitment Action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:00:14
	Rate the difficulty of this task.	1
	Record any problem(s) you encountered in this step.	No problems
Add a premade Event action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:02:45
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	No problems
Add a fourth action	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:21
	Rate the difficulty of this task.	1

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	Record any problem(s) you encountered in this step.	No problems
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	Did you complete this task?	Yes
	Record the time it took you to complete this task.	0:01:31
	Rate the difficulty of this task.	2
	Record any problem(s) you encountered in this step.	At first I didn't know the GCC was on the bottom section of the page and had difficulty finding it
Overall Usability Evaluation	Rating	2
	Long-Form Response	It was very easy to use once I understood what I was doing. However, at first I didn't understand what it was for and had slight issues understanding what I was doing.
Overall Documentation Evaluation	Rating	3
	Long-Form Response	The documentation to actually building the grid was easy, but at first I didn't understand what the terminology referred to (such as levels/actions).
Total time		0:19:42

Appendix D.2.2. Screencast Activity Logs

The screencasts on which these logs are based have been uploaded to Youtube as private videos. Access to these videos is available through the Collaborative Software Development Laboratory upon request. Estimated time markers are in the format hours:minutes:seconds.

Table D.2.2.1. Subject 18 Screencast Activity Log

Task	Estimated time marker	Notes
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	00:12:23 - 00:13:37	<p>The subject is completing the survey after finishing the task. The subject says that the image in the documentation that says "First Draft" is incorrect, and that the pictured popup window no longer appears automatically when you open the SGG for the first time.</p> <p>The instructor tells the subject to write up this issue under the "Record any problems you experienced in this section" text field in the survey.</p>

SIMPLIFYING SUSTAINABILITY GAME DESIGN

2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	00:14:52 - 00:20:10	Subject: "How do you unlock the - whatever this is - conditions?" Instructor: "We'll use unlock conditions in a later step." Instructor: "You will need to read the predicate section." Subject: "This one?" Instructor: "Not that one. The other one." Subject: "I already read this." Instructor: "Okay, then you can move on to the next task."
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	00:24:36	The subject accidentally uses the predicate "approved_action("hot-spot")" instead of the predicate specified in the instructions, "submitted_action("hot-spots")."
	00:25:29 - 00:26:00	Subject accidentally creates two new levels named "Level 4." The instructor tells the subject to delete it. This causes an error page to appear: "There appears to have been an error processing your request." The instructor tells the subject to ignore this issue and use only one of the "Level 4" levels for now.
	00:27:14	The instructor explains slug names to the subject.
Add a premade Event action	00:33:53 - 00:37:26	Trying to edit and save the date on the Event action "Your Future" after it is placed in the grid causes an error page to appear: "Sorry, but the page you requested could not be found." The error occurs again. The subject was instructed to stop the timer and declare the task unfinished on the survey form.
Add a fourth action	00:39:50 - 00:40:31	The subject triggers the same "Sorry, but the page you requested could not be found" error page after trying to edit the "OTEC Energy Video" Activity action. The instructor instructs the subject to log out and log in again.
	00:41:07	The two Level 4s are still visible after logging in again. They cannot be deleted. The subject is instructed to delete the draft grid and start over.
	00:41:38	Level 4 is re-created with the same unlocking predicate.
	00:42:56	"Your Future" (the Event action from the previous task) is successfully edited and saved.
	00:43:44 - 00:45:01	The "OTEC Video" action is edited and saved.

Table D.2.2.2. Subject 20 Screencast Activity Log

SIMPLIFYING SUSTAINABILITY GAME DESIGN

Task	Estimated time marker	Notes
N/A	None	Due to networking issues that required multiple restarts, this session was recorded in 3 videos. Time markers for this video are prefaced with "Part 1," "Part 2," or "Part 3" to designate their position in each video.
N/A	Part 1 00:01:00	Firefox error: "Firefox can't establish a connection to the server at 192.168.56.4:8000."
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	Part 1 00:12:00-00:12:29	Firefox error: "Firefox can't establish a connection to the server at 192.168.56.4:8000." The instructor reloads the page with no effect.
	Part 1 00:13:08	The instructor restarts the web server in Vagrant. This has no effect.
	Part 1 00:14:00 - 00:17:28	The instructor re-initializes the database from 00:14:00 - 00:17:15, then re-starts the server. This results in the same error message.
	Part 1 00:18:00	The instructor stops the screencast and restarts the computer.
	Part 2 00:00:00	The instructor resumes the screencast and restarts the virtual machine.
	Part 2 00:04:00	Firefox error: "The connection to the server was reset while the page was loading."
	Part 2 00:04:13 - 00:04:26	Firefox error: "Firefox can't establish a connection to the server at 192.168.56.4:8000." The instructor stops recording and completely shuts down the computer.
	Part 3 00:00:00	The instructor resumes the recording and restarts the virtual machine.
	Part 3 00:03:55 - 00:05:37	The instructor restarts the web server and successfully accesses it at 192.168.56.4:8000. The subject logs in and successfully creates the level.
2.3.6.3.4.3.2. Step 2: Design the "paths" that players take through the SGG	Part 3 00:07:50	The instructor tells the subject to mark any remaining tasks as unfinished after the hour allocated for the session is up.
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	Part 3 00:11:56 - 00:13:55	The subject uses "Hot Spots" (the activity name, but not the slug name) for the "Unlock condition" field's predicate. The instructor tells the subject the correct name of the slug, and the subject fixes it.
Add a premade Event action	Part 3 00:19:45 - 00:23:19	The subject edits the Event action "Your Future" with the required date and location data without experiencing or reporting any issues, and uses the correct unlocking predicate "submitted_action("take-survey")." While typing the predicate, the subject sees a browser tooltip that offers to auto-fill the field with the predicate that was typed to create the level.

Table D.2.2.3. Subject 12 Screencast Activity Log

Task	Estimated time marker	Notes
N/A	00:00:44	Subject accidentally connects to 192.168.56.4 instead of 192.168.56.4:8000.
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	00:12:14 - 00:13:46	Subject inputs the specified unlocking predicate "submitted_action("hot-spots")" without errors.
Add a premade Event action	00:20:28 - 00:23:00	Subject inputs the specified unlocking predicate "submitted_action("take-survey")" without errors.
Add a fourth action	00:24:47 - 00:25:49	Subject inputs the specified unlocking predicate "submitted_action("take-survey")" without errors.

Table D.2.2.4. Subject 4 Screencast Activity Log

Task	Estimated time marker	Notes
2.3.6.4. Implementing your Smart Grid Game with the Designer: Create a Draft Grid.	00:10:01 - 00:12:00	The instructor mistakenly thinks that the subject created the grid twice, but this is not true and the grid is created successfully.
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	00:13:20 - 00:16:36	Subject: "I'm just confused on this. It says the admin interface, but I don't see it." Instructor: "This SGG is part of the admin interface. Most of the admin interface is under settings, but we're working with just this part." Subject: "Okay." Instructor: "These predicate options are part of deeper-level settings for each of these activities." Subject: "So I need to choose one for each of them." Instructor: "So to recap, each of these actions or other things have an unlock condition predicate somewhere."
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	00:18:10 - 00:21:49	The subject mistypes "hot-spots" (the slug name) as the unlock condition for the level, without enclosing it in a predicate. The instructor corrects the subject and the subject corrects the mistake.
Add a premade Event action	00:29:26 - 00:35:50	Subject adds the Event action and edits its date and location without experiencing or reporting any issues.
	00:31:58 - 00:32:08	Subject mistakenly types the unlocking predicate in the "Unlock condition text" field, instead of "Unlock condition."
	00:33:02	The action is saved with the incorrect predicate.

SIMPLIFYING SUSTAINABILITY GAME DESIGN

	00:33:40	The subject accidentally stops the timer. The instructor tells the subject to write its current time on a sheet of paper and combine it with the time that will be spent to fix the predicate.
	00:33:45 - 00:35:50	The timer is cleared and restarted. The subject fixes the unlock predicate.
Add a fourth action	00:38:15 - 00:41:10	Subject types the predicate "submitted_action("Turn Off Vampires")" in the "Unlock condition" field. This is incorrect because the slug name "turn-off-vampires" should be used in the predicate, not the human-readable name. The subject corrects this.
2.3.6.3.4.2. Grid Consistency Checker (GCC) Tools	00:42:43 - 00:44:44	The GCC Tool identifies a typo in the unlock condition for the Event action added in a previous step: its predicate uses "Take Survey" instead of slug name "take-survey." The subject fixes this error.

Table D.2.2.5. Subject 2 Screencast Activity Log

Task	Estimated time marker	Notes
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	00:14:05 - 00:15:29	Subject clicks the column name "Get Started." The instructor reminds the subject that the goal is to create a new level, not to change the columns.
	00:18:29 - 00:19:01	Subject types "Hop Spots" (a misspelled action name instead of slug name "hot-spots") in the "Unlock condition" field and presses "Submit."
	00:19:18 - 00:20:48	The instructor asks the subject to delete Level 4. The instructor explains slug names and tells the subject to re-read the article.
	00:20:48 - 00:21:27	The subject types "hot-spots" in the "Unlock condition" field. The instructor tells the subject that this is incomplete and that the full predicate needs to be used. The subject edits the "Unlock condition" field to use "submitted_actions("hot-spots"), which is correct.
Add a premade Event action	00:28:20 - 00:31:43	The subject correctly types the unlock condition "submitted_action("take-survey")" with no errors.
Add a fourth action	00:33:00 - 00:34:18	The subject correctly types the unlock condition "submitted_action("turn-off-vampires")" with no errors.

Table D.2.2.6. Subject 6 Screencast Activity Log

Task	Estimated time marker	Notes
2.3.6.4. Implementing your Smart Grid Game	00:14:40 - 00:15:52	Subject: "I don't see the screen with the default template or the default draft."

SIMPLIFYING SUSTAINABILITY GAME DESIGN

with the Designer: Create a Draft Grid.		Instructor: "It is a thing that appears once you click on it." Subject: "Okay." The subject then mistypes the draft grid name as "New emo game" instead of "New Demo Game," but completes the operation successfully.
	00:17:22 - 00:19:02	The subject forgot to start the timer when the subject began reading the documentation, so the grid is deleted and the timer is cleared and restarted. The subject then re-creates the draft grid according to the instructions and times it.
2.3.6.3.4.3.2. Step 2: Design the “paths” that players take through the SGG	00:20:41 - 00:24:00	While reading the documentation, the subject asks if the subject will be copying what is in the documentation later on. The instructor says that the subject will be doing something like this later, but should read the documentation for now.
2.3.6.3.4.3.4. Step 4: Design the layout: Create a level	00:25:00 - 00:27:30	The subject creates the level correctly, using the "submitted_action("hot-spots")" predicate. However, the subject forgot to start the timer, so the level was deleted.
	00:28:43 - 00:30:17	The subject re-creates the level while being timed.
Add a premade Event action	00:36:10 - 00:38:50	The subject adds and edits the Event action "Your Future" correctly. The unlock condition predicate is correctly set to "submitted_action("take-survey")."
Add a fourth action	00:40:08 - 00:41:35	The subject adds and edits the Activity action "OTEC Video" correctly. The unlock condition predicate is correctly set to "submitted_action("turn-off-vampires")."

Bibliography

- [1] R. S. Brewer, G. E. Lee, and P. M. Johnson, "The Kukui Cup: a Dorm Energy Competition Focused on Sustainable Behavior Change and Energy Literacy," in *Proceedings of the 44th Hawaii International Conference on System Sciences*, 2011, 10 pp, non-paginated. [Online]. Available: <http://csdl.ics.hawaii.edu/techreports/10-07/10-07.pdf>

A conference paper by Dr. Philip Johnson and two other researchers from the University of Hawaii, describing the capabilities of Makahiki and WattDepot prior to the first Kukui Cup in 2011. It frames the value of Makahiki in terms of raising students' awareness of their energy consumption patterns by collecting and providing feedback and real-time statistics.

- [2] F. Calisir and F. Calisir, "The relation of interface usability characteristics, perceived usefulness, and perceived ease of use to end-user satisfaction with enterprise resource planning (ERP) systems," *Computers in Human Behavior*, vol. 20, no. 4, pp. 505-515, 2004.

The authors, researchers at Istanbul Technical University in Turkey, used a survey to analyze user satisfaction with enterprise resource planning systems, which consist of "one database, one application, and a standard interface across the entire enterprise," broadly similar to Makahiki's dependence on a competition server accessed via a standard web interface. Perceived usability was found to be the most reliable predictor of satisfaction. It was dependent on how easily the interface could be learned, and how well the system's features met user requirements.

- [3] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From Game Design to Gamefulness: Defining "Gamification"," in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, 2011, pp. 9–15.

The authors are from Hamburg University (Germany), University of the West of England (Bristol, United Kingdom), IT University of Copenhagen (Denmark), and the University of Ontario Institute of Technology (Oshawa, Canada). This literature review attempts to define the meaning of the term "gamification," which has seen increasing use to describe a software application which adopts elements of video games in the service of promoting a service, product, or type of social change. Differentiating between "serious games" and "gamified applications," the authors conclude that gamification refers to the "use [...] of design [...] elements [...] characteristic for games [...] in non-game contexts (15)" without regard to how the application was intended to be used, what its context is, or what medium it was designed for.

- [4] Y. Dittrich, M. John, J. Singer, and B. Tessem, "Editorial: For the Special Issue on Qualitative Software Engineering Research," *Information and Software Technology*, vol. 49, no. 6, pp. 531-539, June 2007.

This editorial provides an overview of differing methods used to integrate qualitative methods into the normally quantitative discipline of software engineering. The authors

argue that qualitative research methods can collect data on subjective parts of the user experience that quantitative research cannot measure. One of the methods discussed is "action research," which cycles between usability testing and the development of new software with the goal of producing iterative design improvements.

- [5] X. Fang and C. W. Holsapple, "An empirical study of web site navigation structures' impacts on web site usability," *Decision Support Systems*, vol. 43, no. 2, pp. 476–491, Mar. 2007.

The authors, researchers at Miami University and University of Kentucky, provide evidence in support of the use of a usage-oriented hierarchy or a combined hierarchy instead of a subject-oriented hierarchy in the design of a web site's navigation structure. Tasks of varying levels of difficulty were developed for use with a customized web site, in which test subjects were required to perform certain tasks within the user interface and their satisfaction, ease of use, correct answer ratio, and navigation time were assessed. In the first round of testing, usage-oriented hierarchy and combined hierarchy had statistically significant lower navigation times. In the second round, users reported statistically significant higher satisfaction and general ease of use for a usage-oriented hierarchy or combined hierarchy than for a subject-oriented hierarchy.

- [6] T. Hewett, R. Baecker, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong, and W. Verplank, "Curricula for Human-Computer Interaction: Chapter 2: Human-Computer Interaction," in *ACM SIGCHI Curricula for Human-Computer Interaction*, 1996. [Online]. Available: <http://old.sigchi.org/cdg/cdg2.html>

A curriculum by the Association for Computing Machinery (ACM) which provides a rough outline of recommended lesson content for courses taught in human-computer interaction education. This chapter presents a working definition of human-computer interaction and of the process of analyzing user interfaces.

- [7] A.R. Hevner, S.T. March, J. Park, and S. Ram, "Design Science In Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75-105, March 2004.

The authors, researchers at the University of South Florida, Vanderbilt University, Korea University, and University of Arizona, respectively, propose a paradigm for information systems research and design and examine elements of this paradigm which are present in three case studies. This study was important to this project primarily for its description of an iterative process for collecting feedback, improving a product based on that feedback, and then collecting feedback again.

- [8] P.M. Johnson, Y. Xu, R.S. Brewer, C.A. Moore, G.E. Lee, and A. Connell, "Makahiki+WattDepot: An open source software stack for next generation energy research and education," in *Proceedings of the 2012 Conference on Information and Communication Technologies for Sustainability*, 2013, 8 pp., non-paginated [Online]. Available: <https://csdl-techreports.googlecode.com/svn/trunk/techreports/2012/12-06/12-06.pdf>

A conference paper by members of the Makahiki development team which describes the nature of the integration of WattDepot into Makahiki and proposes the use of the combined systems for purposes of energy research and education. It describes the capabilities of WattDepot for aggregating energy data and serving it to clients, and the capabilities of Makahiki to request data from WattDepot and integrate it into sustainability-related activities are described.

- [9] P. M. Johnson, Y. Xu, R.S. Brewer, G.E. Lee, M. Katchuck, and C.A. Moore, "Beyond kWh: Myths and fixes for energy competition game design," in *Proceedings of the Meaningful Play 2012 Conference*, 10 pp., 2012, 10 pp., non-paginated [Online]. Available: http://meaningfulplay.msu.edu/proceedings2012/mp2012_submission_72.pdf

A conference paper which analyzes the results of the 2011 Kukui Cup and describes its game mechanics in detail. The authors found that many of the recorded changes in behavior during competition were unsustainable in the long term, and that data collection in this study and others failed to account for changes in energy use after the end of the competition, which the authors argue may actually be the period which is most important for identifying changes in behavior. In this paper, it was used as the basis for a description of Kukui Cup game mechanics.

- [10] A.W. Kushniruk and V.I. Patel, "Cognitive and usability engineering methods for the evaluation of clinical information systems," *Journal of Biomedical Informatics*, vol. 37, pp. 56–76, 2004.

The authors, researchers at York University (Toronto, Canada) and Columbia University (New York) respectively, conduct a literature review of research in the area of usability testing methods for medical information systems. It describes the use of surveys and video recording to collect information about a user's background and thought process. Likely due to technical limitations at the time, it recommends using an external video camera and a PC-to-video converter, instead of screen-recording software; however, the external video camera also serves to record anything the user might say, which captures his or her thought processes. It recommends the sorting of video data into coding categories for analysis, and the use of this data to improve the next iteration of the application design.

- [11] D.C. McFarlane, "Comparison of Four Primary Methods for Coordinating the Interruption of People in Human-Computer Interaction," *Human-Computer Interaction*, vol. 17, pp. 63-139, 2002.

The author, an engineer at Lockheed Martin's Advanced Technology Laboratories, analyzes effects of four possible types of interruptions to the performance of tasks in user interfaces, finding that being interrupted produces more errors and that the literature at the time did not contain much information about solving interruption problems. Four types of interruption - immediate, negotiated, mediated, and scheduled - were tested by 36 volunteers (18 men, 18 women) for an abstract computer task. Task 1 required the

user to track the positions of multiple objects on screen. Task 2, which interrupted Task 1, asked users to match shapes by shape or color. Immediate interruption randomly presented matching tasks without regard for the state of the game task; negotiated interruption notified the user and let them decide when to handle the task; mediated interruption presented matching tasks when the number of objects on screen was highest; and scheduled interruptions presented the matching game every 25 seconds. The author concluded that interruption causes statistically significant differences in user performance. This source was important to this paper primarily for its description of experiment design.

- [12] J. P. Wanderer, A. V Rao, S. H. Rothwell, and J. M. Ehrenfeld, "Comparing two anesthesia information management system user interfaces: a usability evaluation," *Canadian Journal of Anaesthesia*, vol. 59, no. 11, pp. 1023–31, Nov. 2012

The authors, researchers at Vanderbilt University (Wanderer and Ehrenfeld) and Massachusetts General Hospital (Rao and Rothwell), recorded user sessions and survey results for two anesthesia information management systems (AIMS) to test for the existence of significant differences in the usability of two AIMS user interfaces in a simulated clinical database populated by data from twenty anesthesia providers. Use of the user interface labeled as "revised" increased documentation accuracy and reduced the number of user interactions needed to complete tasks. Though the time required to document airway information was reduced, there was no significant difference in the total time required to complete intravenous documentation or to complete all tasks, and test subjects did not report a significant difference in the workload they perceived. Unlike Fang and Holsapple (2007), in which testing was conducted on a user interface constructed specifically for the study, this study by Wanderer et al. used real-world AIMS databases with simulated clinical data.

- [13] Collaborative Software Development Laboratory. (2013). *MIT LICENSE* [Markdown text file]. Available: <http://github.com/csdl/makahiki/blob/master/LICENSE.md>

This file is the LICENSE.md file made available on Makahiki's GitHub file repository. It specifies that the Makahiki source code is made available to developers under the open-source MIT license.

- [14] G.E Lee, Y. Xu, R.S. Brewer, and P. M. Johnson. (2014). *Makahiki Documentation* (Release 2) [PDF]. Available: <http://media.readthedocs.org/pdf/makahiki/latest/makahiki.pdf>

This manual by Lee et al. is a PDF version of the Makahiki framework documentation available at <http://makahiki.readthedocs.org>. Regardless of the date on its cover page, a revised version is generated whenever a change to the documentation in the project's GitHub repository is made.

- [15] A. Sayler, D. Grunwald, J. Black, E. White, and M. Monaco, "Supporting CS Education via Virtualization and Packages: Tools for Successfully Accommodating "Bring-Your-Own

-Device" at Scale," in SIGCSE 2014, March 5-8, 2014, Atlanta, GA, 6 pp., 313-318.

This paper by Sayler et al. describes the use of virtualization in support of the creation of a common software development environment across the entire computer science curriculum at the University of Colorado, Boulder. The authors focus on the benefits of virtualization to their organization, especially with respect to maintaining consistency despite a wide range of operating systems in use by the university's students.

[16] J. Palat, "Introducing Vagrant," *Linux Journal*, vol. 2012, no. 220. [Online]. Available: <http://www.linuxjournal.com/content/introducing-vagrant>

Palat's article, though somewhat dated in its description of Vagrant's state of development, offers an overview of the rationale behind its creation and the problems which it was designed to address - in particular, the problem of maintaining a consistent configuration across the virtual machines of multiple members of a development team